ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO

FACULTAD DE INFORMATICA Y ELECTRONICA

ESCUELA DE INGENIERIA ELECTRONICA

"ANALISIS DE LAS TECNICAS DE CONVIVENCIA ENTRE IPV4 E IPV6 Y SU IMPLEMENTACION EN LOS SERVICIOS: WEB, MAIL, FTP, PROXY, DNS Y DHCP DE LA INTRANET DE LA ESPOCH"

TESIS DE GRADO

Previa la obtención del título de

INGENIERO EN ELECTRONICA Y COMPUTACION

Presentado por:

MIGUEL ANGEL CARRERA BUENAÑO

2009

Ing. Byron Vaca B.

Colaborador de Tesis; por su ayuda y colaboración para la realización de este trabajo.

Dr. Carlos Buenaño P.

Por haber sido el soporte en el desarrollo de esta Tesis a través de sus conocimientos y experiencias.

Hoy al culminar otra etapa de mi vida, es tan hermoso tener tanto que agradecer al artista que un día combino amor, dulzura y esperanza que es Dios y a su obra maestra que son mis abnegados Padres porque son los que me dieron la vida, quienes con su amor y comprensión me ayudaron en los momentos difíciles.

NOMBRE	FIRMA	FECHA
Dr. Romeo Rodríguez DECANO DE LA FACULTAD DE INFORMATICA Y ELECTRONICA	_ _ _ _ _ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _ _ _ _ _
Ing. Paúl Romero DIRECTOR DE LA ESCUELA DE INGENIERIA ELECTRONICA	_ _ _ _ _ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _ _ _ _ _
Ing. Byron Vaca B. DIRECTOR DE TESIS	_ _ _ _ _ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _ _ _ _ _
Dr. Carlos Buenaño P. MIEMBRO DEL TRIBUNAL	_ _ _ _ _ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _ _ _ _ _
Tec. Carlos Rodríguez DIRECTOR DEL CENTRO DE DOCUMENTACION	_ _ _ _ _ _ _ _ _ _ _ _	_ _ _ _ _ _ _ _ _ _ _ _
NOTA DE LA TESIS	_ _ _ _ _ _ _ _ _ _ _ _	

“Yo, MIGUEL ANGEL CARRERA BUENAÑO soy responsable de las ideas, doctrinas y resultados expuestos en esta tesis; y, el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO”.

Miguel Angel Carrera Buenaño

INDICES

INDICE GENERAL

INDICES DE FIGURAS

INDICES DE TABLAS

INTRODUCCION

El Internet se ha convertido en el estándar de-facto de las comunicaciones digitales. Se ha iniciado una revolución con el advenimiento de las vías de acceso inalámbrico que se encuentran conectados permanentemente (“always on”) y en todas partes del mundo (“everywhere access”). Las industrias que están estrechamente vinculadas a las comunicaciones digitales precisan poner en marcha estrategias orientadas hacia la adaptación y obtención del máximo de beneficios que ofrece dicha evolución.

Debido a los problemas surgidos al transcurrir los años, y la dificultad que representar amplios conjuntos de direcciones IP disponibles y el deseo de proporcionar nuevas funcionalidades a ciertos dispositivos aparecidos recientemente, hay en marcha un proceso de actualización de la versión actual del protocolo IP (llamado IPv4) que está en proceso de normalización.

Los expertos de la tecnología predicen que la actual versión 4 (IPv4) del Protocolo de Internet no estará en capacidad de manejar la futura masa de usuarios, su desplazamiento inherente y la gama de características pertinentes que se requiere. Esta revolución móvil trajo como resultado el desarrollo de la naciente versión 6 (Ipv6) del Protocolo de Internet.

Dos tercios del total de direcciones IPv4 en todo el mundo ya han sido asignados dentro de la comunidad global del Internet; como consecuencia de ella, el IPv6 está ahora siendo desplegado para asimilar cuestiones que son de crítica importancia en cuanto a los actuales y futuros requisitos de los clientes.

Esta nueva versión, llamada IP Versión 6 (IPv6), resuelve ciertos problemas de diseño no previstos en la versión IPv4 de Internet, y se considera como el protocolo que definirá la Internet en el siglo XXI.

CAPITULO I

ANALISIS

1 ANALISIS

1.1 ANALISIS DE LAS TECNICAS DE CONVIVENCIA ENTRE IPV4/IPV6

IPv6 es una nueva versión de Internet Protocol, diseñada para suceder a la actual (IPv4). La transición entre ambas será un largo proceso durante el que se ha de garantizar la coexistencia. Para ello, IETF (Internet Engineering Task Force) ha creado el NGTrans Working Group.

La especificación IPv6 introduce en Internet Protocol modificaciones fundamentales. No sólo la longitud de la dirección IP ha sido extendida a 128 bits; también ha sido modificado el formato de la cabecera IP y el modo en que se procesa la información que en ella se alberga. Pasar de IPv4 a IPv6 no es sencillo y los mecanismos que permiten la coexistencia y la transición entre las dos versiones han de estar estandarizadas.

Hoy, cientos de millones de personas están conectadas a Internet y un número equivalente de hosts y dispositivos implementan el protocolo IP. El paso en un día D de

IPv4 a IPv6, siguiendo el modelo del Año 2000, no sería práctico. La migración al nuevo IP en tan corto periodo de tiempo requeriría la redefinición de un plan de direccionamiento IPv6 mundial, la instalación del protocolo en cada router y host, y la modificación de las aplicaciones actuales para que puedan soportarlo. Un proceso caro, sin duda, y que podría causar interrupciones del servicio inaceptables. Sencillamente, tal enfoque no tendría sentido, ya que muchas de las aplicaciones hoy operativas no han sido diseñadas para aprovechar las nuevas características de IPv6; ni siquiera las necesitan.

No hay una regla universal que pueda ser aplicada al proceso de transición de IPv4 a IPv6. En algunos casos, adoptar directamente, sin pasos previos, el nuevo IP será la única solución. En Asia, por ejemplo, las autoridades políticas están impulsando fuertemente IPv6 a fin de sostener el crecimiento económico de la zona, garantizando a cada ciudadano un número suficiente de direcciones IP. Asimismo, se ha de desplegar a gran escala una nueva arquitectura IP (como en el networking doméstico móvil) para proporcionar aplicaciones peer-to-peer y servicios innovadores.

Pero otros planes de transición habrán de asegurar una interoperabilidad gradual entre IPv4 e IPv6 a medida que evoluciona la transición. Es obvio que los ISP y las empresas preferirán preservar las grandes inversiones realizadas en redes IPv4. Los mecanismos propuestos por el grupo NGTrans permitirán interconectar redes IPv4 e IPv6, así como servidores y clientes basados en ambas versiones.

Algunos estudios pronostican que el periodo de transición finalizará entre 2030-2040; en algún momento de esa década, las redes IPv4 deberían haber desaparecido totalmente. Una historia realmente larga comparada con el rápido crecimiento experimentado por Internet.

1.2 LA TRANSICION A IPV6 ES UNA NECESIDAD

Es necesario empezar a migrar a IPv6. Así que, ¿por qué aún no lo hacen las organizaciones? La mayoría de los expertos en TI están de acuerdo en que la interoperabilidad es la clave. Ser capaz de comunicar ambos protocolos, es la mejor y más lógica estrategia de transición.

Los mecanismos de transición son un conjunto de mecanismos y de protocolos implementados en hosts y routers, junto con algunas guías operativas de direccionamiento designadas para hacer la transición de Internet al IPv6 con la menor interrupción posible. Para ello hay tres posibles soluciones técnicas: dual stack, tunnelling y translators.

Método de transición Dual Stack

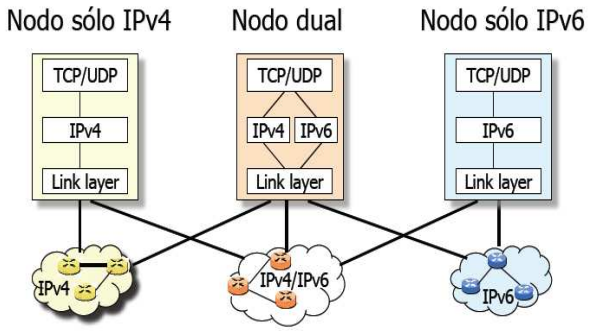

Figura I-1 Método de Transición Dual Stack

Es una de las técnicas más utilizadas, ya que puede ser utilizada en las diferentes partes de la red: equipos clientes, servidores y routers. Para que trabaje eficazmente, el dual stack debe ser implementado en todos los routers de la red. No habrá comunicación entre IPv4 e IPv6; sino que las aplicaciones tendrán que soportar ambos modos. El desafío con dual stack es que todos los equipos de la red han de contar la suficiente potencia de proceso y memoria, para gestionar dos pilas IP diferentes.

Método de transición Tunnels

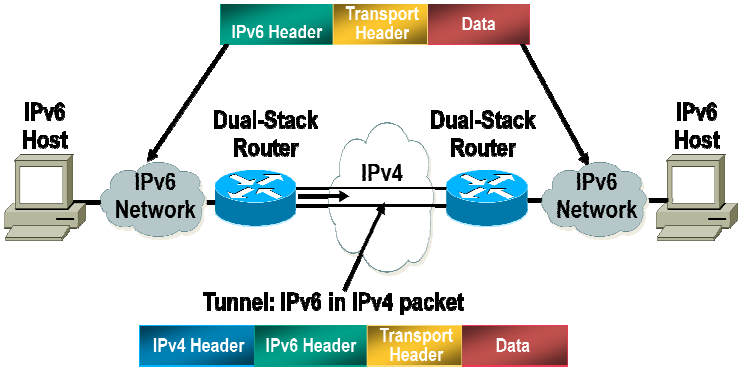

Figura I-2 Método de Transición Tunnels

Cuando un paquete IPv6 tiene que atravesar una red que sólo es IPv4, pueden utilizarse "túneles" para lograrlo. Un túnel trabaja "encapsulando" un paquete IPv6 dentro de un paquete IPv4 para que el mismo pueda viajar por estas redes. El paquete es "desencapsulado" al llegar al destino, que deberá ser un nodo IPv6 o dual stack.

Método de transición Translators

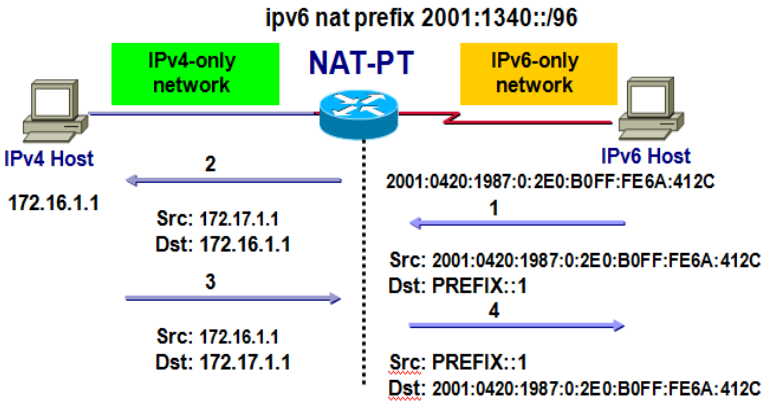

Figura I-3 Método de Transición Translators

Este mecanismo de transición realiza una "traducción" similar a la que efectúa el NAT, donde es modificada la cabecera IPv4 a una cabecera IPv6. El más conocido dentro de

este grupo es NAT-PT. Sin embargo, este tipo de mecanismos no es de los más recomendados.

1.3 Elegir la estrategia de transición más adecuada

Recientemente, el IETF (Internet Engineering Task Force) ha puesto en marcha un grupo de trabajo para investigar a fondo los diferentes métodos, y ofrecer a la comunidad técnica de Internet sugerencias y recomendaciones, que aporten más luz sobre la migración hacia el nuevo protocolo.

Lo que está claro es que IPv6 es la respuesta a la actual sociedad permanentemente conectada, repleta de pequeños dispositivos como teléfonos, PDAs, móviles. Mientras tanto, las soluciones para mejorar el uso de direcciones IPv4 en Internet no hacen más que retrasar lo inevitable.

Migrar hacia IPv6 puede ser complejo en organizaciones grandes, pero las estrategias existentes pueden ayudar mucho a facilitar esta transición. Estos mecanismos no son alternativas a otros, pero en cualquier caso, requieren conocer a fondo nuestra infraestructura, para así poder seleccionar la estrategia más apropiada para lograr nuestro objetivo. Para la mayoría de nosotros, ese objetivo es migrar a IPv6 con costos bajos y que el impacto sea mínimo.

Para la elección de que mecanismo implementar se ha resumido en el siguiente cuadro los aspectos más importantes de los mecanismos.

MECANISMOS DE TRANSICION	
Dual Stack	• Es fácil implementar. • Es la solución inmediata más accesible. • Permite que los nuevos dispositivos IPv6 relacionarse rápidamente con el resto de los dispositivos.

Tunnels	• Para configurar un túnel se necesita saber cuatro cosas esenciales: la dirección ipv4 de origen y destino que serán utilizados para la encapsulación, y las direcciones ipv6 que se utilizaran para realizar la conexión punto a punto. • Necesitaremos contar con dual stack en cada uno de los puntos del túnel. • El mecanismo para la implementación de túneles varía de una plataforma a otra. • Requiere más configuración que los otros métodos.
Translators	• Adolece los mismos problemas de NAT IPv4. • Fiabilidad • Cuello de botella • Incompatibilidad en distintas aplicaciones. • Escalabilidad • Se pierden los beneficios de ipv6.

Tabla I-1 Mecanismos de Transición

Con el análisis anterior hemos concluido que el mecanismo de transición más óptimo para implementar en la Intranet de la Escuela Superior Politécnica de Chimborazo es Dual Stack o Doble Pila.

1.4 IMPLEMENTACION DOBLE-PILA

Este enfoque de doble pila es un mecanismo fundamental para introducir IPv6 en las arquitecturas IPv4 actuales y se prevé que siga siendo muy utilizado durante el próximo futuro. Su punto flaco es que obliga a que cada máquina retenga una dirección IPv4, cada vez más escasas. Así, a medida que se difunde IPv6, la técnica de doble pila tendrá que ser aplicada allí donde específicamente ayuda al proceso de transición, por ejemplo en routers y servidores. Un servidor de doble pila puede soportar clientes sólo IPv4 convencionales, nuevos clientes sólo IPv6, y por supuesto clientes de doble pila. Para aquellos casos en que haya insuficientes direcciones IPv4 se ha definido una

combinación del modelo de conversión y de doble pila de protocolos, conocido como DSTM (Dual Stack Transition Mechanism).

1.4.1 El modelo de referencia OSI

El modelo OSI está constituido por 7 capas que definen las funciones de los protocolos de comunicaciones. Cada capa del modelo representa una función realizada cuando los datos son transferidos entre aplicaciones cooperativas a través de una red intermedia.

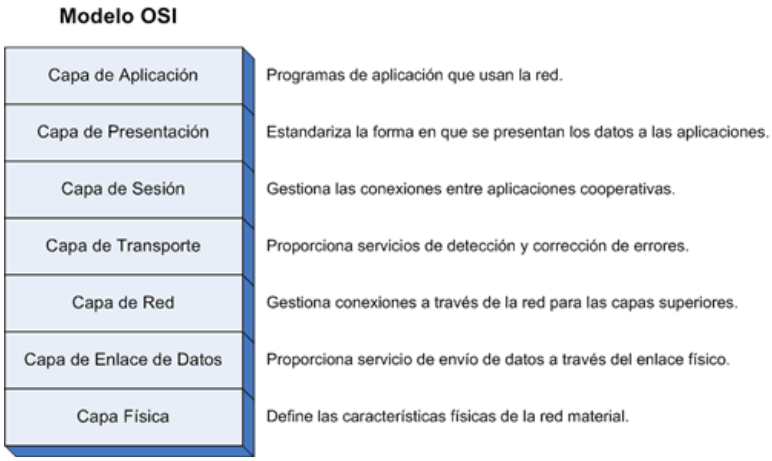

Figura I-4 Modelo OSI

1.4.2 Capas o niveles del TCP/IP

Para conseguir un intercambio fiable de datos entre dos computadoras, se deben llevar a cabo muchos procedimientos separados.

El resultado es que el software de comunicaciones es complejo. Con un modelo en capas o niveles resulta más sencillo agrupar funciones relacionadas e implementar el software de comunicaciones modular.

Las capas están jerarquizadas. Cada capa se construye sobre su predecesora. El número de capas y, en cada una de ellas, sus servicios y funciones son variables con cada tipo de red. Sin embargo, en cualquier red, la misión de cada capa es proveer servicios a las capas superiores haciéndoles transparentes el modo en que esos servicios se llevan a cabo. De esta manera, cada capa debe ocuparse exclusivamente de su nivel

inmediatamente inferior, a quien solicita servicios, y del nivel inmediatamente superior, a quien devuelve resultados.

Capas del modelo TCP/IP

Capa 4: Aplicación, asimilable a las capas 5 (sesión), 6 (presentación) y 7 (aplicación) del modelo OSI.

Capa 3: Transporte, asimilable a la capa 4 (transporte) del modelo OSI.

Capa 2: Internet, asimilable a la capa 3 (red) del modelo OSI.

Capa 1: Acceso al Medio, asimilable a la capa 1 (física) y 2 (enlace de datos) del modelo OSI.

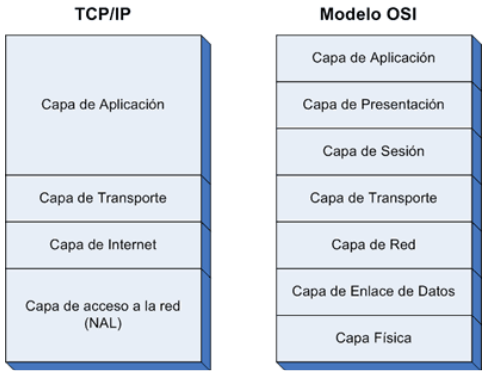

Figura I-5 Comparación en los Modelos OSI y TCP/IP

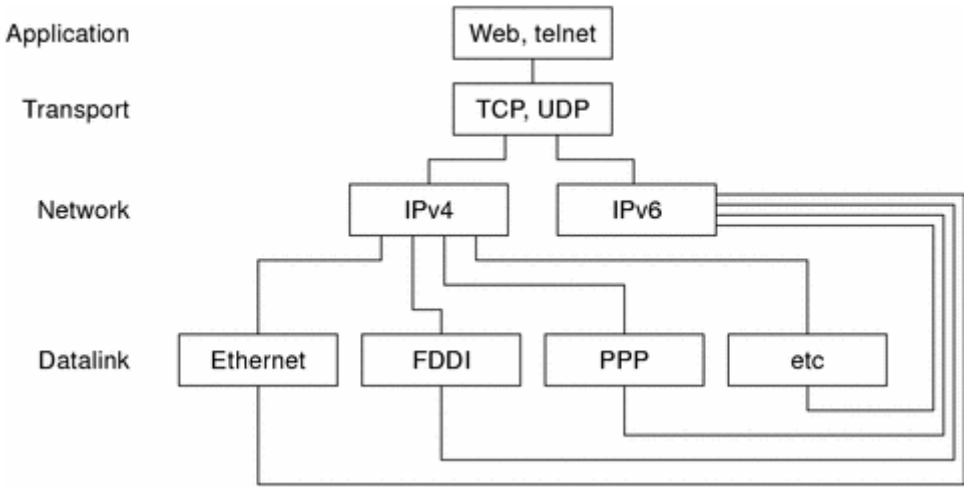

Figura I-6 Doble Pila en el Modelo OSI

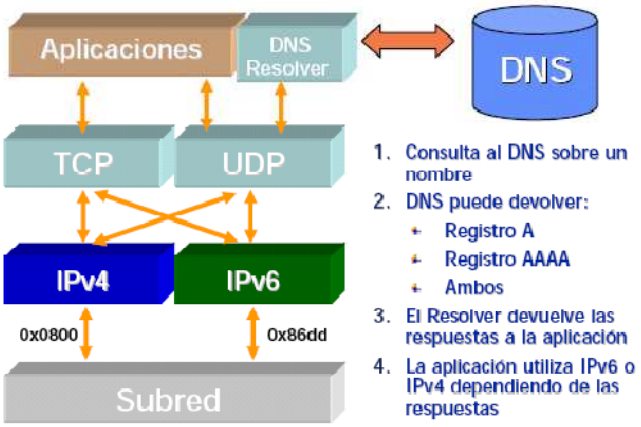

Figura I-7 Resolución del DNS en Doble Pila

1.5 ANALISIS DE LOS SERVICIOS SOBRE INTERNET QUE BRINDA LA ESPOCH

1.5.1 Servicio de Resolución de Nombres

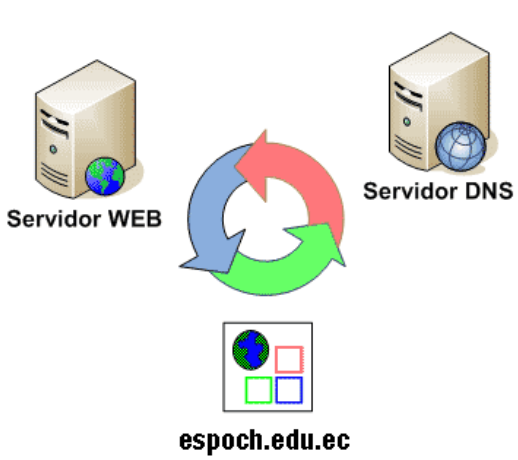

Figura I-8 Servicio de Resolución de Nombres

El Domain Name System (DNS) es una base de datos distribuida y jerárquica que almacena información asociada a nombres de dominio en redes como Internet. Aunque como base de datos el DNS es capaz de asociar distintos tipos de información a cada nombre, los usos más comunes son la asignación de nombres de dominio a direcciones IP y la localización de los servidores de correo electrónico de cada dominio. Utiliza los puertos 53/UDP, 53/TCP.

Encontramos 3 tipos de DNS:

1.- DNS Primario: el DNS primario solo otorga nombres de dominio, no consulta con otros DNS.

2.- DNS Secundario: el DNS secundario otorga, al igual que el primario, nombres de dominio, pero cuando no se encuentra el nombre de dominio preguntando por el cliente en su servidor consulta a otro DNS.

3.- DNS caché: el DNS cache consulta a otros servidores nombres de dominio.

1.5.2 Servicio de Hosting

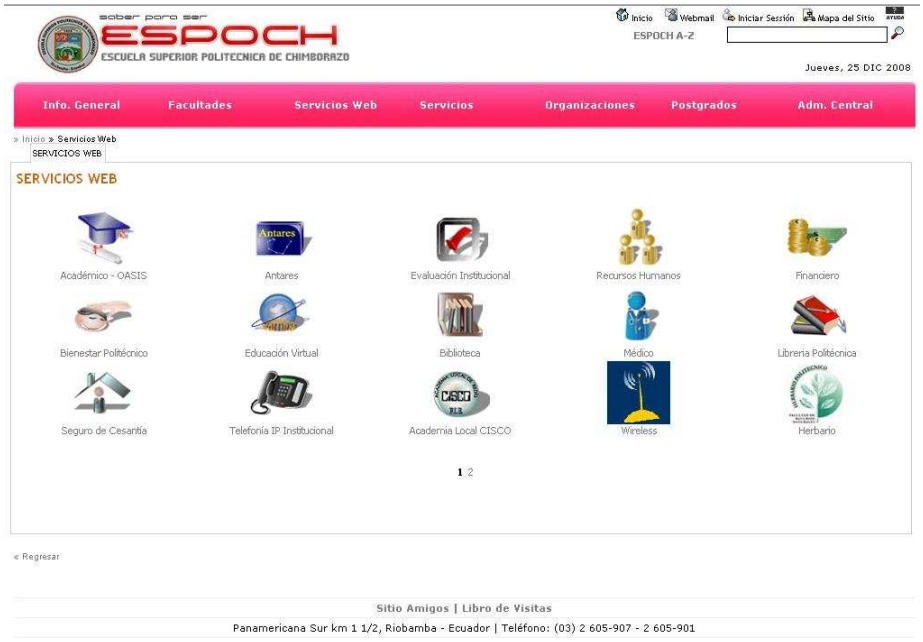

Figura I-9 Servicio de Hosting

Un servidor web es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor web se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página web textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música. o información de todo tipo de acuerdo a los comandos solicitados, usando el

protocolo HTTP o el protocolo HTTPS (la versión cifrada y autenticada). El Servicio WEB es ofrecido por la capa de Aplicación del modelo de capas de red TCP/IP al usuario, utilizando normalmente el puerto de red 80 para http y el 443 para HTTPS.

1.5.3 Servicio de Correo Electrónico

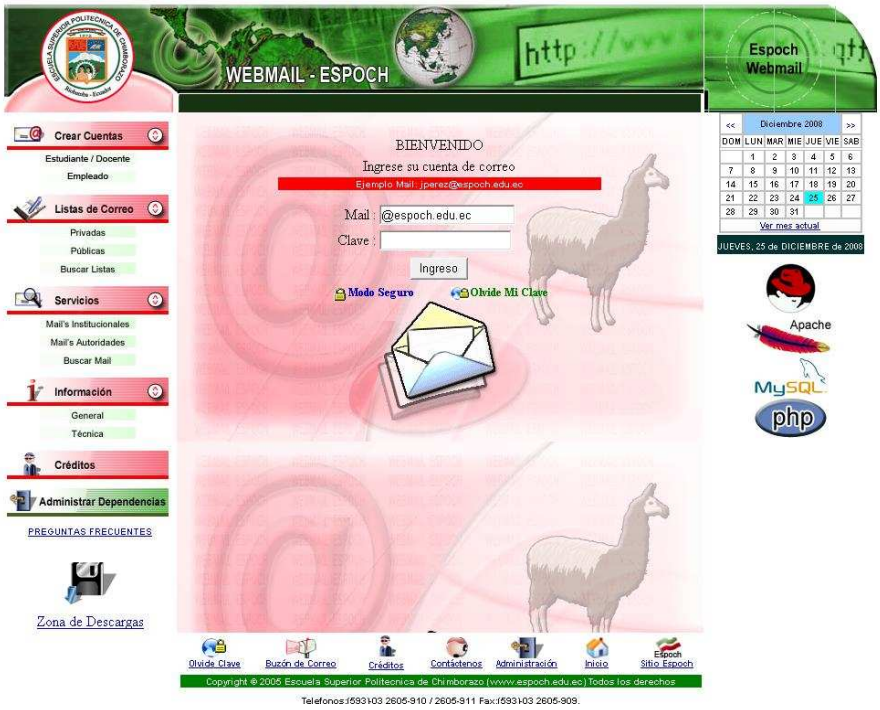

Figura I-10 Servicio de Correo Electrónico

El correo electrónico o e-mail (acrónimo de Electronic Mail) es el sistema de intercambio de mensajes entre usuarios conectados a una red electrónica. Sirve para enviar mensajes entre usuarios conectados a la misma red, o entre usuarios que tienen sus máquinas conectadas a la Red Internet. Este intercambio de mensajes entre una o varias personas se produce de forma asíncrona, por lo que no se requiere la presencia simultánea de los comunicantes.

Para lograr la conexión se definen una serie de protocolos, cada uno con una finalidad concreta:

- SMTP, Simple Mail Transfer Protocol: Es el protocolo que se utiliza para que dos servidores de correo intercambien mensajes, utilizando el puerto 25.
- POP, Post Office Protocol: Se utiliza para obtener los mensajes guardados en el servidor y pasárselos al usuario, utilizando el puerto 110.
- IMAP, Internet Message Access Protocol: Su finalidad es la misma que la de POP, pero el funcionamiento y las funcionalidades que ofrecen son diferentes, utilizando el puerto 110.

1.5.4 Servicio de Proxy

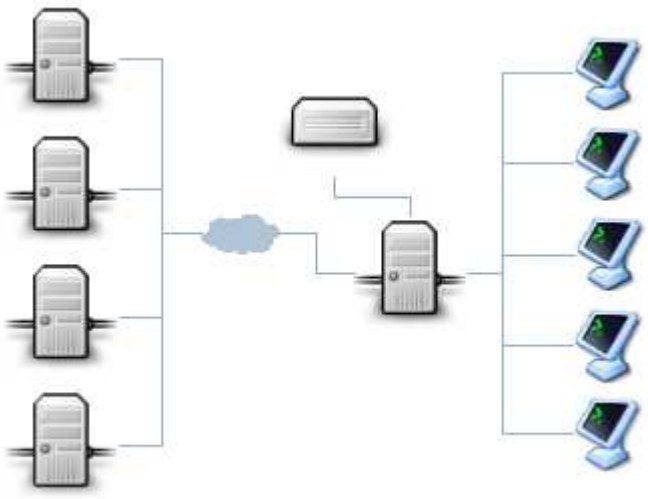

Figura I-11 Servicio de Proxy

Un servidor Proxy es un equipo que actúa de intermediario entre un explorador Web (como Internet Explorer, FireFox) e Internet. Los servidores Proxy ayudan a mejorar el rendimiento en Internet ya que almacenan una copia de las páginas web más utilizadas. Cuando un explorador solicita una página web almacenada en la colección (su caché) del servidor Proxy, el servidor Proxy la proporciona, lo que resulta más rápido que consultar la Web.

Los servidores Proxy también ayudan a mejorar la seguridad, ya que filtran algunos contenidos Web y software malintencionado. También sirve para compartir Internet.

1.5.5 Servicio de Configuración Dinámica de Host

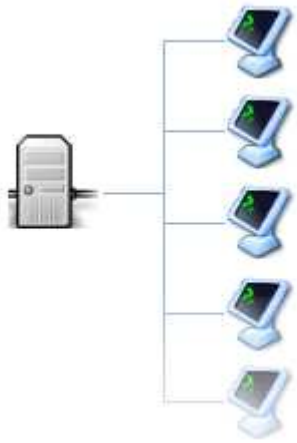

Figura I-12 Servicio de Configuración Dinámica de Host

DHCP (sigla en inglés de Dynamic Host Configuration Protocol - Protocolo Configuración Dinámica de Anfitrión) es un protocolo de red que permite a los nodos de una red IP obtener sus parámetros de configuración automáticamente. Se trata de un protocolo de tipo cliente/servidor en el que generalmente un servidor posee una lista de direcciones IP dinámicas y las va asignando a los clientes conforme éstas van estando libres, sabiendo en todo momento quién ha estado en posesión de esa IP, cuánto tiempo la ha tenido y a quién se la ha asignado después.

1.5.6 Servicio de Transferencia de Archivos

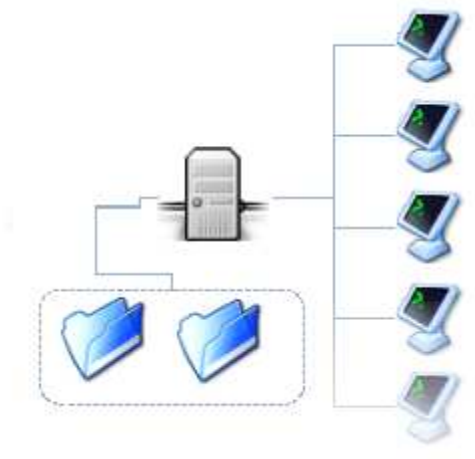

Figura I-13 Servicio de Transferencia de Archivos

En servidor FTP es un servidor que opera sobre el protocolo de transferencia de archivos FTP (File Transfer Protocol por sus siglas en inglés).

Es un protocolo muy común que se ha utilizado durante tanto tiempo como HTTP para transferir archivos en Internet y entre nodos de las redes.

El Servicio FTP es ofrecido por la capa de Aplicación del modelo de capas de red TCP/IP al usuario, utilizando normalmente el puerto de red 20 y el 21.

1.5.7 Servicio de Acceso Remoto

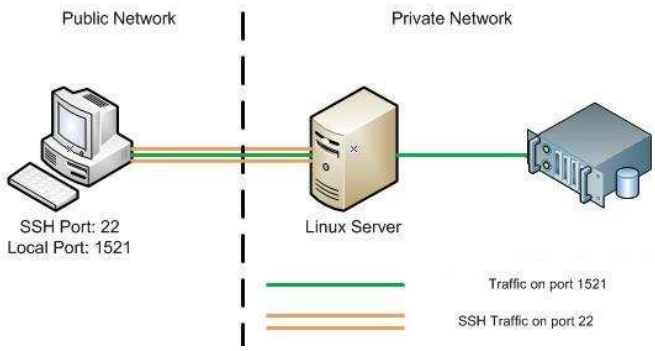

Figura I-14 Servicio de Acceso Remoto

Telnet (TELecommunication NETwork) es el nombre de un protocolo de red (y del programa informático que implementa el cliente), que sirve para acceder mediante una red a otra máquina, para manejarla remotamente como si estuviéramos sentados delante de ella. Para que la conexión funcione, como en todos los servicios de Internet, la máquina a la que se acceda debe tener un programa especial que reciba y gestione las conexiones.

SSH (Secure SHell) es un protocolo similar a Telnet que permite abrir un shell en una máquina remota, con la diferencia de que SSH encripta toda la información que viaja por la red.

1.6 ANALISIS DE HARDWARE Y SOFTWARE DE LOS SERVICIOS

Nombre Servidor	*Modelo Servidor*	*Disco Duro*		*Procesador*				*Plataforma*	*Software y Servicios*
		Espacio Usado GB	*Espacio Total*	*CPU's*	*Tipo*	*Velocidad Ghz*	*RAM GB*		
WebMail	HP ML340 G5	80	438	2	Quad Core	3.0	4	Linux Centos 5, Apache 2, Mysql 5, Php 5	Servidor mail, Postfix, Squirrelmail, Courier IMAP, Spamassain, Clamv
Proxy	IBM XSERIES 235	21	72	2	Xeon	3.06	1	Linux Centos 5, Apache 2, Mysql 5, Php 5	Servidor de aplicaciones, Dns interno, DHCP, Squid Proxy
WebServer	IBM XSERIES 235	20	72	2	Xeon	3.06	1	Linux Centos 5, Apache 2, Mysql 5, Php 5	Servidor de aplicaciones
Dns Externo		6	36	2	Pentium	2.4	0.512	Linux Centos 5,	Servidor DNS

Tabla I-2 Análisis de Hardware y Software

La información mostrada se obtuvo del Departamento de Sistemas y Telemática el mismo que nos facilitara la implementación con el nuevo protocolo IPv6.

1.7 SELECCION DE LOS SERVICIOS EN LOS QUE SE IMPLEMENTARA IPV6

Los servicios a implementarse en la Intranet de la ESPOCH, se implementaran de acuerdo los siguientes aspectos:

- ✓ Poseemos direccionamiento IPv6 de ISP.
- ✓ Acceder a segmento de Red IPv6
- ✓ Dependencia entre servicios.
- ✓ Si los servicios dejan de funcionar, ¿En qué grado afecta a la red de la Espoch?
- ✓ ¿Con qué frecuencia son utilizados?
- ✓ Internet Avanzado utiliza IPv6.
- ✓ ¿Funcionan en servidores independientes?
- ✓ ¿Es un servicio necesario para acceder a IPv6?

Estos parámetros son ponderados de la siguiente manera:

3 Alta

2 Media

1 Baja

Servicio	*Críticos*	*Frecuencia de Uso*	*Internet Avanzado*	*Servidor independiente*	*Requerido para acceder a IPv6*	*TOTAL*
Mail	2	3	3	3	1	12
Web	3	3	3	2	3	13
Ftp	2	1	1	2	1	7
Dns	3	3	3	2	3	14
Proxy	3	3	1	2	2	11
Dhcp	3	3	1	2	2	11

Tabla I-3 Selección de los Servicios a Implementar con IPV6

Los servicios a implementar con el análisis anterior son los siguientes:

- ✓ **Dns:** Servicio para resolución de nombres.
- ✓ **Web:** Servicio para Web Hosting.
- ✓ **Mail:** Servicio de Correo Electrónico.
- ✓ **Proxy:** Servicio para compartir internet y filtrado.
- ✓ **Dhcp;** Servicio de Configuración dinámica de direcciones.
- ✓ **Ftp:** Servicio de transferencia de archivos.

1.8 ESTADO ACTUAL DE LOS SERVICIOS A IMPLEMENTAR EN DOBLE - PILA

1.8.1 Servicio de Resolución de Nombres

Datos Generales

Dirección IP	172.30.60.5
Hostname	proxy.espoch.edu.ec
S.O.	Linux CentOS 5.0
Kernel	2.6.18-92.1.22.el5

Tabla I-4 Descripción del Servidor de Resolución de Nombres

Bind

Incluye el Servidor DNS (named) y herramientas para verificar su funcionamiento.

Version: 9.3.4

Release: 6.0.2.P1.el5_2

bind-libs

Biblioteca compartida que consiste en rutinas para aplicaciones para utilizarse cuando se interactúe con Servidores DNS.

Version: 9.3.4

Release: 6.0.2.P1.el5_2

bind-chroot

Contiene un árbol de ficheros que puede ser utilizado como una jaula chroot para named añadiendo seguridad adicional al servicio.

Version: 9.3.4

Release: 6.0.2.P1.el5_2

bind-utils

Colección de herramientas para consultar Servidores DNS.

Version: 9.3.4

Release: 6.0.2.P1.el5_2

1.8.2 Servicio de Hosting

Datos Generales

Dirección IP	172.30.60.5
Hostname	proxy.espoch.edu.ec
S.O.	Linux CentOS 5.0
Kernel	2.6.18-92.1.22.el5

Tabla I-5 Descripción del Servidor de Hosting

Apache

Apache es el servicio que se encarga de resolver las peticiones de páginas de Internet de los clientes utilizando el protocolo de Internet http.

Version: 2.2.3

Release: 11.el5_2.centos.4

1.8.3 Servicio de Correo Electrónico

Datos Generales

Dirección IP	172.30.60.10
Hostname	mail.espoch.edu.ec
S.O.	Linux CentOS 5.2
Kernel	2.6.18-92.1.22.el5

Tabla I-6 Descripción del Servidor de Correo Electrónico

Mysql

MySQL™ es un DBMS (bataBase Management System) o sistema de gestión de base de datos SQL (Structured Query Language o Lenguaje Estructurado de Consulta).

Version 5.0.45

Apache

Apache es el servicio que se encarga de resolver las peticiones de páginas de Internet de los clientes utilizando el protocolo de Internet http.

Version 2.2.3

Php

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde

una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools).

Version: 5.1.6

Release: 20.el5_2.1

Postfix

El paquete Postfix contiene un Agente de Transporte de Correo (MTA). Es útil para enviar correo a otros usuarios de tu máquina. También puede configurarse como servidor de correo central para tu dominio, agente de reenvío de correo o, simplemente, como agente de entrega de correo a tu Proveedor de Servicios de Internet (ISP) local.

Version 2.5.3

Courier-imap

Cyrus IMAP (Internet Message Access Protocol). A diferencia de otros servidores IMAP, Cyrus usa su propio método para almacenar el correo de los usuarios. Cada mensaje es almacenado en su propio fichero. El beneficio de usar ficheros separados es una mayor fiabilidad ya que sólo un mensaje se pierde en caso de error del sistema de ficheros. Los metadatos, tales como el estado de un mensaje (leído, etc.) se almacenan en una base de datos. Además, los mensajes son indexados para mejorar el rendimiento de Cyrus, especialmente con muchos usuarios e ingentes cantidades de mensajes. No hay nada tan rápido como el servidor IMAP Cyrus.

Todos los usuarios son autenticados por el servidor IMAP. Esto lo convierte en una magnífica solución cuando se tiene una gran cantidad de usuarios.

La administración es llevada a cabo mediante comandos especiales de IMAP. Esto le permite usar tanto la interfaz de línea de comandos como los interfaces web. Este método es mucho más seguro que un interfaz web para /etc/passwd. Desde la versión 2.1 de Cyrus, se usa la versión 2 de la librería SASL para la autenticación. En la configuración descrita en este artículo se implementa una autenticación de tres capas. Cyrus se autentica con saslauth daemon, quien redirige la petición al mecanismo que le hayamos definido, por ejemplo sasldb, que buscará la información del usuario en una base de datos Berkeley DB.

Version: 4.1.3

Release: 1

Cyrus-sasl

SASL son las siglas de Simple Authentication and Security Layer, un método para añadir soporte para la autenticación a protocolos basados en la conexión que ha sido estandarizado por la IETF (Internet Engineering Task Force). Se usa en servidores (en este caso Cyrus IMAP) para manejar las peticiones de autenticación de los clientes. Para ello, el protocolo incluye un comando para identificar y autenticar un usuario contra un servidor y para, opcionalmente, negociar la protección de las subsiguientes interacciones del protocolo. Si se negocia su uso, una capa de seguridad es añadida entre el protocolo y la conexión.

La librería SASL de Cyrus también usa la librería OpenSSL para cifrar los datos.

Version: 2.1.22

Release: 4

Clamav

ClamAV es una herramienta antivirus GPL para UNIX. El propósito principal de este software es la integración con los servidores de correo (escaneo de datos adjuntos). El paquete proporciona un servicio multihilo flexible y escalable, un analizador de línea de comandos y una utilidad para la actualización automática vía Internet. Los programas están basados en una librería distribuida con el paquete Clam AntiVirus, la cual puede ser usada por su propio software. Y lo más importante, la base de datos se mantiene actualizada constantemente.

Otras características destacables son el soporte de firmas digitales en la actualización de la base de datos, el análisis durante el acceso bajo Linux y FreeBSD, la detección de más de 20000 virus, gusanos y troyanos, el soporte integrado para archivos comprimidos con Rar, Zip, Gzip y Bzip2 y formatos de correo Mbox, Maildir y ficheros crudos de correo.

Version: 0.93.3

Release: 1.11.el5.al

SpamAssassin

SpamAssassin es un filtro de correo que trata de identificar el spam mediante el análisis del texto y el uso en tiempo real de algunas listas negras a través de Internet.

A partir de su base de datos de reglas, utiliza un amplio abanico de pruebas heurísticas en las cabeceras y el cuerpo de los correos para identificar el spam,

también conocido como correo electrónico comercial no solicitado. Una vez identificado, el correo puede ser opcionalmente marcado como spam o más tarde filtrado usando el cliente de correo del usuario.

SpamAssassin normalmente identifica acertadamente entre un 95 y un 99% del spam, dependiendo del tipo de correo que se reciba. También incluye soporte para informar de mensajes de spam, automática o manualmente, a bases de datos como Vipul's Razor.

Version 3.2.5.

1.8.4 Servicio de Proxy

Datos Generales

Dirección IP	172.30.60.5
Hostname	proxy.espoch.edu.ec
S.O.	Linux CentOS 5.2
Kernel	2.6.18-92.1.22.el5

Tabla I-7 Descripción del Servidor de Proxy

Squid

Cuando navegamos a través de un proxy (Squid), cada petición que hace nuestro navegador se delega al servidor proxy y es este el que se descarga la pagina o el elemento web que se ha solicitado y se lo pasa a nuestro navegador. Por tanto es un intermediario entre los usuarios y la web. Al estar en medio de ese tráfico puede realizar dos funciones muy importantes: controlar los accesos (permitir o denegar

según se disponga en sus normas); y además hacer cacheo de elementos (páginas web, imágenes, iconos que una vez se piden se guardan en la memoria del proxy), con lo que si se solicita un elemento que ya se ha pedido en lugar de volver a bajárselo de internet, lo sirve el propio proxy.

Con esta técnica de cache nos podemos ahorrar desde un 20 a un 40% de tráfico de internet.

Entre otras cosas, Squid puede funcionar como Servidor Intermediario (Proxy) y caché de contenido de Red para los protocolos HTTP, FTP, GOPHER y WAIS, Proxy de SSL, caché transparente, WWCP, aceleración HTTP, caché de consultas DNS y otras muchas más como filtración de contenido y control de acceso por IP y por usuario.

Version: 2.6.STABLE6

Release: 5.el5_1.3

1.8.5 Servicio de Configuración Dinámica de Host

Datos Generales

Dirección IP	172.30.60.5
Hostname	proxy.espoch.edu.ec
S.O.	Linux CentOS 5.2
Kernel	2.6.18-92.1.22.el5

Tabla I-8 Descripción del Servidor de Configuración Dinámica de Host

DHCP

DHCP (Dynamic Host Configuration Protocol) es un protocolo de red para asignar automáticamente información TCP/IP a equipos cliente.

Cada cliente DHCP se conecta a un servidor DHCP centralizado que devuelve la configuración de red del cliente (incluyendo la dirección IP, la puerta de enlace y los servidores DNS).

Version: 3.0.5

Release: 13.el5

1.8.6 Servicio de Transferencia de Archivos

Datos Generales

Dirección IP	172.30.60.5
Hostname	proxy.espoch.edu.ec
S.O.	Linux CentOS 5.2
Kernel	2.6.18-92.1.22.el5

Tabla I-9 Descripción del Servidor de Transferencia de Archivos

FTP

El Protocolo de transferencia de archivos (FTP) es uno de los protocolos más viejos y populares que se encuentran en la Internet hoy día.

Su objetivo es el de transmitir archivos exitosamente entre máquinas en una red sin que el usuario tenga que iniciar una sesión en el host remoto o que requiera tener conocimientos sobre cómo utilizar el sistema remoto.

FTP permite a los usuarios acceder a archivos en sistemas remotos usando un conjunto de comandos estándar muy simples.

Version: 2.0.5

Release: 12.el5

1.9 TOPOLOGIA DE LA RED ACTUAL

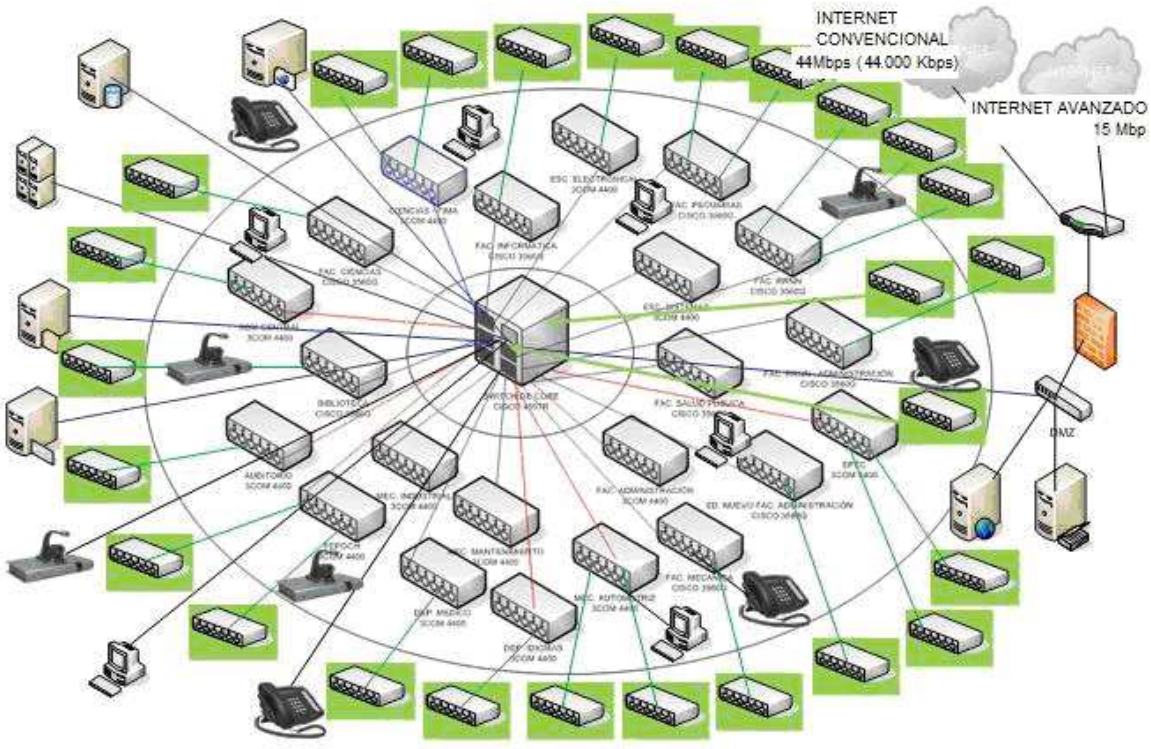

Figura I-15 Topología Lógica de la Intranet

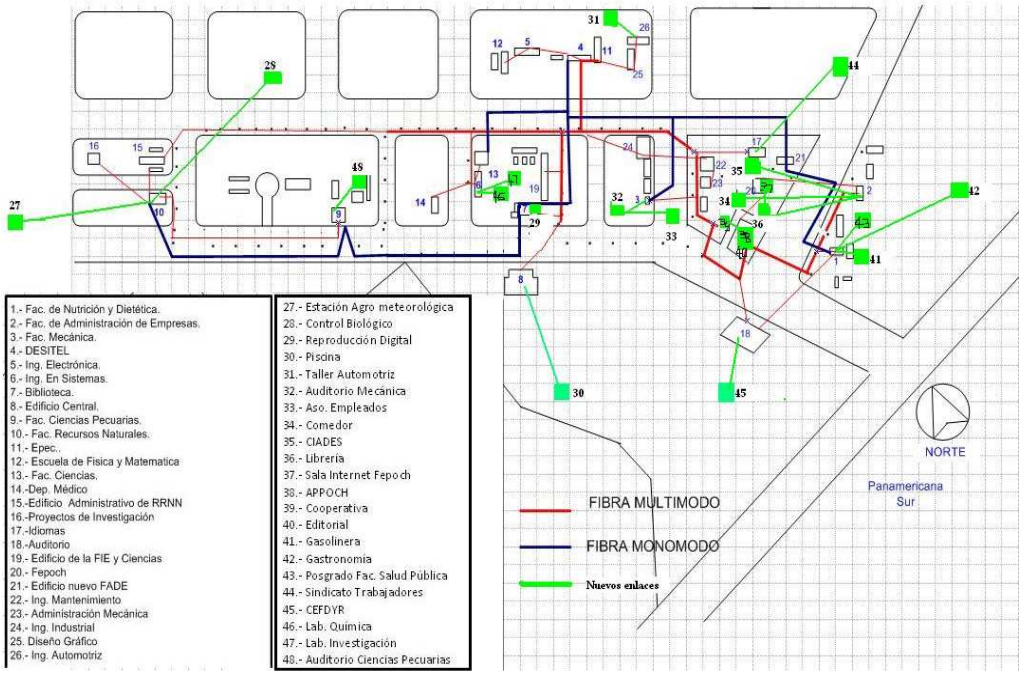

Figura I-16 Topología Física de la Intranet

1.10 ANALISIS COMPARATIVO IPv4/IPv6 DE LOS SERVICIOS A IMPLEMENTAR, BASADO EN LOS RFCs.

1.10.1 Sistema de Nombres de Dominio

En el Protocolo de Internet Versión 4:

- ✓ Esta versión es una versión de 32bits y consta de cuatro grupos binarios de 8bits cada uno (8x4=32), cada grupo de 8bits se llama octeto, o lo que es lo mismo, cuatro grupos decimales, formado cada uno por tres dígitos. el valor decimal de cada octeto puede ser entre 0 y 255.
- ✓ Utiliza registros de recurso (A) de dirección de host en el Sistema de nombres de dominio (DNS) para correlacionar nombres de host con direcciones IPv4.

- ✓ Utiliza registros de recurso (A) de puntero en el dominio DNS IN-ADDR.ARPA para correlacionar direcciones IPv4 con nombres de host.

En el Protocolo de Internet Versión 6:

- ✓ La configuración de estas direcciones es bastante más estructurada que la actual, ya que se trata de una serie de 8 grupos de 16bits (de 0 a ffff), separados por : en los que el valor 0 se puede sustituir por :: esta dirección consta de 128 bits, su representación esta en hexadecimal
- ✓ Utiliza registros de recurso (AAAA) de dirección de host en el Sistema de nombres de dominio (DNS) para correlacionar nombres de host con direcciones IPv6.
- ✓ Utiliza registros de recurso (PTR) de puntero en el dominio DNS IP6.INT para correlacionar direcciones IPv6 con nombres de host.

1.10.2 Protocolo de Transferencia de Hipertexto

Para brindar el servicio HTTP, utilizaremos el servidor Apache, el mismo que viene nativo el soporte para IPv6, es decir que puede implementarse tanto con IPv4 e IPv6.

En el Protocolo de Internet Versión 4:

- ✓ Direcciones de 32 bits.
- ✓ Para poner a escuchar Apache en un interfaz IPv4 seria Listen :80
- ✓ Para poner a escuchar Apache en un interfaz IPv4 con dirección IPv4 seria Listen 172.30:60.26:80

En el Protocolo de Internet Versión 6:

- ✓ Direcciones de 128 bits.
- ✓ Las direcciones IPv6 deben escribirse entre corchetes.

- ✓ Para poner a escuchar Apache en un interfaz IPv6 seria Listen [::]:80
- ✓ Para poner a escuchar Apache en un interfaz IPv6 con dirección IPv4 seria Listen [2800:68:a:60::26]:80

1.10.3 Protocolo de Transferencia de Ficheros

En el Protocolo de Internet Versión 4:

El Modelo FTP

El intérprete de protocolo de usuario (user-PI), inicia la conexión de control, ésta sigue el Protocolo Telnet; las órdenes FTP estándar las genera el user-PI y se transmiten al proceso servidor a través de la conexión de control.

Funciones de Transferencia de Datos

Existen dos tipos de conexión: datos y control; los ficheros se transmiten por la conexión de datos y la conexión de control se usa para enviar órdenes, que describen la función a realizar y las repuestas a estas órdenes.

Las órdenes incluyen:

- ✓ MODE (modo) especifica cómo se transmiten los bits de datos.
- ✓ STRU (estructura).
- ✓ TYPE, definen la forma de representación de los datos.

Entre los tipos de datos que permite transmitir el FTP tenemos:

- ✓ Tipo ASCII: tipo por defecto, debe ser admitido por las implementaciones del FTP, su principal propósito es transferir ficheros de texto.

- ✓ Tipo EBCDIC: transferencia eficaz entre ordenadores que usan EBCDIC como su representación de carácter interno.
- ✓ Tipo Imagen: los datos se envían como bits contiguos, están empaquetados en bytes de transferencia de 8 bits.
- ✓ TIPO Local: los datos se transfieren en bytes lógicos del tamaño especificado por el segundo parámetro obligatorio, tamaño de byte.

Ordenes de Parámetros de Transferencia

- ✓ Los parámetros de transferencia de datos tienen valores por defecto y las órdenes que especifican valores para ellos, sólo se deben utilizar si se van a cambiar los parámetros por defecto; esto implica que el servidor debe "recordar" los valores por defecto aplicables. Entre las principales órdenes por defecto:
- ✓ Puerto de Datos (Port): es una especificación ordenador-puerto, para el puerto que será usado en la conexión de datos; existen valores por defecto tanto para el puerto de usuario y del servidor.
- ✓ PASIVO (PASV): esta orden solicita al server-DTP que "escuche" en un puerto de datos (que no es el puerto por defecto) y espere a recibir una conexión en lugar de iniciar una al recibir una orden de transferencia. La respuesta a este comando incluye la dirección IP

En el Protocolo de Internet Versión 6:

- ✓ EPRT: permite la especificación de una dirección extendida para la conexión de datos, la dirección extendida consiste de un protocolo de red, es decir, la dirección de red y de transporte. El formato es el siguiente:

EPRT<space><d><net-prt><d><net-addr><d><tcp-port><d>

De donde:

<space>: espacio

<d>: caracter delimitador ASCII 33-126

<net-prt>: una dirección IP definida de IPv4 e IPv6

<net-addr>: es un protocolo específico de una representación string de una dirección de red; en el siguiente formato:

- ✓ Puntos decimales. Por ejemplo: 132.235.1.2
- ✓ Cadena String. Por ejemplo: 1080::8:800:200C:417A

<tcp-port>: el número de puerto en el cual el host escucha para la conexión.

Ejemplo del Comando EPRT

- ✓ EPRT |1|132.235.1.2|6275|
- ✓ EPRT |2|1080::8:800:200C:417A|5282|

Algunas de las respuestas que puede dar el servidor son:

Código	**Significado**
200	Comando ok
500 and 501	Errores de Sintaxis
522	No soporta solicitud del protocolo

De forma general los errores se los clasifica de la siguiente manera:

- ✓ 5yz Respuesta Negativa
- ✓ x2z Conexiones

- ✓ xy2 Falla de Puerto extendido

EPSV: este comando solicita al servidor que escuche un puerto y espere por una conexión y la respuesta es el número de puerto para escuchar la conexión.

Las respuestas se las clasifica:

- ✓ 2yz Positive Completion
- ✓ x2z Conexiones
- ✓ xy9 Extended Passive Mode Entered

El formato de la respuesta es:

<texto indicando que el servidor es entering extended passive mode> /(<d><d><d><tcp-port><d>)

Los primeros dos campos deberían estar en blanco, el siguiente es el número de puerto tcp en el cual el servidor escuchará.

El comando EPSV puede ser usado con el argumento “ALL” para informar que los Traductores de Direcciones de Red que el servidor rechazará todas las conexiones que no sean de EPSV por ejemplo: EPRT, PORT PASV etc.

Para todas las transferencias FTP entre dos máquinas se recomienda utilizar el comando EPSV porque mejora el rendimiento; utilizar el comando EPSV no requiere NATs para cambiar direcciones de red en el tráfico que se envía, el NAT podría cambiar si se utilizara EPRT.

1.10.4 Protocolo Simple de Transferencia de Correos

En el Protocolo de Internet Versión 4:

El objetivo, es transferir un correo de forma confiable y eficiente; se basa en el modelo cliente-servidor, donde un cliente envía un mensaje a uno o varios receptores.

El funcionamiento es el siguiente:

- ✓ Un cliente establece una conexión con el servidor SMTP, espera que envíe un mensaje “220 Service ready” o “421 Service non available”.
- ✓ El cliente envía un HELO, con esto el servidor se identifica.
- ✓ Cliente comienza la transacción con la orden MAIL, como argumento se puede pasar la dirección de correo al que el servidor notificará cualquier fallo; el servidor responde “250 OK”.
- ✓ Hay que comunicar a quien se envía el correo, esto se hace con RCPT TO:<destino@host>;el servidor contestará “250 OK” o bien “550 No such user here”, si no encuentra al destinatario.
- ✓ El cliente envía una orden DATA para indicar que se envía el contenido del mensaje; el servidor responde “354 Start mail input, end with <CLRF>.<CLRF>”.
- ✓ El cliente envía el cuerpo del mensaje, línea a línea; una vez finalizado, se termina con un <CLRF>.<CLRF>, el servidor contestará “250 OK”, o un mensaje de error apropiado.
- ✓ Tras el envío, el cliente, con la orden QUIT corta la conexión. Las respuestas que da el servidor pueden ser:

  2XX, afirmativa

  3XX, temporal afirmativa

  4XX, error, se espera a que se repita la instrucción

5XX, error

En el Protocolo de Internet Versión 6:

Hay que ser cuidadosos en la interoperabilidad de IPv4–IPv6, el primer aspecto a considerar son los registros Mail Exchange (MX) necesarios para la operación Doble Pila IPv4-IPv6. Los mensajes de mail se entregan basándose en el Sistema de Resolución de Nombres, los Registros de Recurso MX (RR MX) son buscados en el DNS donde se establece los nombres de hosts en los que corren los Agentes de Transferencia de Correo (MTAs) asociados con la parte del dominio de la dirección de mail, para la búsqueda el DNS utiliza registros IN tanto para IPv4 e IPv6, registros similares son usados para enrutar el mail en IPv4/IPv6, se conocen como IN MX.

Un RR MX tiene dos parámetros:

Valor de prioridad

Nombre de host destino: utilizado para buscar una dirección IP al iniciar una conexión SMTP. *Ejemplo:*

Como los sitios de IPv4 no cuentan con la interoperabilidad IPv4-IPv6; durante el período de transición es necesario que los dominios tengan registros MX tanto para IPv4 como IPv6, para cada dirección que exista, considerando que existirán entradas solo para IPv4 o IPv6 dependiendo del soporte en Doble Pila.

example.org. IN MX 1 mx1.example.org.

IN MX 10 mx10.example.org.

mx1.example.org. IN A 192.0.2.1; **doble-pila**

IN AAAA 2001:db8:ffff::1

mx10.example.org. IN AAAA 2001:db8:ffff::2 ; **IPv6**

Para un registro MX, hay varios estados posibles:

Uno o más registros para un destino IPv4.

Uno o más registros para un destino IPv6.

Una combinación de registros IPv4-IPv6 (A/AAAA)

Configuración MX

Asegurar la conectividad hacia ambos protocolos: si un sitio está en doble-pila tendrá configurados registros A y AAAA.

Conectividad entre los MX primario y secundario.

1.10.5 Proxy

En el Protocolo de Internet Versión 4:

Proxy de web / Proxy cache de web

Se trata de un proxy para una aplicación específica; el acceso a la web. Aparte de la utilidad general de un proxy, proporciona una caché para las páginas web y los contenidos descargados, que es compartida por todos los equipos de la red, con la consiguiente mejora en los tiempos de acceso para consultas coincidentes.

Al mismo tiempo libera la carga de los enlaces hacia Internet.

Funcionamiento

1. El cliente realiza una petición (Ejemplo: mediante un navegador web) de un recurso de Internet (una página web o cualquier otro archivo) especificado por una URL.
2. Cuando el proxy caché recibe la petición, busca la URL resultante en su caché local. Si la encuentra, contrasta la fecha y hora de la versión de la página demanda con el servidor remoto, si la página no ha cambiado desde que se cargo en caché la devuelve inmediatamente, ahorrándose de esta manera mucho tráfico pues solo intercambia un paquete para comprobar la versión, si la versión es antigua o simplemente no se encuentra en la caché, lo captura del servidor remoto, lo devuelve al que lo pidió y guarda o actualiza una copia en su caché para futuras peticiones.

El caché utiliza normalmente un algoritmo para determinar cuándo un documento está obsoleto y debe ser eliminado de la caché, dependiendo de su antigüedad, tamaño e histórico de acceso.

Dos de esos algoritmos básicos son el LRU (el usado menos recientemente, en inglés "Least Recently Used") y el LFU (el usado menos frecuentemente, "Least Frequently Used"). Los proxies web también pueden filtrar el contenido de las páginas Web servidas. Algunas aplicaciones que intentan bloquear contenido Web ofensivo están implementadas como proxies Web. Otros tipos de proxy cambian el formato de las páginas web para un propósito o una audiencia específicos, para, por ejemplo, mostrar una página en un teléfono móvil o una PDA. Algunos operadores de red también tienen proxies para interceptar virus y otros contenidos hostiles servidos por páginas Web remotas.

Proxies transparentes

Usan los proxies para reforzar las políticas de uso de la red o para proporcionar seguridad y servicios de caché. Normalmente, un proxy Web o NAT no es transparente a la aplicación cliente: debe ser configurada para usar el proxy, manualmente.

Por lo tanto, el usuario puede evadir el proxy cambiando simplemente la configuración. Una ventaja de tal es que se puede usar para redes de empresa.

Un proxy transparente combina un servidor proxy con NAT de manera que las conexiones son enrutadas dentro del proxy sin configuración por parte del cliente, y habitualmente sin que el propio cliente conozca de su existencia.

Este es el tipo de proxy que utilizan los proveedores de servicios de internet (ISP).

Reverse Proxy

Un reverse proxy es un servidor proxy instalado en el domicilio de uno o más servidores web. Todo el tráfico entrante de Internet y con el destino de uno de esos servidores web pasa a través del servidor proxy. Hay varias razones para instalar un "reverse proxy":

- ✓ Seguridad: el servidor proxy es una capa adicional de defensa y por lo tanto protege los servidores web.
- ✓ Cifrado / Aceleración SSL: cuando se crea un sitio web seguro, habitualmente el cifrado SSL no lo hace el mismo servidor web, sino que es realizado por el "reverse proxy", el cual está equipado con un hardware de aceleración SSL (Security Sockets Layer).
- ✓ Distribución de Carga: el "reverse proxy" puede distribuir la carga entre varios servidores web. En ese caso, el "reverse proxy" puede necesitar reescribir las URL de

cada página web (traducción de la URL externa a la URL interna correspondiente, según en qué servidor se encuentre la información solicitada).

- ✓ Caché de contenido estático: Un "reverse proxy" puede descargar los servidores web almacenando contenido estático como imágenes u otro contenido gráfico.

Proxy NAT (Network Address Translation) / Enmascaramiento

Otro mecanismo para hacer de intermediario en una red es el NAT.

La traducción de direcciones de red (NAT, Network Address Translation) también es conocida como enmascaramiento de IPs. Es una técnica mediante la cual las direcciones fuente o destino de los paquetes IP son reescritas, sustituidas por otras (de ahí el "enmascaramiento").

Esto es lo que ocurre cuando varios usuarios comparten una única conexión a Internet. Se dispone de una única dirección IP pública, que tiene que ser compartida. Dentro de la red de área local (LAN) los equipos emplean direcciones IP reservadas para uso privado y será el proxy el encargado de traducir las direcciones privadas a esa única dirección pública para realizar las peticiones, así como de distribuir las páginas recibidas a aquel usuario interno que la solicitó. Estas direcciones privadas se suelen elegir en rangos prohibidos para su uso en Internet como 192.168.x.x, 10.x.x.x, 172.16.x.x y 172.31.x.x

El acceso a Internet mediante NAT proporciona una cierta seguridad, puesto que en realidad no hay conexión directa entre el exterior y la red privada, y así nuestros equipos no están expuestos a ataques directos desde el exterior.

Mediante NAT también se puede permitir un acceso limitado desde el exterior, y hacer que las peticiones que llegan al proxy sean dirigidas a una máquina concreta que haya sido determinada para tal fin en el propio proxy.

La función de NAT reside en los Cortafuegos y resulta muy cómoda porque no necesita de ninguna configuración especial en los equipos de la red privada que pueden acceder a través de él como si fuera un mero encaminador.

Proxy Abierto

Este tipo de proxy que acepta peticiones desde cualquier ordenador, esté o no conectado a su red.

En esta configuración el proxy ejecutará cualquier petición de cualquier ordenador que pueda conectarse a él, realizándola como si fuera una petición del proxy. Por lo que permite que este tipo de proxy se use como pasarela para el envío masivo de correos de spam. Un proxy se usa, normalmente, para almacenar y redirigir servicios como el DNS o la navegación Web, mediante el cacheo de peticiones en el servidor proxy, lo que mejora la velocidad general de los usuarios. Este uso es muy beneficioso, pero al aplicarle una configuración "abierta" a todo internet, se convierte en una herramienta para su uso indebido.

Debido a lo anterior, muchos servidores, como los de IRC, o correo electrónicos, deniegan el acceso a estos proxys a sus servicios, usando normalmente listas negras ("BlackList").

En el Protocolo de Internet Versión 6:

Análisis políticas de seguridad

- ✓ Authentication Header (AH) ofrece integridad en los datos y campos inmutables para todo el paquete a ser transferido; gracias al Traductor de Direcciones de Red (NAT) y al Traductor de Direcciones de Puerto (PAT) existen modificaciones en las direcciones origen y destino permitiéndoles ser consideradas como campos inmutables.
- ✓ Encapsulating Security Payload (ESP) ofrece confidencialidad y protección de integridad para un paquete, encapsulándolo y encriptándolo, pero este proceso no lo realiza en la cabecera del paquete, por tanto al realizar la verificación de integridad de los paquetes, esta no coincide con el AH.
- ✓ Al realizar la encriptación el Traductor de Direcciones de Red y Puerto (NAPT) es incapaz de extraer información necesaria del puerto que se encuentra en la capa superior. Para solucionar esto, IPv6 expande su perímetro global a nodos individuales de manera que lo prefijos globales válidos sean distribuidos en los sistemas autónomos permitiendo que los nodos individuales obtengan una única dirección IPv6 global cambiando el diseño y la organización del Proxy para controlar de mejor manera el flujo de información y la seguridad de la misma.

1.10.6 Protocolo de Configuración Dinámica de Host

En el Protocolo de Internet Versión 4:

DHCPv4–El administrador activa DHCP para cada interfaz. La administración se efectúa por interfaz lógica.

DHCPv4–El servidor DHCP proporciona la máscara de subred de cada dirección. La opción de nombre de host establece el nombre de nodo en todo el sistema.

DHCPv4 requiere una configuración de cliente explícita. Deberá configurar el sistema DHCPv4 para direccionar a voluntad; esto se suele llevar a cabo durante la instalación inicial del sistema

DHCPv4 utiliza la dirección MAC y un ID de cliente opcional para identificar al cliente y así asignarle una dirección. Cada vez que el mismo cliente llega a la red, obtiene la misma dirección, si es posible.

En el Protocolo de Internet Versión 6:

DHCPv6–No es necesaria una configuración explícita. Este protocolo se activa en una interfaz física determinada.

DHCPv6–La máscara de subred la proporcionan los anuncios de encaminador, no el servidor DHCPv6. No existe la opción de nombre de host DHCPv6.

DHCPv6 no requiere una configuración de cliente explícita. Por el contrario, el uso de DHCP es una propiedad de la red, y la señal para utilizarlo se encuentra en los mensajes de anuncio de los encaminadores locales. El cliente DHCP crea y destruye automáticamente las interfaces lógicas según sea necesario.

El mecanismo de DHCPv6 es muy parecido, desde el punto de vista administrativo, a la configuración de direcciones sin estado IPv6 (automática) actual. Para la configuración de direcciones sin estado se activaría un indicador en el encaminador local para indicar que, para un conjunto de prefijos determinado, cada cliente deberá configurar automáticamente una dirección propia utilizando el prefijo anunciado, así como un símbolo o número aleatorio de interfaz local. Para DHCPv6 se requieren los mismos prefijos, pero las

direcciones se obtienen y gestionan mediante un servidor DHCPv6 en lugar de asignarse de forma "aleatoria.”

DHCPv6 utiliza básicamente el mismo esquema, pero hace que el ID de cliente sea obligatorio y le impone una estructura. El ID de cliente de DHCPv6 consta de dos partes: un Identificador único de DHCP (DUID) y un Identificador de identidad de asociación (IAID). El DUID identifica el sistema cliente (no solo una interfaz, como en DHCPv4), y el IAID identifica la interfaz en ese sistema.

CAPITULO II

DESARROLLO

2 DESARROLLO

2.1 DIRECCIONAMIENTO

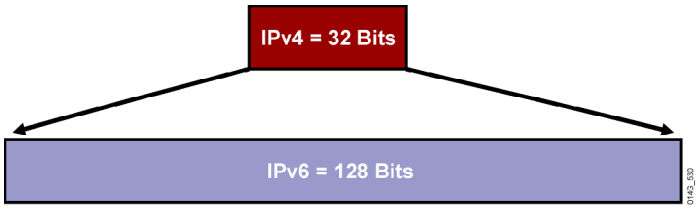

Figura II-17 Direccionamiento IPv6

Las direcciones IPv6 son de 128 bits e identifican interfaces individuales o conjuntos de interfaces. Al igual que en IPv4 en los nodos se asignan a interfaces, y son de 32 bits.

2.1.1 Tipos de direccionamiento Ipv6

Se clasifican en tres tipos:

- ✓ Unicast identifican a una sola interfaz. Un paquete enviado a una dirección unicast es entregado sólo a la interfaz identificada con dicha dirección.
- ✓ Anycast identifican a un conjunto de interfaces. Un paquete enviado a una dirección anycast, será entregado a alguna de las interfaces identificadas con la dirección del conjunto al cual pertenece esa dirección anycast.
- ✓ Multicast identifican un grupo de interfaces. Cuando un paquete es enviado a una dirección multicast es entregado a todos las interfaces del grupo identificadas con esa dirección.

En el IPv6 no existen direcciones broadcast, su funcionalidad ha sido mejorada por las direcciones multicast.

Unicast Global y Anycast

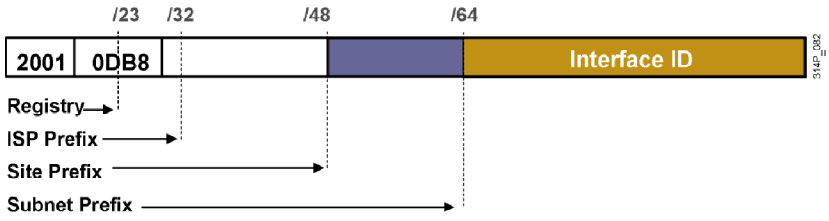

Figura II-18 Dirección Unicast Global y Anycast

- ✓ Utiliza un prefijo de enrutamiento global. Facilita la agregación.
- ✓ Una interface puede tener varias direcciones: unicast, anycast, multicast.
- ✓ Cada interface de tener, por lo menos, una dirección de loopback (::1/128) y una de enlace local (FE80::/10).
- ✓ Opcionalmente, cada interface puede tener múltiples direcciones locales y globales únicas.

- ✓ Las direcciones anycast se utilizan en redes conectadas a múltiples ISP (multihoming) con múltiples conexiones entre sí.

Direcciones Link-Local

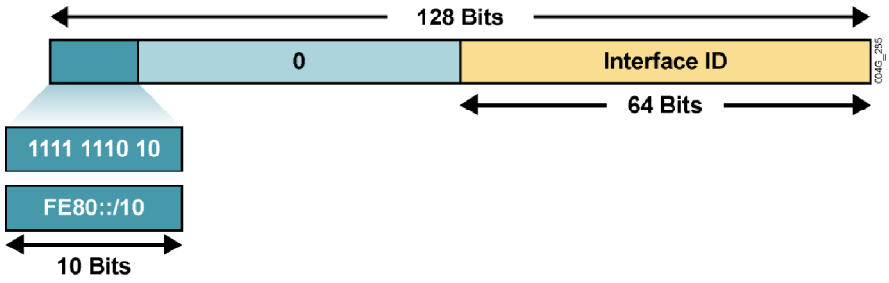

Figura II-19 Dirección Link Local

- ✓ Son creadas dinámicamente en todas las interfaces IPv6 interfaces utilizando el prefijo FE80::/10 y un identificador de interface de 64 bits.
- ✓ Se utilizan para configuración automática de direcciones, descubrimiento de vecinos y descubrimiento de gateway.
- ✓ Conectan dispositivos en la misma red local sin necesidad de direcciones globales. Similar a las direcciones privadas en IPv4.
- ✓ Para la comunicación entre nodos se debe especificar la interface de salida porque todos las interfaces están conectadas a FE80::/10.

2.1.2 Representación de las direcciones

Existen tres formas de representar las direcciones IPv6 como strings de texto.

- ✓ x:x:x:x:x:x:x:x donde cada x es el valor hexadecimal de 16 bits, de cada uno de los 8 campos que definen la dirección. No es necesario escribir los ceros a la izquierda de cada campo, pero al menos debe existir un número en cada campo.

Ejemplos:

FEDC:BA98:7654:3210:FEDC:BA98:7654:3210

1080:0:0:0:8:800:200C:417A

- ✓ Como será común utilizar esquemas de direccionamiento con largas cadenas de bits en cero, existe la posibilidad de usar sintácticamente :: para representarlos. El uso de :: indica uno o más grupos de 16 bits de ceros. Dicho símbolo podrá aparecer una sola vez en cada dirección.

  Por ejemplo:

1080:0:0:0:8:800:200C:417A	unicast address
FF01:0:0:0:0:0:0:101	multicast address
0:0:0:0:0:0:0:1	loopback address
0:0:0:0:0:0:0:0	unspecified addresses

Tabla II-10 Representación de una Dirección IPv6

Podrán ser representadas como:

1080:0:0:0:8:800:200C:417A	unicast address
FF01:: 101	multicast address
::1	loopback address
::	unspecified addresses

Tabla II-11 Representación de una Dirección IPv6 simplificada

- ✓ Para escenarios con nodos IPv4 e IPv6 es posible utilizar la siguiente sintaxis: x:x:x:x:x:x:d.d.d.d, donde x representan valores hexadecimales de las seis partes más significativas (de 16 bits cada una) que componen la dirección y las d, son valores decimales de los 4 partes menos significativas (de 8 bits cada una), de la representación estándar del formato de direcciones IPv4.

  Ejemplos:

  0:0:0:0:0:0:13.1.68.3

  0:0:0:0:0:FFFF:129.144.52.38

  O en la forma comprimida

  ::13.1.68.3

  ::FFFF:129.144.52.38

2.1.3 Representación de los prefijos de las direcciones

Los prefijos de identificadores de subredes, routers y rangos de direcciones IPv6 son expresados de la misma forma que en la notación CIDR utilizada en IPv4. Un prefijo de dirección IPv6 se representa con la siguiente notación:

direccion-ipv6/longitud-prefijo, donde

direccion-ipv6: es una dirección IPv6 en cualquiera de las notaciones mencionadas anteriormente.

longitud-prefijo: es un valor decimal que especifica cuantos de los bits más significativos, representan el prefijo de la dirección.

2.1.4 LAS PRINCIPALES DIFERENCIAS ENTRE IPV4 Y IPV6 SE RESUMEN A CONTINUACION

DESCRIPCION	IPV4	IPV6
Direcciones de origen y destino	32 bits de longitud	128 bits de longitud
IPSec el apoyo:	Opcional	Requerido.
La configuración de direcciones IP:	manualmente o mediante DHCP	Vía Dirección automática - DHCP ya no es necesario, ni es la configuración manual.
Flujo de paquetes de identificación para el manejo de QoS en el encabezado:	No de identificación de flujo de paquetes	El flujo de paquetes de identificación para el manejo QoS existe flujo a través de la Etiqueta del campo
Emisión direcciones	Difunden direcciones se utilizan para transmitir tráfico a todos los nodos en una subred.	Difunden direcciones se sustituirán por un enlace local alcance todos los nodos de las direcciones multicast.
Fragmentación	Interpretada por el envío y de acogida en los enrutadores.	Interpretada por el envío de acogida

Re ensamblaje	Tiene que ser capaz de volver a un paquete de 576-byte	Tiene que ser capaz de volver a un paquete de 1.500 bytes.
Solicitud ARP cuadros	Usado por ARP para resolver una dirección IPv4 a una capa de enlace la dirección	Reemplazo con Vecino Solicitaciones mensajes.
ICMP Router Discovery	Se utiliza para determinar la dirección IPv4 de la mejor puerta de enlace predeterminada.	Reemplazo con ICMPv6 Router Router Advertisement Licitación y mensajes
ICMP Router Discovery	Se utiliza para administrar el grupo subred local.	Reemplazo con Multicast Listener Discovery (MLD) mensajes
Encabezado checksum	Incluido	Exclusión

Tabla II-12 Diferencias entre IPv4 e IPv6

2.1.5 FORMATO DE LA CABECERA DE IPV6

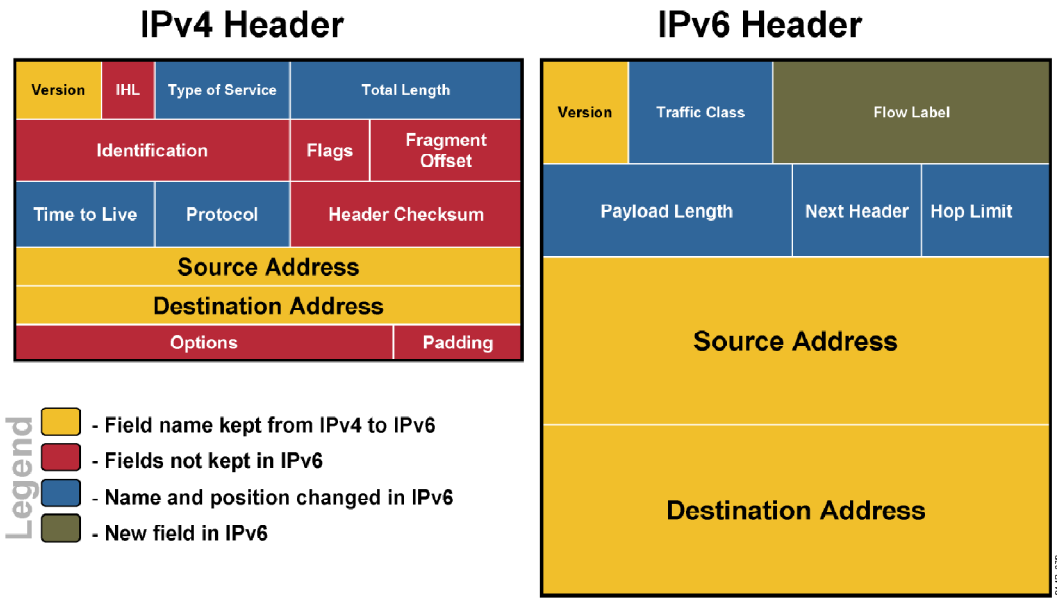

Figura II-20 Cabecera de IPv4 e IPv6

Campo	Tamaño	Descripción
Version	4 bits	«0110» indica versión 6.
Traffic Class	8 bits	Se usa para identificar la «clase» de tráfico, o la prioridad, de forma que los paquetes se puedan reenviar con distintas prioridades para asegurar la QoS. Se utiliza para distinguir las fuentes que deben beneficiarse del control de flujo de otras. Se asignan prioridades de 0 a 7 a fuentes que pueden disminuir su velocidad en caso de congestión. Se asignan valores de 8 a 15 al tráfico en tiempo real (datos de

		audio y video incluidos) en donde la velocidad es constante.
Flow Label	20 bits	Los paquetes que pertenecen a un flujo de clase de tráfico concreto se etiquetan para identificar a qué «flujo» pertenecen contiene un número único escogido por la fuente que intenta facilitar el trabajo de los routers y permitir la implementación de funciones de calidad de servicio como RSVP (Resource reSerVation setup Protocol [Protocolo de reserva de recursos]). Este indicador puede considerarse como un marcador de un contexto en el router. El router puede entonces llevar a cabo procesamientos particulares: escoger una ruta, procesar información en "tiempo real", etc.
Payload Length	16 bits	Tamaño, en bytes, del resto del paquete, incluyendo las cabeceras de extensión.
Next Header	8 bits	Identifica el tipo de cabecera que sigue inmediatamente a la cabecera de IPv6. Usa los mismos valores que en el campo Protocolo de IPv4 (RFC 1700). Puede ser un protocolo (de una capa superior ICMP, UDS, TCP, etc.) o una extensión

Hop Limit	8 bits	Número de enlaces que puede atravesar un paquete antes de descartarlo. Cada vez que se reenvía este campo se decrementa en 1. Reemplaza el campo "TTL" (Time-to-Live [Tiempo de vida]) en IPv4, su valor disminuye con cada nodo que reenvía el paquete. Si este valor llega a 0 cuando el paquete IPv6 pasa por un router, se rechazará y se enviará un mensaje de error ICMPv6. Esto se utiliza para evitar que los datagramas circulen indefinidamente. Tiene la misma función que el campo Time to live (Tiempo de vida) en IPv4, es decir, contiene un valor que representa la cantidad de saltos y que disminuye con cada paso por un router. En teoría, en IPv4, hay una noción del tiempo en segundos, pero ningún router la utiliza. Por lo tanto, se ha cambiado el nombre para que refleje su verdadero uso.
Source Address	128 bits	Dirección del nodo emisor.
Destination Address	128 bits	Dirección del nodo de destino, que puede ser un nodo final o un nodo intermedio.

Tabla II-13 Descripción de la Cabeza IPv6

2.1.6 Pruebas

La Escuela Superior Politécnica de Chimborazo, forma parte del Consorcio Ecuatoriano para el Desarrollo de Internet Avanzado (CEDIA), se le asignó el prefijo IPv6:

2800:0068:000a::/48

Para optimizar el direccionamiento se subneteo a con barra /64, por lo tanto el direccionamiento será desde 2800:0068:000a:0001::/64 hasta 2800:0068:000a:FFFF::/64

2.1.7 Implementación

IMPLEMENTACION DE DUAL STACK (DOBLE PILA IPV4/IPV6)			
NOMBRE DE LA VLAN	**NUMERO DE LA VLAN**	**DIRECCION IPV4**	**DIRECCION IPV6**
		IP/MASCARA	**IP/MASCARA**
Autoridades	2	172.30.10.0 /24	2800:68:a:10::/64
Secretarias	3	172.30.20.0/24	2800:68:a:20::/64
Wireless	4	172.30.30.0/24	2800:68:a:30::/64
Sin Internet	5	172.30.50.0/24	2800:68:a:50::/64
Servicios	6	172.30.60.0/24	2800:68:a:60::/64
Financiero	7	172.30.70.0/24	2800:68:a:70::/64
Telefonía	9	172.30.90.0/24	2800:68:a:90::/64

FACULTADES			
Salud Publica	10	172.30.100.0/24	2800:68:a:100::/64
Administración	11	172.30.101.0/24	2800:68:a:101::/64
Mecánica	12	172.30.102.0/24	2800:68:a:102::/64
Ciencias	13	172.30.103.0/24	2800:68:a:103::/64
Ciencias Pecuarias	16	172.30.106.0/24	2800:68:a:106::/64
Recursos Naturales	17	172.30.107.0/24	2800:68:a:107::/64
Física y Matemática	18	172.30.108.0/24	2800:68:a:108::/64
ESCUELAS			
Ing. en Sistemas	4	172.30. 40.0/24	2800:68:a:40::/64
Ing. en Sistemas	14	172.30.104.0/24	2800:68:a:104::/64
Ing. Electrónica	15	172.30.105.0/24	2800:68:a:105::/64
Diseño Grafico	19	172.30.109.0/24	2800:68:a:109::/64
DEPENDENCIAS			
Gremios	20	172.30.120.0/24	2800:68:a:120::/64
Video Vigilancia	22	172.30.122.0/24	2800:68:a:122::/64
Laboratorio Desitel	24	172.30.124.0/24	2800:68:a:124::/64
Laboratorio Desitel	25	172.30.125.0/24	2800:68:a:125::/64

Tabla II-14 Direccionamiento para la Intranet

2.2 TOPOLOGIA PROPUESTA PARA EL LABORATORIO DE PRUEBAS

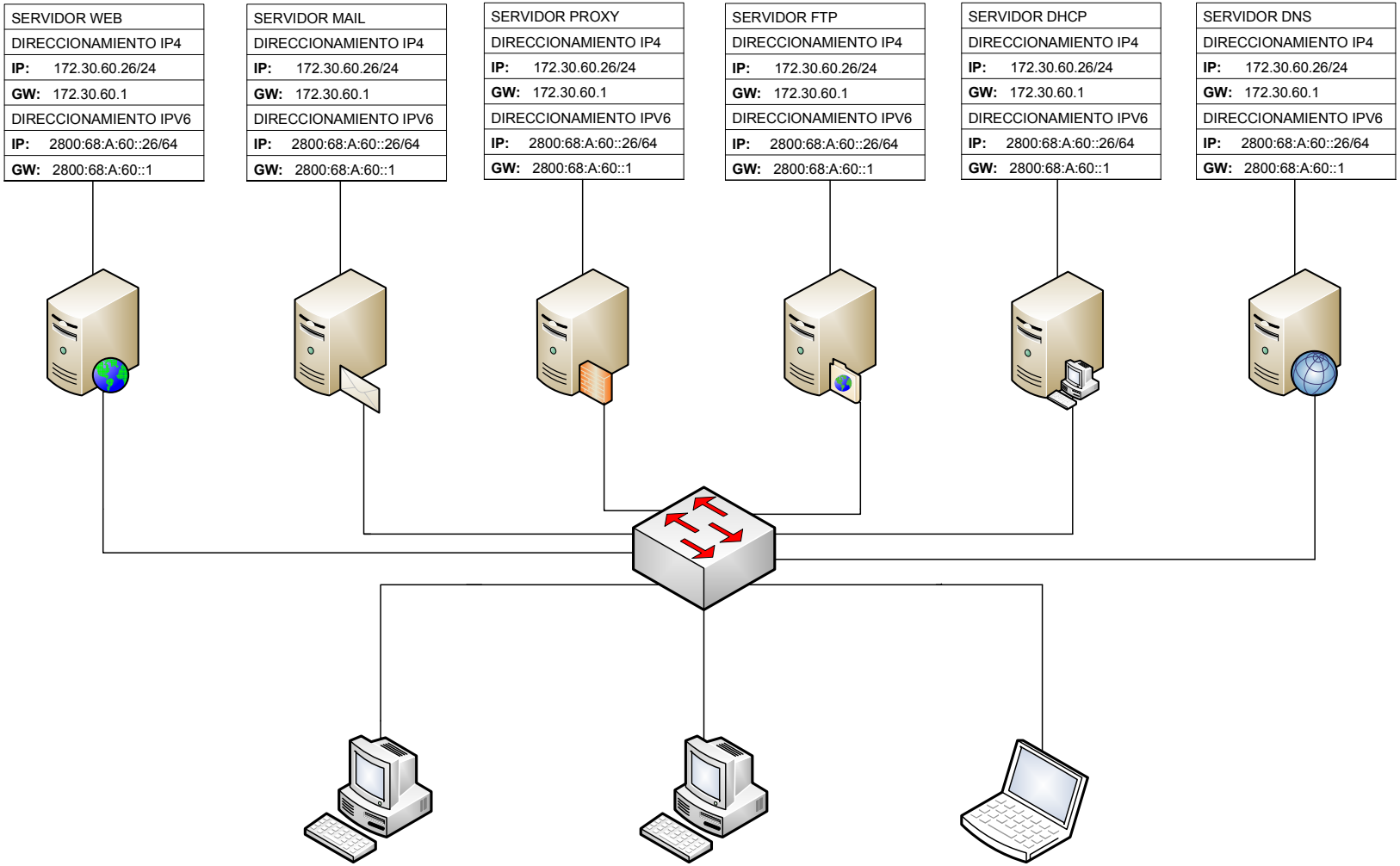

Figura II-21 Topología Propuesta para el laboratorio de pruebas

2.3 SERVIDOR DE SISTEMA DE NOMBRES DE DOMINIO (DNS)

2.3.1 Activar IPv6

El archivo /etc/sysconfig/network es usado para especificar información sobre la configuración de red deseada.

NETWORKING=<valor>

donde <valor> es uno de los siguientes valores booleanos:

yes — Se debería configurar el servicio de red.

NETWORKING=IPV6<valor>

donde <valor> es uno de los siguientes valores booleanos:

yes — Habilita ipv6.

No — No habilita ipv6.

HOSTNAME=<valor>

donde <valor> debería ser el Fully Qualified Domain Name (FQDN), nombre de dominio cualificado completo, tal como hostname.expample.com, pero puede ser cualquier nombre de host necesario.

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=serverlan.espoch.edu.ec
```

Figura II-22 /etc/sysconfig/network

Los archivos de configuración de interfaz controlan las interfaces de software para dispositivos de red individuales. Cuando su sistema arranca, utiliza estos archivos para saber qué interfaces debe activar y cómo deben ser configuradas. Estos archivos habitualmente se conocen como ifcfg-*<nombre>*, donde *<nombre>* hace referencia al nombre del dispositivo que controla el archivo de configuración, cada interfaz de red configurada, tal como ifcfg-eth0 para la interfaz de red Ethernet eth0.

DEVICE=<nombre>

donde <nombre> es el nombre del dispositivo físico (a excepción de los dispositivos PPP asignados de forma dinámica en donde éste es el nombre lógico).

BOOTPROTO=<protocolo>

donde <protocolo> es uno de los siguientes:

none — No se debería utilizar ningún protocolo de tiempo de arranque.

Bootp — Se debería utilizar el protocolo BOOTP.

Dhcp — Se debería utilizar el protocolo DHCP.

HWADDR=<dirección-MAC>

donde <dirección-MAC> es la dirección de hardware del dispositivo Ethernet en la forma de AA:BB:CC:DD:EE:FF. Esta directriz es útil en las máquinas con múltiples NICs para asegurarse de que a las interfaces se les asignen los nombres correctos de dispositivos sin importar el orden de carga configurado para cada módulo NIC. Esta directriz no debería ser usada en conjunto con MACADDR.

ONBOOT=<respuesta>

donde <respuesta> es una de las siguientes:

yes — El dispositivo debería activarse en el momento de arranque.

no — Este dispositivo no debería activarse en el momento de arranque.

TYPE=<nombre>

donde <respuesta> es Ethernet

USERCTL=<respuesta>

donde <respuesta> es una de las siguientes:

yes — Los usuarios que no sean root pueden controlar este dispositivo.

no — No se les permite controlar este dispositivo a los usuarios que no sean root.

IPV6INIT=<respuesta>

donde <respuesta> es una de las siguientes:

yes — Habilita ipv6 en esta interfaz.

no — No habilita ipv6 en esta interfaz.

IPV6ADDR=<dirección>

donde <dirección> es la dirección IPv6.

PEERDNS=<respuesta>

donde <respuesta> es una de las siguientes:

yes — Modifica /etc/resolv.conf si está activada la directriz DNS. Si está usando DCHP, la opción yes es la predeterminada.

no — No modificar /etc/resolv.conf.

IPADDR=<dirección>

donde <dirección> es la dirección IP.

NETMASK=<máscara>

Donde <máscara> es el valor de la máscara de red.

GATEWAY=<dirección>

donde <dirección> es la dirección IPv4 del enrutador o dispositivo de puerta de enlace (si existe).

DEVICE=eth0
BOOTPROTO=none
HWADDR=00:1C:C4:95:0C:7E
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
IPV6INIT=*yes*
IPV6ADDR=*2800:68:a:60::26*
PEERDNS=*no*
IPADDR=*172.30.60.26*
NETMASK=*255.255.255.0*
GATEWAY=*172.30.60.1*

Figura II-23 /etc/sysconfig/network-scripts/ifcfg-eth0

2.3.2 Configuración de hosts

Cuando el sistema intente resolver un nombre de host a una dirección IP, o de determinar el nombre de host para una dirección IP, hará referencia al archivo /etc/hosts antes de usar los servidores de nombres. Si aparece la dirección IP en el archivo /etc/hosts, no se utilizarán los servidores de nombres. Si la red contiene ordenadores cuyas direcciones IP no aparecen en DNS, se recomienda añadirlas al archivo /etc/hosts.

La sintaxis es:

<dirección IP> < nombre de dominio cualificado (FQDN)> < alias>

# direccion ip	nombre del domino FQDN	alias
127.0.0.1	localhost.localdomain	localhost
172.30.60.26	*dns.espoch.edu.ec*	*dns*
::1	*localhost.localdomain*	*localhost*
2800:68:a:60::26	dns6.espoch.edu.ec	dns6

Figura II-24 /etc/hosts

El archivo host.conf determina el orden de llamada a los servicios de resolución de nombres. Los parámetros posibles, separados por espacios o comas, son:

hosts: Búsqueda en el archivo /etc/hosts

bind: Llamada a un servidor de nombres

```
order hosts, bind
```

Figura II-25 /etc/host.conf

2.3.3 Archivos de Configuración del DNS

El archivo named.conf es una colección de declaraciones que utilizan opciones anidadas rodeadas por corchetes, { }.

Los administradores deben tener mucho cuidado cuando estén modificando named.conf para evitar errores sintácticos, puesto que hasta el error más pequeño puede impedir que el servicio named arranque.

Declaración options

La declaración options define opciones de configuración de servidor globales y configura otras declaraciones por defecto. Puede ser usado para especificar la ubicación del directorio de trabajo named, los tipos de consulta permitidos y más.

listen-on

Especifica la interfaz de red en la cual named escucha por solicitudes. Por defecto, todas las interfaces son usadas.

Al usar esta directiva en un servidor DNS que también actúa como un gateway, BIND puede ser configurado para sólo contestar solicitudes que se originan desde algunas de las redes.

query-source

Indica a BIND qué puerto usar para las consultas.

Las siguientes directivas son las que hacen posible la implementación de Doble Pila IPv4/IPv6.

```
listen-on port 53 { any; };
query-source  port 53;
listen-on-v6 port 53 { any; };
query-source-v6 port 53;
```

Figura II-26 /var/named/chroot/etc/named

Archivos de zona

Los Archivos de zona contienen información sobre un espacio de nombres particular. Estos son almacenados en el directorio de trabajo named, por defecto /var/named/. Cada archivo de zona es nombrado de acuerdo a la opción file en la declaración zone, usualmente en una forma que relaciona al dominio en cuestión e identifica el archivo como un archivo que contiene datos de zona.

Declaración zone

Una declaración zone define las características de una zona, tal como la ubicación de su archivo de configuración y opciones específicas de la zona.

type

Define el tipo de zona.

master — Designa el servidor de nombres actual como el servidor autoritativo para esa zona. Una zona se puede configurar como tipo master si los archivos de configuración de la zona residen en el sistema.

slave — Designa el servidor de nombres como un servidor esclavo para esa zona. También especifica la dirección IP del servidor de nombres maestro para la zona.

file

Especifica el nombre del archivo en el directorio de trabajo named que contiene los datos de configuración de zona.

allow-update

Especifica los hosts que están autorizados para actualizar dinámicamente la información en sus zonas. Por defecto, no se autoriza la actualización dinámica de la información.

Tenga cuidado cuando autorice a los hosts para actualizar la información de su zona. No habilite esta opción si no tiene confianza en el host que vaya a usar. Es mejor que el administrador actualice manualmente los registros de zona y que vuelva a cargar el servicio named.

La implementación de la zona sería así:

```
zone "espoch.edu.ec" IN {
                type master;
                file "espoch.edu.ec.zone";
                allow-update { none; };
};
```

```
zone "server" IN {
                type master;
                file "server.zone";
allow-update { none; };
};
```

Figura II-27 /var/named/chroot/etc/named.conf

En la declaración, la zona es identificada como espoch.edu.ec, el tipo es configurado a master y el servicio named se instruye para leer el archivo /var/named/espoch.edu.ec.zone. También le dice a named que no permita a ningún otro host que realice actualizaciones. Una declaración zone de servidor esclavo para espoch.edu.ec se ve un poco diferente comparado con el ejemplo anterior. Para un servidor esclavo, el tipo se coloca a slave y en lugar de la línea allow-update está una directiva diciéndole a named la dirección IP del servidor maestro.

Los mismos parámetros para la zona de resolución inversa.

Una vez configurado el archivo named.conf, se procede a configurar y crear los archivos de zonas ubicados en /var/named/chroot/var/named/.

Registros de recursos de archivos de zona

El componente principal de un archivo de zona es su registro de recursos.

Hay muchos tipos de registros de recursos de archivos de zona. A continuación le mostramos los tipos de registros más frecuentes:

$INCLUDE

Configura a named para que incluya otro archivo de zona en el archivo de zona donde se usa la directiva. Así se pueden almacenar configuraciones de zona suplementarias aparte del archivo de zona principal.

$TTL

Ajusta el valor Time to Live (TTL) predeterminado para la zona. Este es el tiempo, en segundos, que un registro de recurso de zona es válido. Cada recurso puede contener su propio valor TTL, el cual ignora esta directiva.

Cuando se decide aumentar este valor, permite a los servidores de nombres remotos hacer caché a la información de zona para un período más largo de tiempo, reduciendo el número de consultas para la zona y alargando la cantidad de tiempo requerido para proliferar cambios de registros de recursos.

A

Registro de dirección que resuelve un nombre de un anfitrión hacia una dirección IPv4 de 32 bits., como en el siguiente ejemplo:

<host> IN A <IP-address>

Si el valor <host> es omitido, el registro A apunta a una dirección IP por defecto para la parte superior del espacio de nombres. Este sistema es el objetivo para todas las peticiones no FQDN.

AAAA

Registro de dirección que resuelve un nombre de un anfitrión hacia una dirección IPv6 de 128 bits.

CNAME

Se refiere al Registro del nombre canónico, el cual enlaza un nombre con otro. Esta clase de registros es también conocida como un alias record.

El próximo ejemplo indica a named que cualquier petición enviada a <alias-name> apuntará al host, <real-name>. Los registros CNAME son usados normalmente para apuntar a servicios que usan un esquema de nombres común, tal como www para servidores Web.

<alias-name> IN CNAME <real-name>

MX

Registro de Mail eXchange, el cual indica dónde debería ir el correo enviado a un espacio de nombres particular controlado por esta zona.

IN MX <preference-value><email-server-name>

En este ejemplo, <preference-value> permite una clasificación numérica de los servidores de correo para un espacio de nombres, dando preferencia a algunos sistemas de correo sobre otros. El registro de recursos MX con el valor más bajo <preference-value> es preferido sobre los otros. Sin embargo, múltiples servidores de correo pueden tener el mismo valor para distribuir el tráfico de forma pareja entre ellos.

El <email-server-name> puede ser un nombre de servidor o FQDN.

IN MX 10 mail.example.com. IN MX 20 mail2.example.com.

En este ejemplo, el primer servidor de correo mail.example.com es preferido al servidor de correo mail2.example.com cuando se recibe correo destinado para el dominio example.com.

NS

Se refiere al Registro NameServer, el cual anuncia los nombres de servidores con autoridad para una zona particular.

El siguiente ejemplo es un ejemplo de un registro NS:

IN NS <nameserver-name>

Aquí, el <nameserver-name> debería ser un FQDN.

Luego, dos nombres de servidores son listados como servidores con autoridad para el dominio.

No es importante si estos nombres de servidores son esclavos o maestros; ambos son todavía considerados como servidores con autoridad.

IN NS dns1.example.com. IN NS dns2.

PTR

Registro PoinTeR (puntero), diseñado para apuntar a otra parte del espacio de nombres.

Los registros PTR son usados principalmente para la resolución inversa de nombres, pues ellos apuntan direcciones IP de vuelta a un nombre particular.

SOA

Registro de recursos Start Of Authority, que declara información importante de autoridad relacionada con espacios de nombres al servidor de nombres. Está situado detrás de las directivas, un registro SOA es el primer registro en un archivo de zona. El ejemplo siguiente muestra la estructura básica de un registro de recursos SOA:

```
@ IN SOA <primary-name-server>  <hostmaster-email> (
                    <serial-number>
                    <time-to-refresh>
                    <time-to-retry>
                    <time-to-expire>
                    <minimum-TTL> )
```

Figura II-28 Estructura del Registro de SOA

El símbolo @ coloca la directiva $ORIGIN (o el nombre de la zona, si la directiva $ORIGIN no está configurada) como el espacio de nombres que está siendo definido por este registro de recursos SOA. El nombre del host del servidor de nombres que tiene autoridad para este dominio es la directiva <primary-name-server> y el correo electrónico de la persona a contactar sobre este espacio de nombres es la directiva <hostmaster-email>.

La directiva <serial-number> es un valor numérico que es incrementado cada vez que se cambia el archivo de zona para así indicar a named que debería recargar esta zona. La directiva <time-to-refresh> es el valor numérico que los servidores esclavos utilizan para determinar cuánto tiempo debe esperar antes de preguntar al servidor de nombres maestro si se han

realizado cambios a la zona. El valor <serial-number> es usado por los servidores esclavos para determinar si está usando datos de la zona desactualizados y si debería refrescarlos.

La directiva <time-to-retry> es un valor numérico usado por los servidores esclavos para determinar el intervalo de tiempo que tiene que esperar antes de emitir una petición de actualización de datos en caso de que el servidor de nombres maestro no responda. Si el servidor maestro no ha respondido a una petición de actualización de datos antes de que se acabe el intervalo de tiempo <time-to-expire>, los servidores esclavos paran de responder como una autoridad por peticiones relacionadas a ese espacio de nombres.

La directiva <minimum-TTL> es la cantidad de tiempo que otros servidores de nombres guardan en caché la información de zona. Cuando se configura BIND, todos los tiempos son siempre referenciados en segundos. Sin embargo, es posible usar abreviaciones cuando se especifiquen unidades de tiempo además de segundos, tales como minutos (M), horas (H), días (D) y semanas (W).

SEGUNDOS	UNIDADES DE TIEMPO
60	1M
1800	30M
3600	1H
10800	3H
21600	6H
43200	12H
86400	1D

259200	3D
604800	1W
31536000	365D

Figura II-29 Unidades de tiempo

```
$TTL  86400
@             IN SOA        dns.espoch.edu.ec root.dns.espoch.edu.ec (
                                    42              ; serial (d. adams)
                                    3H              ; refresh
                                    15M             ; retry
                                    1W              ; expiry
                                    1D )            ; minimum
              IN NS  serverlan
serverlan     IN A   172.30.60.26
proxy  IN A   172.30.60.26
dns    IN A   172.30.60.26
dhcp   IN A   172.30.60.26
ftp    IN A   172.30.60.26
web    IN A   172.30.60.26
mail   IN A   172.30.60.26
serverlan     IN AAAA       2800:68:a:60::26
proxy6 IN AAAA       2800:68:a:60::26
dns6   IN AAAA       2800:68:a:60::26
dhcp6  IN AAAA       2800:68:a:60::26
ftp6   IN AAAA       2800:68:a:60::26
web6   IN AAAA       2800:68:a:60::26
mail6  IN AAAA       2800:68:a:60::26
```

Figura II-30 /var/named/chroot/var/named/espoch.edu.ec.zone

```
$TTL  86400
@             IN SOA      @  root.dns.espoch.edu.ec (
                                  42            ; serial (d. adams)
                                  3H            ; refresh
                                  15M           ; retry
                                  1W            ; expiry
                                  1D )          ; minimum
              IN NS       @
              IN A        127.0.0.1
              IN AAAA           ::1
26      IN PTR      serverlan.espoch.edu.ec.
26      IN PTR      proxy.espoch.edu.ec.
26      IN PTR      dns.espoch.edu.ec.
26      IN PTR      dhcp.espoch.edu.ec.
26      IN PTR      ftp.espoch.edu.ec.
26      IN PTR      web.espoch.edu.ec.
26      IN PTR      mail.espoch.edu.ec.
0.6.0.0.a.0.0.0.8.6.0.0.0.0.0.8.2 IN PTR      serverlan.espoch.edu.ec.
0.6.0.0.a.0.0.0.8.6.0.0.0.0.0.8.2 IN PTR      proxy6.espoch.edu.ec.
0.6.0.0.a.0.0.0.8.6.0.0.0.0.0.8.2 IN PTR      dns6.espoch.edu.ec.
0.6.0.0.a.0.0.0.8.6.0.0.0.0.0.8.2 IN PTR      dhcp6.espoch.edu.ec.
0.6.0.0.a.0.0.0.8.6.0.0.0.0.0.8.2 IN PTR      ftp6.espoch.edu.ec.
0.6.0.0.a.0.0.0.8.6.0.0.0.0.0.8.2 IN PTR      web6.espoch.edu.ec.
0.6.0.0.a.0.0.0.8.6.0.0.0.0.0.8.2 IN PTR      mail6.espoch.edu.ec.
```

Figura II-31 /var/named/chroot/var/named/server.zone

A continuación, es necesario configurar el /etc/resolv.conf

```
search espoch.edu.ec
nameserver 2800:68:A:60::26
nameserver 172.30.60.26
nameserver ::1
```

Figura II-32 /etc/resolv.conf

La línea `search' especifica en que dominios buscar para cualquier nombre de máquina al que se desea conectar, `nameserver' especifica la dirección del servidor de nombres, en este caso el localhost, que es donde se ejecuta.

2.3.4 Levantar el Demonio Named del servicio DNS

Una vez configurados los archivos se debe levantar el demonio named con el comando:

service named start

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

Para que se inicie de manera automática la próxima vez que inicie el sistema, utilice lo siguiente:

chkconfig named on

Una vez levantado el servicio se hará las pruebas correspondientes, con los siguientes comandos.

Nslookup (Name System Lookup)

Es una herramienta que permite consultar un servidor de nombres y obtener información relacionada con el dominio o el host y así diagnosticar los eventuales problemas de configuración que pudieran haber surgido en el DNS.

```
[root@serverlan ~]# nslookup www.google.com
Server:         2800:68:a:60::26
Address:        2800:68:a:60::26#53

Non-authoritative answer:
www.google.com  canonical name = www.l.google.com.
```

```
Name:   www.l.google.com
Address: 64.233.169.103
Name:   www.l.google.com
Address: 64.233.169.104
Name:   www.l.google.com
Address: 64.233.169.147
Name:   www.l.google.com
Address: 64.233.169.99
```

Figura II-33 Prueba del comando nslookup en el Internet con IPv4

```
[root@serverlan ~]# nslookup -type=AAAA www.ipv6forum.com
Server:         2800:68:a:60::26
Address:        2800:68:a:60::26#53
Non-authoritative answer:
www.ipv6forum.com       has AAAA address 2001:a18:1:20::22
Authoritative answers can be found from:
com     nameserver = J.GTLD-SERVERS.NET.
com     nameserver = K.GTLD-SERVERS.NET.
com     nameserver = L.GTLD-SERVERS.NET.
com     nameserver = M.GTLD-SERVERS.NET.
com     nameserver = A.GTLD-SERVERS.NET.
com     nameserver = B.GTLD-SERVERS.NET.
com     nameserver = C.GTLD-SERVERS.NET.
com     nameserver = D.GTLD-SERVERS.NET.
com     nameserver = E.GTLD-SERVERS.NET.
com     nameserver = F.GTLD-SERVERS.NET.
com     nameserver = G.GTLD-SERVERS.NET.
com     nameserver = H.GTLD-SERVERS.NET.
com     nameserver = I.GTLD-SERVERS.NET.
```

Figura II-34 Prueba del comando nslookup en el Internet en IPv6

```
[root@serverlan ~]# nslookup ftp.espoch.edu.ec
Server:         2800:68:a:60::26
Address:        2800:68:a:60::26#53

Name:   ftp.espoch.edu.ec
Address: 172.30.60.26

[root@serverlan ~]#
```

Figura II-35 Prueba del comando nslookup en la Intranet con IPv4

```
[root@serverlan ~]# nslookup -type=AAAA dhcp6.espoch.edu.ec
Server:         2800:68:a:60::26
Address:        2800:68:a:60::26#53

dhcp6.espoch.edu.ec     has AAAA address 2800:68:a:60::26

[root@serverlan ~]#
```

Figura II-36 Prueba del comando nslookup en la Intranet con IPv6

Dig (Domain Information Groper)

Permite realizar consultas a los servidores DNS, por lo que es muy útil para comprobar si el DNS está correctamente configurado en nuestra máquina. Permite comprobar tanto el mapeo de nombres a IPs como el mapeo inverso de IPs a nombres, pero sólo sirve para Internet, ya que no mira en /etc/hosts (sólo utiliza /etc/resolv.conf). Su sintaxis es:

dig [@servidor_dns] <nombre> [opciones] [tipo]

[@servidor_dns]: nombre o IP del servidor DNS al que queremos dirigir nuestra consulta, utilizará los servidores DNS listados en /etc/resolv.conf

<nombre>: nombre de dominio cuya IP queremos resolver.

[tipo]: tipo de consulta. Valores posibles:

A: IP del servidor que aloja al dominio (por defecto).

NS: servidores DNS.

MX: servidores de correo.

ANY: todas las anteriores.

AAAA: IP en IPv6.

```
[root@serverlan ~]# dig www.google.com

; <<>> DiG 9.3.4-P1 <<>> www.google.com
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3430
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 7, ADDITIONAL: 0

;; QUESTION SECTION:
;www.google.com.                 IN      A

;; ANSWER SECTION:
www.google.com.         604800  IN      CNAME   www.l.google.com.
www.l.google.com.       300     IN      A       64.233.169.99
www.l.google.com.       300     IN      A       64.233.169.103
www.l.google.com.       300     IN      A       64.233.169.104
www.l.google.com.       300     IN      A       64.233.169.147
```

```
;; AUTHORITY SECTION:
l.google.com.        86400   IN      NS      g.l.google.com.
l.google.com.        86400   IN      NS      a.l.google.com.
l.google.com.        86400   IN      NS      b.l.google.com.
l.google.com.        86400   IN      NS      c.l.google.com.
l.google.com.        86400   IN      NS      d.l.google.com.
l.google.com.        86400   IN      NS      e.l.google.com.
l.google.com.        86400   IN      NS      f.l.google.com.

;; Query time: 907 msec
;; SERVER: 2800:68:a:60::26#53(2800:68:a:60::26)
;; WHEN: Mon Feb  9 16:06:35 2009
;; MSG SIZE  rcvd: 228

[root@serverlan ~]#
```

Figura II-37 Prueba del comando dig en el Internet con IPv4

```
[root@serverlan ~]# dig AAAA www.ipv6forum.com

; <<>> DiG 9.3.4-P1 <<>> AAAA www.ipv6forum.com
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9244
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 0
;; QUESTION SECTION:
;www.ipv6forum.com.             IN      AAAA

;; ANSWER SECTION:
www.ipv6forum.com.      1786    IN      AAAA    2001:a18:1:20::22
```

```
;; AUTHORITY SECTION:
ipv6forum.com.          545     IN      NS      ns2.hexago.com.
ipv6forum.com.          545     IN      NS      ns1.hexago.com.
;; Query time: 0 msec
;; SERVER: 2800:68:a:60::26#53(2800:68:a:60::26)
;; WHEN: Mon Feb  9 16:10:10 2009
;; MSG SIZE  rcvd: 106
```

Figura II-38 Prueba del comando dig en el Internet con IPv6

```
[root@serverlan ~]# dig dns.espoch.edu.ec



; <<>> DiG 9.3.4-P1 <<>> dns.espoch.edu.ec
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58514
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; QUESTION SECTION:
;dns.espoch.edu.ec.             IN      A

;; ANSWER SECTION:
dns.espoch.edu.ec.      86400   IN      A       172.30.60.26

;; AUTHORITY SECTION:
espoch.edu.ec.          86400   IN      NS      serverlan.espoch.edu.ec.

;; ADDITIONAL SECTION:
serverlan.espoch.edu.ec. 86400  IN      A       172.30.60.26
serverlan.espoch.edu.ec. 86400  IN      AAAA    2800:68:a:60::26
```

```
;; Query time: 0 msec
;; SERVER: 2800:68:a:60::26#53(2800:68:a:60::26)
;; WHEN: Mon Feb  9 16:03:12 2009
;; MSG SIZE  rcvd: 119

[root@serverlan ~]#
```

Figura II-39 Prueba del comando dig en la Intranet con IPv4

```
[root@serverlan ~]# dig AAAA proxy6.espoch.edu.ec

; <<>> DiG 9.3.4-P1 <<>> AAAA proxy6.espoch.edu.ec
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1298
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; QUESTION SECTION:
;proxy6.espoch.edu.ec.          IN      AAAA

;; ANSWER SECTION:
proxy6.espoch.edu.ec.   86400   IN      AAAA    2800:68:a:60::26

;; AUTHORITY SECTION:
espoch.edu.ec.          86400   IN      NS      serverlan.espoch.edu.ec.

;; ADDITIONAL SECTION:
serverlan.espoch.edu.ec. 86400  IN      A       172.30.60.26
serverlan.espoch.edu.ec. 86400  IN      AAAA    2800:68:a:60::26

;; Query time: 0 msec
;; SERVER: 2800:68:a:60::26#53(2800:68:a:60::26)
```

```
;; WHEN: Mon Feb  9 16:05:25 2009
;; MSG SIZE  rcvd: 134

[root@serverlan ~]#
```

Figura II-40 Prueba del comando dig en la Intranet con IPv6

Host

Una herramienta simple para hacer búsquedas en Servidores DNS. Es utilizada para convertir nombres en direcciones IP y viceversa. De modo predefinido realiza las búsquedas en las Servidores DNS definidos en el fichero /etc/resolv.conf, pudiendo definirse opcionalmente el Servidor DNS a consultar.

```
[root@serverlan ~]# host www.google.com
www.google.com is an alias for www.l.google.com.
www.l.google.com has address 64.233.169.104
www.l.google.com has address 64.233.169.147
www.l.google.com has address 64.233.169.99
www.l.google.com has address 64.233.169.103
[root@serverlan ~]#
```

Figura II-41 Prueba del comando host en el Internet con IPv4

```
[root@serverlan ~]# host -t  AAAA www.ipv6forum.com
www.ipv6forum.com has IPv6 address 2001:a18:1:20::22
[root@serverlan ~]#
```

Figura II-42 Prueba del comando host en el Internet con IPv6

```
[root@serverlan ~]# host ftp.espoch.edu.ec
ftp.espoch.edu.ec has address 172.30.60.26
[root@serverlan ~]#
```

Figura 43 Prueba del comando host en la Intranet con IPv4

```
[root@serverlan ~]# host -t AAAA  dhcp6.espoch.edu.ec
dhcp6.espoch.edu.ec has IPv6 address 2800:68:a:60::26
[root@serverlan ~]#
```

Figura II-44 Prueba del comando host en la Intranet con IPv6

2.4 SERVIDOR DE SISTEMA DE NOMBRES DE DOMINIO (DNS) CACHING

2.4.1 Activar IPv6

network, añadir el nombre del servidor, dominio y "yes" en networking:

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=serverlan.espoch.edu.ec
```

Figura II-45 /etc/sysconfig/network

ifcfg-eth0, se configura la tarjeta de red en Doble-Pila.

```
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:1C:C4:95:0C:7E
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
IPV6INIT=yes
```

```
IPV6ADDR=2800:68:a:60::26
PEERDNS=no
IPADDR=172.30.60.26
NETMASK=255.255.255.0
GATEWAY=172.30.60.1
```

Figura II-46 /etc/sysconfig/network-scripts/ifcfg-eth0

2.4.2 Configuración de hosts

hosts, facilita el manejo de las direcciones IP; aquí se configura la dirección IP, el nombre del servidor y el nombre del dominio.

# direccion ip	nombre del domino FQDN	alias
127.0.0.1	localhost.localdomain	localhost
172.30.60.26	*dns.espoch.edu.ec*	*dns*
::1	*localhost.localdomain*	*localhost*
2800:68:a:60::26	*dns6.espoch6.edu.ec*	*dns6*

Figura II-47 /etc/hosts

2.4.3 Archivos de Configuración del DNS-CACHING

named.conf

Archivo principal del DNS

named.ip6.local

Sirve para la traducción inversa del localhost en IPv6

named.ca

Contiene los registros de servidores raíz

Mapa de los servidores Raíz

Figura II-48 Mapa de los servidores Raíz

Descripción de los servidores raíz

Server	Operator	Locations	IP Addresses	AS Number
A	VeriSign, Inc.	**Sites: 2** Global: 2 Local: 0 **Dulles, VA, US ***; **Ashburn, VA, US ***	IPv4: 198.41.0.4 IPv6: 2001:503:BA3E::2:30	19836
B	Information Sciences Institute	**Sites: 1** Global: 1 Local: 0 **Marina Del Rey, CA, US ***	IPv4: 192.228.79.201 IPv6: 2001:478:65::53	none
C	Cogent Communications	**Sites: 6** Herndon, VA, US; Los Angeles, CA, US; New York, NY, US; Chicago, IL, US; Frankfurt, DE; Madrid, ES	IPv4: 192.33.4.12	2149
D	University of Maryland	**Sites: 1** Global: 1 Local: 0 **College Park, MD, US**	IPv4: 128.8.10.90	27
E	NASA Ames Research Center	**Sites: 1** Global: 1 Local: 0	IPv4: 192.203.230.10	297

		Mountain View, CA, US		
F	Internet Systems Consortium, Inc.	**Sites: 46** Global: 2 Local: 44 Ottawa, Canada *; **Palo Alto, CA, US ***; San Jose, CA, US; New York, NY, US *; **San Francisco, CA, US ***; Madrid, ES; Hong Kong, HK; Los Angeles, CA, US *; Rome, Italy; Auckland, NZ *; Sao Paulo, BR; Beijing, CN; Seoul, KR *; Moscow, RU; Taipei, TW; Dubai, AE; Paris, FR *; Singapore, SG; Brisbane, AU; Toronto, CA *; Monterrey, MX; Lisbon, PT *; Johannesburg, ZA; Tel Aviv, IL; Jakarta, ID; Munich, DE *; Osaka, JP *; Prague, CZ *; Amsterdam, NL *; Barcelona, ES *; Nairobi, KE; Chennai, IN; London, UK *; Santiago de Chile, CL; Dhaka, BD; Karachi, PK; Torino, IT; Chicago, IL, US *; Buenos Aires, AR; Caracas, VE; Oslo, NO; Panama, PA; Quito, EC; Kuala Lumpur, Malaysia *; Suva, Fiji; Cairo, Egypt	IPv4: 192.5.5.241 IPv6: 2001:500:2f::f	3557
G	U.S. DOD Network Information Center	**Sites: 1** Global: 1 Local: 0 **Columbus, OH, US**	IPv4: 192.112.36.4	568
H	U.S. Army Research Lab	**Sites: 1** Global: 1 Local: 0	IPv4: 128.63.2.53 IPv6: 2001:500:1::803f:235	13

		Aberdeen Proving Ground, MD, US *		
I	Autonomica	**Sites: 31** Stockholm, SE; Helsinki, FI; Milan, IT; London, UK; Geneva, CH; Amsterdam, NL; Oslo, NO; Bangkok, TH; Hong Kong, HK; Brussels, BE; Frankfurt, DE; Ankara, TR; Bucharest, RO; Chicago, IL, US; Washington, DC, US; Tokyo, JP; Kuala Lumpur, MY; Palo Alto, CA, US; Jakarta, ID; Wellington, NZ; Johannesburg, ZA; Perth, AU; San Francisco, CA, US; Singapore, SG; Miami, FL, US; Ashburn, VA, US; Mumbai, IN; Beijing, CN; Manila, PH; Doha, QA; Colombo, LK	IPv4: 192.36.148.17	29216
J	VeriSign, Inc.	**Sites: 52** Dulles, VA, US (3 sites); Ashburn, VA, US *; Vienna, VA, US; Miami, FL, US; Atlanta, GA, US; Seattle, WA, US; Chicago, IL, US; New York, NY, US; Los Angeles, CA, US; Honolulu, HI, US; Mountain View, CA, US (2 sites); San Francisco, CA, US (2 sites) *; Dallas, TX, US; Amsterdam, NL; London, UK; Stockholm, SE (2 sites); Tokyo, JP; Seoul, KR; Beijing, CN; Singapore, SG; Dublin, IE; Kaunas, LT; Nairobi, KE; Montreal, CA; Quebec, CA; Sydney, AU; Cairo, EG; Warsaw, PL; Brasilia, BR; Sao Paulo, BR; Sofia, BG; Prague, CZ; Johannesburg, ZA; Toronto, CA; Buenos Aires, AR; Madrid, ES; Vienna, AT; Fribourg, CH; Hong Kong, HK; Turin, IT; Mumbai, IN; Oslo, NO; Brussels, BE; Paris,	IPv4: 192.58.128.30 IPv6: 2001:503:C27::2:30	26415

		FR; Helsinki, FI; Frankfurt, DE; Riga, LV		
K	RIPE NCC	**Sites: 17** Global: 5 Local: 12 **London, UK ***; **Amsterdam, NL ***; Frankfurt, DE; Athens, GR *; Doha, QA; Milan, IT *; Reykjavik, IS; Helsinki, FI *; Geneva, CH *; Poznan, PL; Budapest, HU *; Abu Dhabi, AE; **Tokyo, JP**; Brisbane, AU *; **Miami, FL, US ***; **Delhi, IN**; Novosibirsk, RU	IPv4: 193.0.14.129 IPv6: 2001:7fd::1	25152
L	ICANN	**Sites: 2** Global: 2 Local: 0 **Los Angeles, CA, US ***; **Miami, FL, US ***	IPv4: 199.7.83.42 IPv6: 2001:500:3::42	20144
M	WIDE Project	**Sites: 6** Global: 5 Local: 1 **Tokyo, JP (3 sites) ***; Seoul, KR; **Paris, FR ***; **San Francisco, CA, US ***	IPv4: 202.12.27.33 IPv6: 2001:dc3::35	7500

Tabla II-15 Descripción de los Servidores Raíz

Servers in total: 167

Notes:
* indicates IPv6 capable sites
bold indicates global sites

```
[root@serverlan ~]# dig

; <<>> DiG 9.3.4-P1 <<>>

;; global options:  printcmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64179

;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0,
ADDITIONAL: 14

;; QUESTION SECTION:

;.                      IN      NS

;; ANSWER SECTION:

.               516326  IN      NS      C.ROOT-SERVERS.NET.

.               516326  IN      NS      D.ROOT-SERVERS.NET.

.               516326  IN      NS      E.ROOT-SERVERS.NET.

.               516326  IN      NS      F.ROOT-SERVERS.NET.

.               516326  IN      NS      G.ROOT-SERVERS.NET.

.               516326  IN      NS      H.ROOT-SERVERS.NET.

.               516326  IN      NS      I.ROOT-SERVERS.NET.

.               516326  IN      NS      J.ROOT-SERVERS.NET.

.               516326  IN      NS      K.ROOT-SERVERS.NET.

.               516326  IN      NS      L.ROOT-SERVERS.NET.

.               516326  IN      NS      M.ROOT-SERVERS.NET.

.               516326  IN      NS      A.ROOT-SERVERS.NET.

.               516326  IN      NS      B.ROOT-SERVERS.NET.


;; ADDITIONAL SECTION:

A.ROOT-SERVERS.NET.     602726  IN      A       198.41.0.4

A.ROOT-SERVERS.NET.     602726  IN      AAAA    2001:503:ba3e::2:30

B.ROOT-SERVERS.NET.     602726  IN      A       192.228.79.201

C.ROOT-SERVERS.NET.     602726  IN      A       192.33.4.12

D.ROOT-SERVERS.NET.     602726  IN      A       128.8.10.90

E.ROOT-SERVERS.NET.     602726  IN      A       192.203.230.10

F.ROOT-SERVERS.NET.     602726  IN      A       192.5.5.241
```

```
F.ROOT-SERVERS.NET.     602726  IN      AAAA    2001:500:2f::f
G.ROOT-SERVERS.NET.     602726  IN      A       192.112.36.4
H.ROOT-SERVERS.NET.     602726  IN      A       128.63.2.53
H.ROOT-SERVERS.NET.     602726  IN      AAAA    2001:500:1::803f:235
I.ROOT-SERVERS.NET.     602726  IN      A       192.36.148.17
J.ROOT-SERVERS.NET.     602726  IN      A       192.58.128.30
J.ROOT-SERVERS.NET.     602726  IN      AAAA    2001:503:c27::2:30

;; Query time: 0 msec
;; SERVER: 2800:68:a:60::26#53(2800:68:a:60::26)
;; WHEN: Mon Feb  9 15:49:33 2009
;; MSG SIZE  rcvd: 500
[root@serverlan ~]#
```

Figura II-49 Prueba de los Servidores Raíz con el comando dig

2.4.4 Levantar el Demonio Named del servicio DNS

Una vez configurados los archivos se debe levantar el demonio named con el comando:

service named start

Para que se inicie de manera automática la próxima vez que inicie el sistema, utilice lo siguiente:

chkconfig named on

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

2.5 SERVIDOR DE HOSTING (WEB)

2.5.1 Activar IPv6

network, añadir el nombre del servidor, dominio y "yes" en networking:

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=serverlan.espoch.edu.ec
```

Figura II-50 /etc/sysconfig/network

ifcfg-eth0, se configura la tarjeta de red en Doble-Pila.

```
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:1C:C4:95:0C:7E
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
IPV6INIT=yes
IPV6ADDR=2800:68:a:60::26
PEERDNS=no
IPADDR=172.30.60.26
NETMASK=255.255.255.0
GATEWAY=172.30.60.1
```

Figura II-51 /etc/sysconfig/network-scripts/ifcfg-eth0

2.5.2 Configuración de hosts

hosts, facilita el manejo de las direcciones IP; aquí se configura la dirección IP, el nombre del servidor y el nombre del dominio.

# direccion ip	nombre del domino FQDN	alias
127.0.0.1	localhost.localdomain	localhost
172.30.60.26	*web.espoch.edu.ec*	*web*
::1	*localhost.localdomain*	*localhost*
2800:68:a:60::26	*web6.espoch.edu.ec*	*web6*

Figura II-52 /etc/hosts

2.5.3 Archivos de Configuración del DNS

resolv.conf, es el cliente del sistema de nombres de dominio, responsable de mapear una petición de información de un programa en un host.

```
search espoch.edu.ec
nameserver 2800:68:A:60::26
nameserver 172.30.60.26
nameserver ::1
```

Figura II-53 /etc/resolv.conf

2.5.4 Configuración del Servidor Web

Toda la configuración del servicio httpd es manejada por su archivo de configuración, /etc/httpd/conf/httpd.conf

Listen

El comando Listen establece los puertos en los que secure Web server acepta las peticiones entrantes secure Web server está configurado para escuchar en el puerto 80 para comunicaciones no seguras y (en máquinas virtuales que define el servidor seguro) en el puerto 443 para comunicaciones seguras.

Para puertos por debajo de 1024, el comando httpd deberá ser ejecutado como root. Para el puerto 1024 y superiores, el comando httpd puede ser ejecutado como si se fuera un usuario cualquiera.

El comando Listen también se puede usar para especificar direcciones IP específicas en las cuales aceptará conexiones el servidor.

ServerAdmin

Debería ser la dirección de correo del administrador del secure Web server. Esta dirección de correo aparecerá en los mensajes de error generados por el servidor para páginas web, de tal manera que los usuarios pueden comunicar errores enviando correo al administrador. El comando ServerAdmin ya se encuentra en la dirección root@localhost.

NameVirtualHost

La directriz NameVirtualHost asocia una dirección IP y el número de puerto, si es necesario, para cualquier máquina virtual basada en nombres. El hospedaje virtual basado en nombres permite a un Servidor HTTP Apache servir a dominios diferentes sin utilizar múltiples direcciones IP.

VirtualHost

<VirtualHost> y </VirtualHost> envuelven directivas de configuración que se aplican a máquinas virtuales. La mayoría de las directivas de configuración pueden usarse en etiquetas de máquina virtual, y sólo se aplicarán a esa máquina virtual.

Existen varias etiquetas VirtualHost que rodean a algunos modelos de directivas de configuración así como de espacios en blanco que tendrá que rellenar con información para configurar la máquina virtual.

ServerAdmin

Configure la directriz ServerAdmin a la dirección de correo electrónico del administrador del servidor Web. Esta dirección de correo aparecerá en los mensajes de error en las páginas generadas por el servidor Web, de tal manera que los usuarios pueden comunicar errores enviando correo al administrador.

DocumentRoot

DocumentRoot es el directorio que contiene la mayoría de los archivos HTML que se entregarán en respuesta a peticiones.

El directorio predeterminado DocumentRoot para servidores web seguros y no seguros es /var/www/html. Por ejemplo, el servidor puede recibir una petición para el siguiente documento:

ServerName

Use la directriz ServerName para configurar un nombre de servidor y un número de puerto (que coincida con la directriz Listen) para el servidor. El ServerName no necesita coincidir con el nombre real de la máquina. Por ejemplo, el servidor Web puede ser www.example.com pero el nombre del servidor es en realidad foo.example.com. El valor especificado en

ServerName debe ser un nombre del Servicio de Nombres de Dominio (Domain Name Service, DNS) válido que pueda ser resuelto por el sistema

ErrorLog

ErrorLog especifica el archivo donde se guardan los errores del servidor. Por defecto, esta directriz es configurada a /var/log/httpd/error_log.

CustomLog

CustomLog identifica el archivo de registro y su formato. Por defecto, el registro de acceso es guardado al archivo /var/log/httpd/access_log mientras que los errores se guardan en el archivo /var/log/httpd/error_log.

DirectoryIndex

DirectoryIndex es la página por defecto que entrega el servidor cuando hay una petición de índice de un directorio especificado con una barra (/) al final del nombre del directorio.

Cuando un usuario pide una página, recibe la página del índice del directorio, DirectoryIndex, si existe, o un listado de directorios generado por el servidor. El valor por defecto para DirectoryIndex es index.html. El servidor intentará encontrar cualquiera de estos archivos y entregará el primero que encuentre. Si no encuentra ninguno de estos archivos y Options Indexes está configurado para ese directorio, el servidor genera y devuelve una lista, en formato HTML, de los subdirectorios y archivos dentro del directorio, a menos que la característica de listar directorios esté desactivada.

```
Listen [2800:68:a:60::26]:80
Listen [::1]:80
Listen 172.30.60.26:80
Listen 127.0.0.1:80
ServerAdmin root@localhost
# Virtual Host para direccionamiento IPv6

NameVirtualHost [2800:68:a:60::26]:80

<VirtualHost [2800:68:a:60::26]:80>
   ServerAdmin webmaster@web6. espoch.edu.ec
   DocumentRoot /var/www/html/web6
   ServerName web6. espoch.edu.ec
   ErrorLog logs/web6. espoch.edu.ec-error_log
   CustomLog logs/web6. espoch.edu.ec-access_log common
   DirectoryIndex index.html
</VirtualHost>

# Virtual Host para direccionamiento IPv4

NameVirtualHost 172.30.60.26:80

<VirtualHost 172.30.60.26:80>
   ServerAdmin webmaster@web.espoch.edu.ec
   ServerName web. espoch.edu.ec
   DocumentRoot /var/www/html/web
   ErrorLog logs/web. espoch.edu.ec-error_log
   CustomLog logs/web. espoch.edu.ec-access_log common
   DirectoryIndex index.html
</VirtualHost>
```

Figura II-54 /etc/httpd/conf/httpd.conf

2.5.5 Levantar el Servicio de Apache

Una vez configurados los archivos se debe levantar el demonio httpd con el comando:

service httpd start

Para que se inicie de manera automática la próxima vez que inicie el sistema, utilice lo siguiente:

chkconfig httpd on

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

2.6 SERVIDOR DE CORREO ELECTRONICO (MAIL)

2.6.1 Activar IPv6

network, añadir el nombre del servidor, dominio y “yes” en networking:

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=serverlan.espoch.edu.ec
```

Figura II-55 etc/sysconfig/network

ifcfg-eth0, se configura la tarjeta de red en Doble-Pila.

```
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:1C:C4:95:0C:7E
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
```

```
IPV6INIT=yes
IPV6ADDR=2800:68:a:60::26
PEERDNS=no
IPADDR=172.30.60.26
NETMASK=255.255.255.0
GATEWAY=172.30.60.1
```

Figura II-56 etc/sysconfig/network-scripts/ifcfg-eth0

2.6.2 Configuración de hosts

hosts, facilita el manejo de las direcciones IP; aquí se configura la dirección IP, el nombre del servidor y el nombre del dominio.

# direccion ip	nombre del domino FQDN	alias
127.0.0.1	localhost.localdomain	localhost
172.30.60.26	*mail.espoch.edu.ec*	*mail*
::1	*localhost.localdomain*	*localhost*
2800:68:a:60::26	*mail6.espoch6.edu.ec*	*mail6*

Figura II-57 /etc/hosts

2.6.3 Archivos de Configuración del DNS

resolv.conf, es el cliente del sistema de nombres de dominio, responsable de mapear una petición de información de un programa en un host.

```
search espoch.edu.ec
nameserver 2800:68:A:60::26
nameserver 172.30.60.26
nameserver ::1
```

Figura II-58 /etc/resolv.conf

2.6.4 Configuración del Servidor Mail

```
inet_protocols = ipv4, ipv6
queue_directory = /var/spool/postfix
command_directory = /usr/sbin
daemon_directory = /usr/libexec/postfix
mail_owner = postfix
myhostname = serverlan.espoch.edu.ec
mydomain = espoch.edu.ec
myorigin = $mydomain
mydestination = $myhostname, localhost.$mydomain, localhost
unknown_local_recipient_reject_code = 550
mynetworks = 172.30.0.0/16, 127.0.0.0/8
mynetworks = [::1]/128 [[2800:68:a::]/48
relay_domains = $mydestination
recipient_delimiter = +
home_mailbox = Maildir/
header_checks = regexp:/etc/postfix/header_checks
XXX_destination_concurrency_limit
debug_peer_level = 2
debugger_command =
        PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin
        xxgdb $daemon_directory/$process_name $process_id & sleep 5
sendmail_path = /usr/sbin/sendmail.postfix
newaliases_path = /usr/bin/newaliases.postfix
mailq_path = /usr/bin/mailq.postfix
setgid_group = postdrop
html_directory = no
manpage_directory = /usr/share/man
sample_directory = /usr/share/doc/postfix-2.3.3/samples
readme_directory = /usr/share/doc/postfix-2.3.3/README_FILES
```

Figura II-59 /etc/postfix/main.cf

2.6.5 Configuración del Servidor Dovecot

protocols

Habilito protocolo los protocolos imap pop3 imaps pop3s

listen

Habilito para que dovecot pueda escuchar tanto IPv6 como IPv3.

```
protocols = imap pop3 imaps pop3s
listen = [::]
listen = *
```

Figura II-60 /etc/dovecot.conf

2.6.6 Levantar el Servicio de Correo Electrónico

Una vez configurados los archivos se debe levantar el demonio postfix con el comando:

service postfix start

Para que se inicie de manera automática la próxima vez que inicie el sistema, utilice lo siguiente:

chkconfig postfix on

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

2.6.7 Levantar el Servicio de Dovecot

Una vez configurados los archivos se debe levantar el demonio dovecot con el comando:

service dovecot start

Para que se inicie de manera automática la próxima vez que inicie el sistema, utilice lo siguiente:

chkconfig dovecot on

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

2.7 CONFIGURACION DEL SERVIDOR PROXY

2.7.1 Activar IPv6

network, añadir el nombre del servidor, dominio y “yes” en networking:

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=serverlan.espoch.edu.ec
```

Figura II-61 /etc/sysconfig/network

ifcfg-eth0, se configura la tarjeta de red en Doble-Pila.

```
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:1C:C4:95:0C:7E
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
IPV6INIT=yes
IPV6ADDR=2800:68:a:60::26
PEERDNS=no
IPADDR=172.30.60.26
NETMASK=255.255.255.0
GATEWAY=172.30.60.1
```

Figura II-62 etc/sysconfig/network-scripts/ifcfg-eth0

2.7.2 Configuración de hosts

hosts, facilita el manejo de las direcciones IP; aquí se configura la dirección IP, el nombre del servidor y el nombre del dominio.

127.0.0.1	localhost.localdomain	localhost
172.30.60.26	*proxy.espoch.edu.ec*	*proxy*
::1	*localhost.localdomain*	*localhost*
2800:68:a:60::26	*proxy6.espoch.edu.ec*	*proxy6*

Figura II-63 /etc/hosts

2.7.3 Archivos de Configuración del DNS

resolv.conf, es el cliente del sistema de nombres de dominio, responsable de mapear una petición de información de un programa en un host.

```
search espoch.edu.ec
nameserver 2800:68:A:60::26
nameserver 172.30.60.26
nameserver ::1
```

Figura II-64 /etc/resolv.conf

2.7.4 Configuración del Servidor Proxy

Toda la configuración de servicio de squid es manejada por su archivo de configuración, /usr/local/etc/squid.conf

http_port

Squid por defecto utilizará el puerto 3128 para atender peticiones, sin embargo se puede especificar que lo haga en cualquier otro puerto o bien que lo haga en varios puertos a la vez.

cache_me

Específica la cantidad de memoria que se utilizará para lo siguiente:

• Objetos en tránsito.

• Objetos frecuentemente utilizados (Hot).

• Objetos negativamente almacenados en el caché.

Los datos de estos objetos se almacenan en bloques de 4 Kb. El parámetro cache_mem especifica un límite máximo en el tamaño total de bloques acomodados, donde los objetos en tránsito tienen mayor prioridad. Sin embargo los objetos Hot y aquellos negativamente almacenados en el caché podrán utilizar la memoria no utilizada hasta que esta sea requerida. De ser necesario, si un objeto en tránsito es mayor a la cantidad de memoria especificada, Squid excederá lo que sea necesario para satisfacer la petición. De modo predefinido se establecen 8 MB.

cache_swap_low

Determina el nivel en porcentaje de capacidad mínima aceptada por squid, es decir, los objetos se mantendrán en la memoria temporal hasta que se llegue al límite mínimo.

cache_swap_high

Indica el nivel en porcentaje de capacidad máxima aceptada por squid, es decir, los objetos se mantendrán en la cache hasta que se llegue al límite máximo.

cache_dir

Determina el tamaño de la caché en el disco duro que utilizará squid para almacenar información. La cantidad se determina en función de lo que el usuario necesite.

cache_access_log

Determina en que directorio se realizará el registro de accesos al squid.

cache_log

Específica la ubicación del archivo donde se reporta datos generales del funcionamiento de squid, así como los mensajes del sistema.

cache_store_log

Indica la ubicación del archivo donde se reporta los objetos que son guardados en memoria temporal, así como el tiempo que van a permanecer. Para deshabilitar esta opción se utiliza none.

dns_nameservers

Específica los servidores dns.

acl [nombre de la lista] src [lo que compone a la lista]

Es necesario establecer Listas de Control de Acceso que definan una red o bien ciertas maquinas en particular. A cada lista se le asignará una Regla de Control de Acceso que permitirá o denegará el acceso a Squid

http_access [deny o allow] [lista de control de acceso]

Estas definen si se permite o no el acceso hacia **Squid**. Se aplican a las **Listas de Control de Acceso**.

```
http_port 8080
http_port [2800:68:a:60::26]:8080
http_port 172.30.60.26:8080
cache_mem 8 MB
cache_swap_low 90
cache_swap_high 95
cache_dir ufs /usr/local/var/cache 100 16 256
access_log /usr/local/var/logs/access.log squid
cache_log /usr/local/var/logs/cache.log
dns_nameservers 2800:68:a:60::26
acl espoch src 2800:68:a:60::/64 172.30.60.0/24 2800:68:a:124::/64 2800:68:a:30::/64
http_access allow espoch
```

Figura II-65 /usr/local/etc/squid.conf

2.7.5 Levantar el Demonio Squid del servicio de Proxy

Una vez configurados los archivos se debe levantar el demonio squid con el comando:

/usr/local/sbin/squid start

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

2.8 CONFIGURACION DEL SERVIDOR PARA LA ASIGNACION DINAMICA DE HOST

2.8.1 Activar IPv6

network, añadir el nombre del servidor, dominio y “yes” en networking:

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=serverlan.espoch.edu.ec
```

Figura II-66 /etc/sysconfig/network

ifcfg-eth0, se configura la tarjeta de red en Doble-Pila.

```
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:1C:C4:95:0C:7E
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
IPV6INIT=yes
IPV6ADDR=2800:68:a:60::26
PEERDNS=no
IPADDR=172.30.60.26
NETMASK=255.255.255.0
GATEWAY=172.30.60.1
```

Figura II-67 /etc/sysconfig/network-scripts/ifcfg-eth0

2.8.2 Configuración de hosts

hosts, facilita el manejo de las direcciones IP; aquí se configura la dirección IP, el nombre del servidor y el nombre del dominio.

# direccion ip	nombre del domino FQDN	alias
127.0.0.1	localhost.localdomain	localhost
172.30.60.26	*dhcp.espoch.edu.ec*	*dhcp*
::1	*localhost.localdomain*	*localhost*
2800:68:a:60::26	*dhcp6.espoch.edu.ec*	*dhcp6*

Figura II-68 /etc/hosts

2.8.3 Archivos de Configuración del DNS

resolv.conf, es el cliente del sistema de nombres de dominio, responsable de mapear una petición de información de un programa en un host.

```
search espoch.edu.ec
nameserver 2800:68:A:60::26
nameserver 172.30.60.26
nameserver ::1
```

Figura II-69 /etc/resolv.conf

2.8.4 Configuración del DHCP versión 6 en el Servidor

Toda la configuración de servicio de dhcp6s es manejada por su archivo de configuración, /etc/dhcp6s.conf.

interface <interface name>

Declaración de una interfaz se utiliza para informar al servidor DHCPv6 que los enlaces y anfitriones declarado dentro de ella están conectados al mismo segmento de red.

server-preference <server preference value>;

La prioridad más alta del servidor, por lo general es 255.

renew-time <renew time>;

Tiempo de renovación de las direcciones.

rebind-time <rebind time>;

Tiempo de restablecimiento de conexión.

prefer-life-time <preferred lifetime>;

Tiempo de vida preferido para cada dirección.

valid-life-time <valid lifetime>;

Tiempo de vida válido para cada dirección.

allow rapid-commit;

Habilita al dhcp6s para permitir una verificación rápida de los mensajes entre el cliente y las solicitudes de respuesta.

option dns_server <ipv6 addresses or domain name list>;

Determina los servidores DNS.

link <link name>{

Es utilizado para proveer al servidor DHCPv6 la asignación de pools, rangos y prefijos de direcciones IPv6.

pool

Declara el pool de direcciones a utilizar para la asignación.

range <ipv6 address> to <ipv6 address>/<prefix length>;

Para indicar desde donde hasta que direcciones pueden ser asignadas a los host.

prefix <prefix>/<prefix length>;

Especifica el prefijo de la subred.

host <host name>{

Comienza una declaración de host, es decir para asignar una dirección estática a un host específico.

duid <DUID>;

Es el Identificador Unico para un cliente DHCPv6.

iaidinfo[iaid <IAID number>];

Describe la información del iaid.

iaid <IAID number>;

IAID es el Identificador de la Identity Association; IA es una colección de direcciones asignadas a un cliente.

address[<ipv6 address>/<prefix length>];

Especifica la dirección del host a asignar.

```
interface eth0 {
    server-preference 255;
    renew-time 60;
    rebind-time 90;
    prefer-life-time 130;
    valid-life-time 200;
    allow rapid-commit;
    option dns_servers 2800:68:a:60::26 espoch.edu.ec;

link vlan6 {
   pool{
      range 2800:68:a:60::3 to 2800:68:a:60::30/64;
      prefix 2800:68:a::/48;
   };
};

link vlan24 {
   pool{
      range 2800:68:a:124::3 to 2800:68:a:124::30/64;
      prefix 2800:68:a::/48;
   };
};
```

Figura II-70 /etc/dhcp6s.conf

2.8.5 Levantar el Demonio dhcp6s

Una vez configurados los archivos se debe levantar el demonio dhcp6s con el comando:

service dhcp6s start

Para que se inicie de manera automática la próxima vez que inicie el sistema, utilice lo siguiente:

chkconfig dhcp6s on

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

2.8.6 Configuración del DHCP versión 6 en el Cliente

Toda la configuración del dhcp cliente es manejada por su archivo de configuración, /etc/dhcp6c.conf.

interface

Indica la interfaz a utilizar.

send rapid-commit

Habilita al dhcp6c para solicitar al servidor dhcp6s ejecute una verificación rápida.

request domain-name-servers

Habilita el dhcp6c para solicitar las direcciones del servidor DNS que se encuentran configurados en el servidor dhcp6s.

```
interface eth0 {
      send rapid-commit;
      request domain-name-servers;
         };
```

Figura II-71 /etc/dhcp6c.conf

2.8.7 Levantar el Demonio dhcp6c

Una vez configurado, se procede a levantar el servicio de dhcp6s, con el siguiente comando:

dhcp6c eth0

2.9 CONFIGURACION DEL SERVIDOR FTP

2.9.1 Activar IPv6

network, añadir el nombre del servidor, dominio y "yes" en networking:

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=serverlan.espoch.edu
```

Figura II-72 /etc/sysconfig/network

ifcfg-eth0, se configura la tarjeta de red en Doble-Pila.

```
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:1C:C4:95:0C:7E
ONBOOT=yes
TYPE=Ethernet
USERCTL=no
IPV6INIT=yes
IPV6ADDR=2800:68:a:60::26
PEERDNS=no
IPADDR=172.30.60.26
NETMASK=255.255.255.0
GATEWAY=172.30.60.1
```

Figura II-73 /etc/sysconfig/network-scripts/ifcfg-eth0

2.9.2 Configuración de hosts

hosts, facilita el manejo de las direcciones IP; aquí se configura la dirección IP, el nombre del servidor y el nombre del dominio.

# direccion ip	nombre del domino FQDN	alias
127.0.0.1	localhost.localdomain	localhost
172.30.60.26	*ftp.espoch.edu.ec*	*ftp*
::1	*localhost.localdomain*	*localhost*
2800:68:a:60::26	*ftp6.espoch6.edu.ec*	*ftp6*

Figura II-74 /etc/hosts

2.9.3 Archivos de Configuración del DNS

resolv.conf, es el cliente del sistema de nombres de dominio, responsable de mapear una petición de información de un programa en un host.

```
search espoch.edu.ec
nameserver 2800:68:A:60::26
nameserver 172.30.60.26
nameserver ::1
```

Figura II-75 /etc/resolv.conf

2.9.4 Configuración del VSFTP

Toda la configuración del servicio vsftpd es manejada por su archivo de configuración, /etc/vsftpd/vsftpd.conf.

anonymous_enable

Al estar activada, se permite que los usuarios anónimos se conecten. Se aceptan los nombres de usuario anonymous y ftp.

write_enable

Cuando está activada, se permiten los comandos FTP que pueden modificar el sistema de archivos.

local_umask

Especifica el valor de umask para la creación de archivos. Observe que el valor por defecto está en forma octal (un sistema numérico con base ocho), que incluye un prefijo de "0". De lo contrario el valor es tratado como un valor entero de base 10. El valor por defecto 022.

dirmessage_enable

Al estar activada, se mostrará un mensaje cada vez que un usuario entra en un directorio con un archivo de mensaje. Este mensaje se encuentra dentro del directorio al que se entra.

El nombre de este archivo se especifica en la directriz message_file y por defecto es message.

xferlog_enable

Cuando se activa, vsftpd registra las conexiones (solamente formato vsftpd) y la información de transferencia, al archivo de registro especificado en la directriz vsftpd_log_file (por defecto es /var/log/vsftpd.log). Si

xferlog_std_format está configurada a YES, se registra la información de transferencia de archivo pero no las conexiones y en su lugar se utiliza el archivo de registro especificado en xferlog_file (por defecto /var/log/xferlog). Es importante observar que se utilizan ambos archivos y formatos de registro si dual_log_enable tiene el valor de YES.

connect_from_port_20

Cuando está activada, vsftpd se ejecuta con privilegios suficientes para abrir el puerto 20 en el servidor durante las transferencias de datos en modo activo.

Al desactivar esta opción, se permite que vsftpd se ejecute con menos privilegios, pero puede ser incompatible con algunos clientes FTP.

xferlog_std_format

Cuando se activa en combinación con xferlog_enable, sólo se escribe un archivo de registro compatible con wu-ftpd al archivo especificado en la directriz xferlog_file (por defecto /var/log/xferlog).

Es importante resaltar que este archivo solamente registra transferencias de archivos y no las conexiones al servidor.

ftpd_banner

Si está activada, se muestra la cadena de caracteres especificada en esta directriz cuando se establece una conexión con el servidor. banner_file puede sobre escribir esta opción.

chroot_list_enable

Cuando está activada, se coloca en una prisión de chroot a los usuarios locales listados en el archivo especificado en la directriz chroot_list_file.

Si se utiliza en combinación con la directriz chroot_local_user, los usuarios locales listados en el archivo especificado en la directriz chroot_list_file, no se colocan en una prisión chroot luego de conectarse.

chroot_list_file

Específica el archivo que contiene una lista de los usuarios locales a los que se hace referencia cuando la directriz chroot_list_enable está en YES.

Listen

Cuando esta directriz está activada vsftpd se ejecuta en modo independiente, para activar ipv4.

listen_address

Especifica la dirección IPv4 en la cual vsftpd escucha por las conexiones de red.

pam_service_name

Especifica el nombre de servicio PAM para vsftpd.

userlist_enable

Cuando está activada, se les niega el acceso a los usuarios listados en el archivo especificado por la directriz userlist_file. Puesto que se niega el acceso al cliente antes de solicitar la contraseña, se previene que los usuarios suministren contraseñas sin encriptar sobre la red.

listen_ipv6

Cuando esta directriz está activada vsftpd se ejecuta en modo independiente, pero solamente escucha a los sockets IPv6.

listen_address6

Especifica la dirección IPv6 en la cual vsftpd escucha por conexiones de red cuando listen_ipv6 está configurada a YES.

```
#/etc/vsftpd/vsftpd.conf
anonymous_enable=YES
write_enable=NO
local_umask=022
xferlog_enable=YES
connect_from_port_20=YES
xferlog_std_format=YES
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd/chroot_list
listen=YES
listen_address=172.30.60.26
pam_service_name=vsftpd
userlist_enable=YES
```

Figura II-76 /etc/vsftpd/vsftpd.conf

Si se están ejecutando varias copias de vsftpd con diferentes direcciones IP, el archivo de configuración para cada copia del demonio vsftpd debe tener un valor diferente para esta directriz. Como necesitamos para ipv6 el archivo de configuración será /etc/vsftpd/vsftpd6.conf

```
#/etc/vsftpd/vsftpd6.conf
anonymous_enable=YES
write_enable=NO
local_umask=022
xferlog_enable=YES
connect_from_port_20=YES
xferlog_std_format=YES
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd/chroot_list
listen_ipv6=YES
listen_address6=2800:68:a:60::26
pam_service_name=vsftpd
userlist_enable=YES
```

Figura II-77 /etc/vsftpd/vsftpd6.conf

2.9.5 Levantar el Demonio Vsftpd

Una vez configurados los archivos se debe levantar el demonio vsftpd con el comando:

service vsftpd start

Para que se inicie de manera automática la próxima vez que inicie el sistema, utilice lo siguiente:

chkconfig vsftpd on

Para verificar que se ha levantado correctamente el servicio se puede observar el archivo /var/log/messages, el comando sería tail –f /var/log/messages

CAPITULO III

PRUEBAS Y DEPURACION

3 PRUEBAS Y DEPURACION

En esta etapa de prueba y depuraciones, se verifico los servicios levantados en IPv4, IPv6 e IPv4/ IPv6, en la Intranet de la Escuela superior Politécnica de Chimborazo

3.1 CLIENTE EN IPV4

3.1.1 Servicio de Resolución de Nombres

Resolución del Servidor DNS en IPv4, la misma que se accedió a la página html://web.espoch.edu.ec.

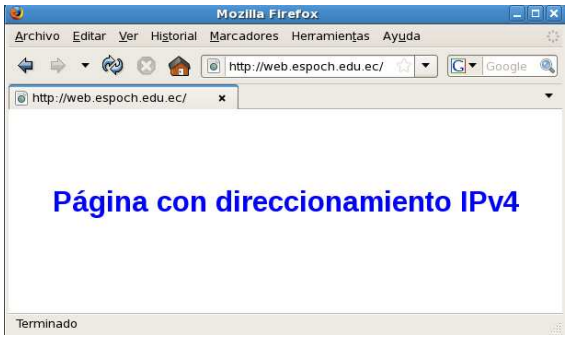

Figura III-78 Prueba del Servidor DNS con IPv4

3.1.2 Servicio de Hosting

Resolución del servidor Apache en IPv4, la misma que se accedió a la página html://172.30.60.26

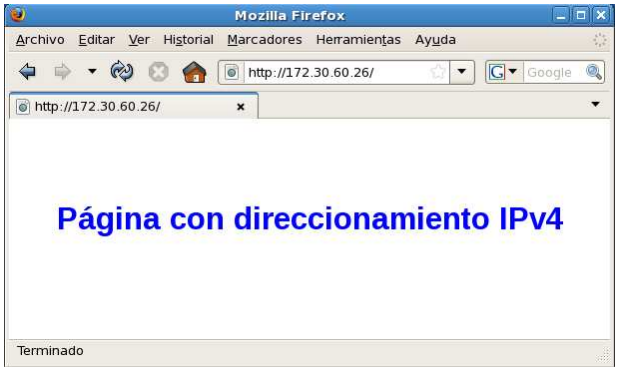

Figura III-79 Prueba del Servidor Apache con IPv4

3.1.3 Servicio de Correo Electrónico

Resolución del servidor de Correo en IPv4, con dirección 172.30.60.26 el mismo que se probó con varios usuarios.

Figura III-80 Prueba del Servidor de Correo Electrónico con IPv4

3.1.4 Servicio de Proxy

Resolución del servidor Proxy 'Squid' en IPv4, las peticiones hechas al servidor la dirección y el número de puerto es 172.30.60.26:8080, la herramienta que nos ayudo para ver estas peticiones fue un sniffer WIRESHARK.

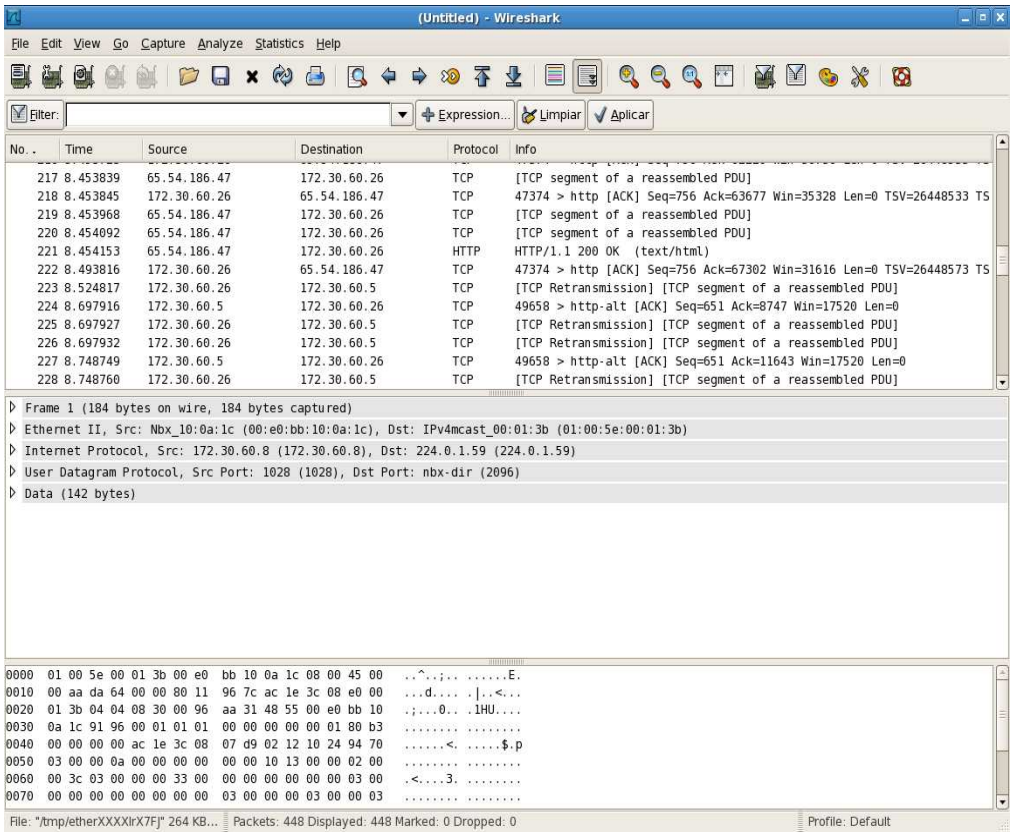

Figura III-81 Prueba del Servidor Proxy con IPv4

3.1.5 Servicio de Configuración Dinámica de Host

Resolución del servidor DHCP en IPv4, las peticiones hechas al servidor con la dirección 172.30.60.26, la petición hecha fue desde la plataforma Windows Vista.

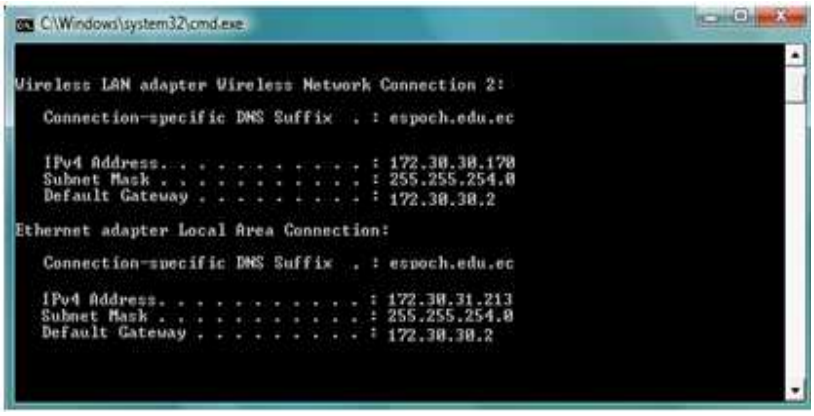

Figura III-82 Prueba del Servidor DHCP con IPv4

3.1.6 Servicio de Transferencia de Archivos

Resolución del servidor FTP en IPv4, las peticiones hechas al servidor con la dirección 172.30.60.26.

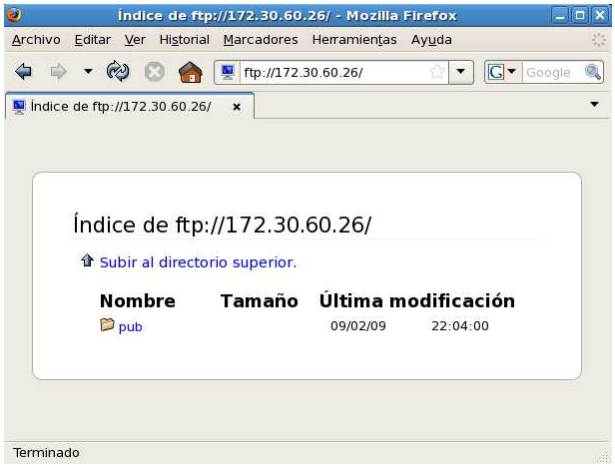

Figura III-83 Prueba del Servidor FTP con IPv4

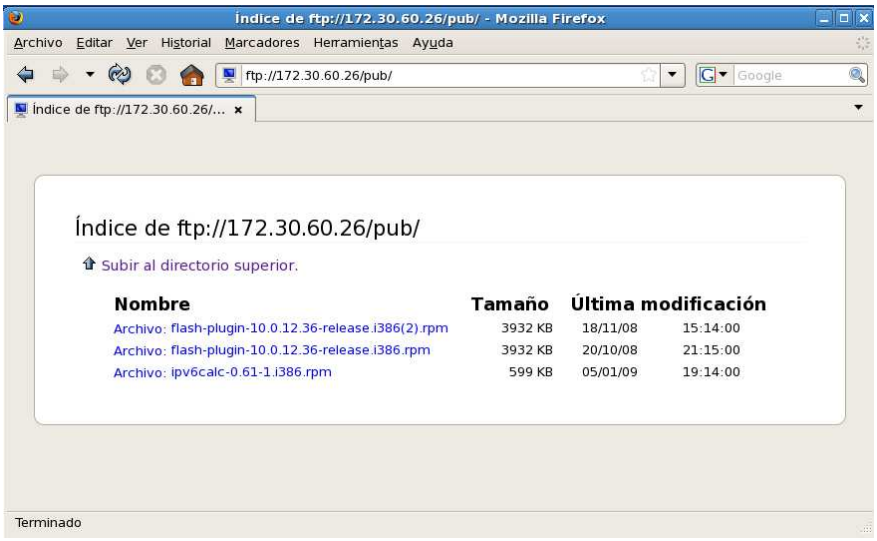

Figura III-84 Lista de archivos FTP con IPv4

3.2 CLIENTE EN IPV6

3.2.1 Servicio de Resolución de Nombres

Resolución del Servidor DNS en IPv6, la misma que se accedió a la página html://web6.espoch.edu.ec.

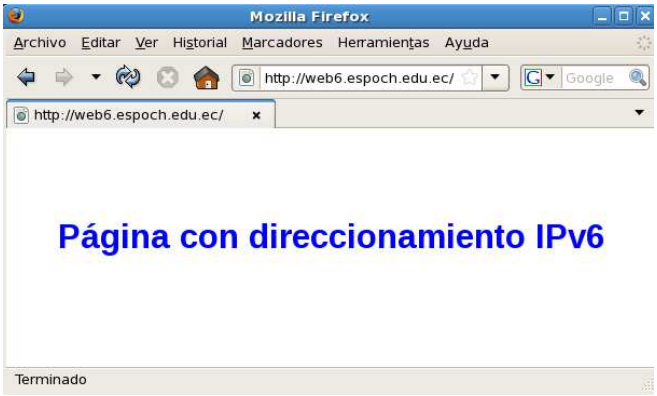

Figura III-85 Prueba del Servidor DNS con IPv6

3.2.2 Servicio de Hosting

Resolución del servidor Apache en IPv6, la misma que se accedió a la página html://[2800:68:a:60::26]

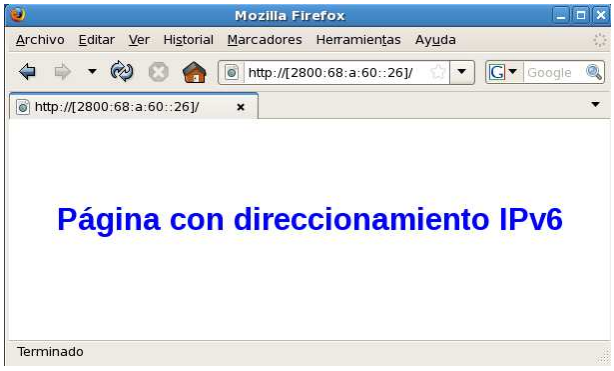

Figura III-86 Prueba del Servidor Apache con IPv6

3.2.3 Servicio de Correo Electrónico

Resolución del servidor de Correo en IPv6, con dirección 2800:68:a:60::26 el mismo que se probó con varios usuarios

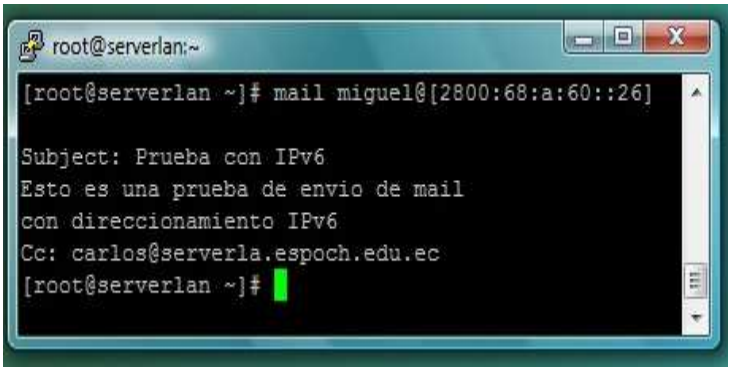

Figura III-87 Prueba del Servidor de Correo Electrónico con IPv6

3.2.4 Servicio de Proxy

Resolución del servidor Proxy 'Squid' en IPv6, las peticiones hechas al servidor con la dirección y el número de puerto es [2800:68:a:60::26]:8080, la herramienta que nos ayudo para ver estas peticiones fue un sniffer WIRESHARK.

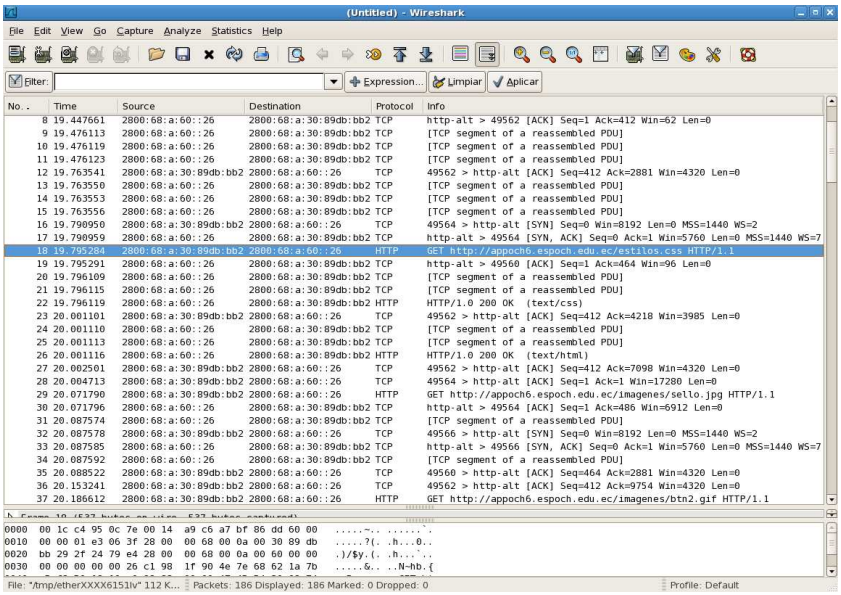

Figura III-88 Prueba del Servidor Proxy con IPv6

3.2.5 Servicio de Configuración Dinámica de Host

Resolución del servidor DHCP en IPv6, las peticiones hechas al servidor con la dirección 2800:68:a:60::26

Figura III-89 Prueba del Servidor DHCP con IPv6

3.2.6 Servicio de Transferencia de Archivos

Resolución del servidor FTP en IPv6, las peticiones hechas al servidor con la dirección 2800:68:a:60::26.

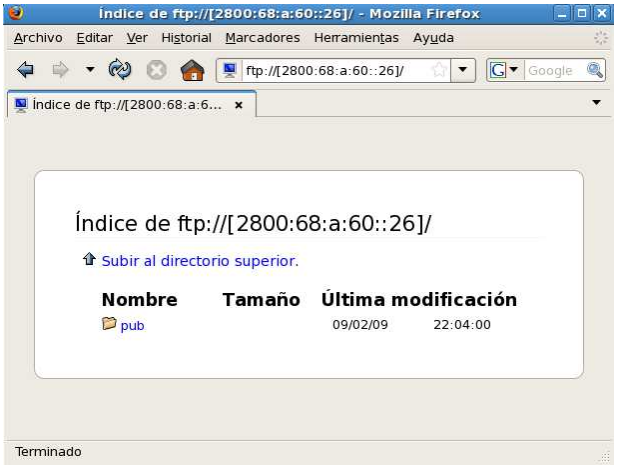

Figura III-90 Prueba del Servidor FTP con IPv6

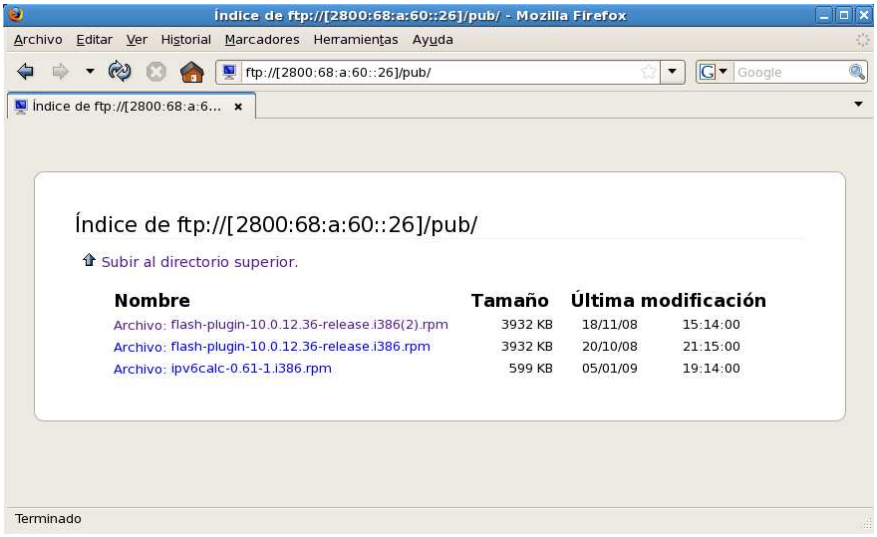

Figura III-91 Lista de archivos FTP con IPv6

3.3 CLIENTE EN DOBLE-PILA

3.3.1 Servicio de Resolución de Nombres

Resolución del Servidor DNS en doble pila.

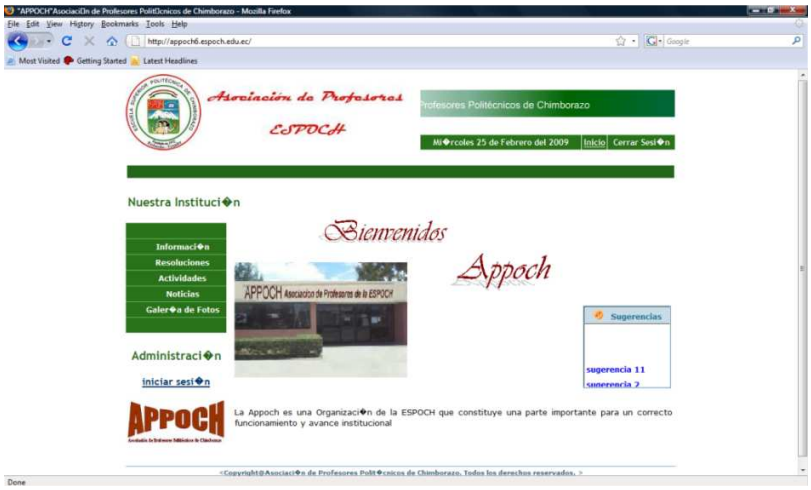

Figura III-92 Prueba del Servidor DNS con Doble Pila IPv4/IPv6

3.3.2 Servicio de Hosting

Resolución del servidor Apache en doble pila IPv4/IPv6, pruebas hechas en la Intranet de la Espoch.

Figura III-93 Prueba del Servidor Apache con Doble Pila IPv4/IPv6

Figura III-94 Prueba del Servidor Apache con Doble Pila IPv4/IPv6

Figura III-95 Prueba del Servidor Apache con Doble Pila IPv4/IPv6

Figura III-96 Prueba del Servidor Apache con Doble Pila IPv4/IPv6

3.3.3 Servicio de Correo Electrónico

Resolución del servidor de Correo Electrónico en doble pila IPv4/IPv6, pruebas hechas en la Intranet de la Espoch.

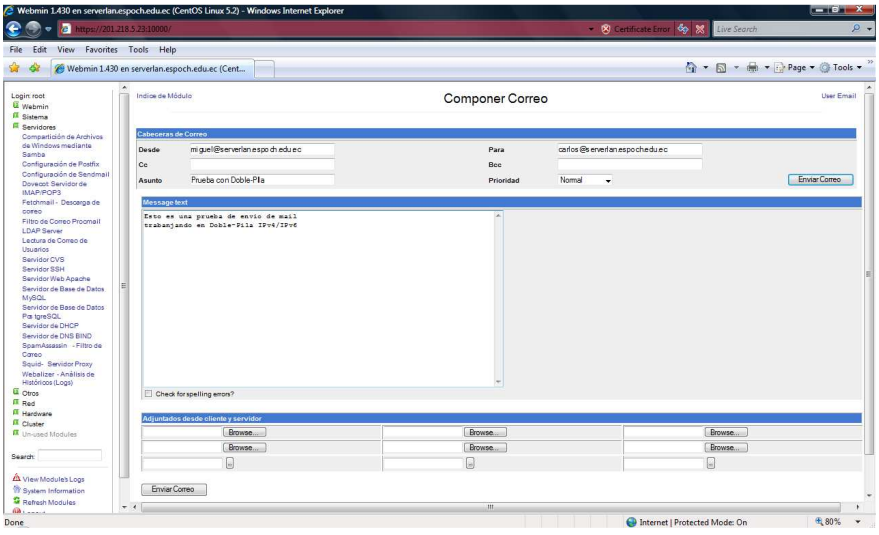

Figura III-97 Prueba del Servidor de Correo Electrónico con Doble Pila IPv4/IPv6 envió

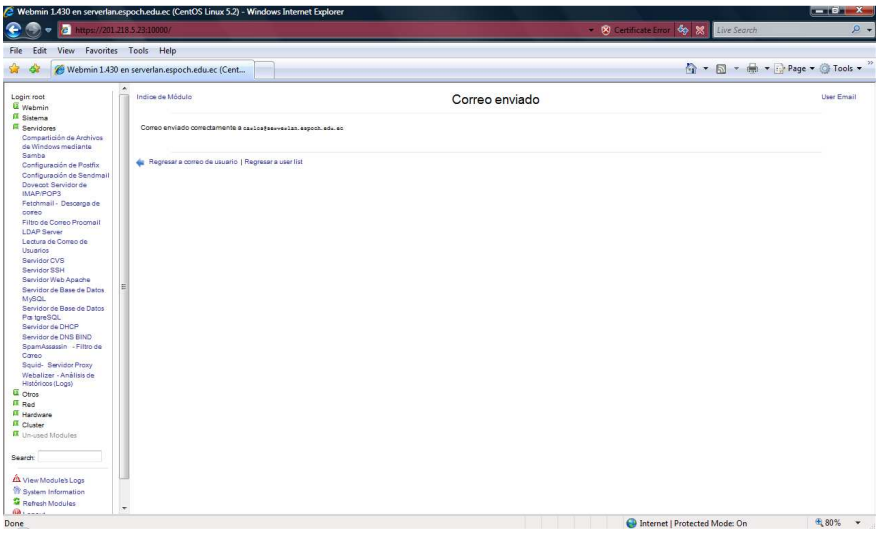

Figura III-98 Prueba del Servidor de Correo Electrónico con Doble Pila IPv4/IPv6 confirmación

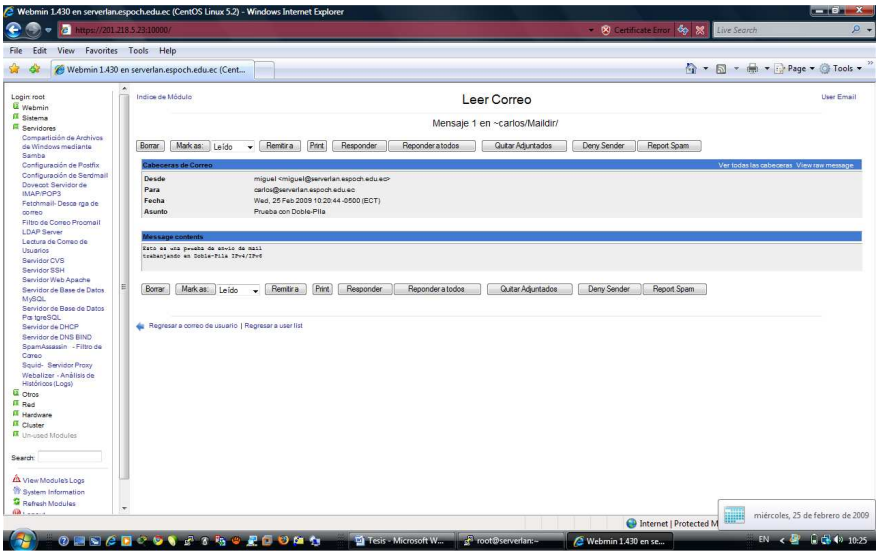

Figura III-99 Prueba del Servidor de Correo Electrónico con Doble Pila IPv4/IPv6 desde el usuario Carlos

3.3.4 Servicio de Proxy

Resolución del servidor Proxy 'Squid' en doble pila IPv4/IPv6, la herramienta que nos ayudo para ver estas peticiones fue un sniffer WIRESHARK.

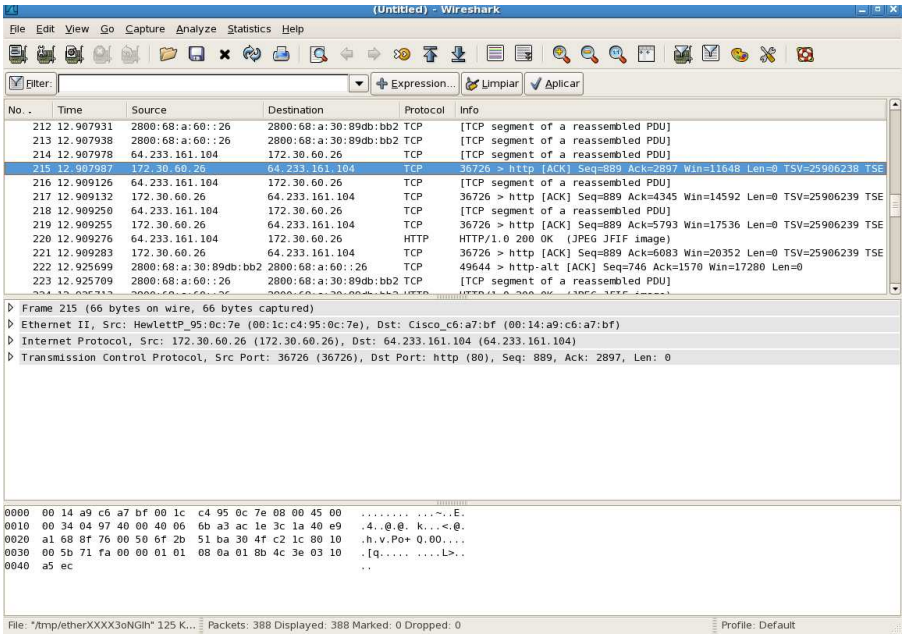

Figura III-100 Prueba del Servidor Proxy con Doble Pila IPv4/IPv6

3.3.5 Servicio de Configuración Dinámica de Host

Resolución del servidor DHCP en doble pila IPv4/IPv6.

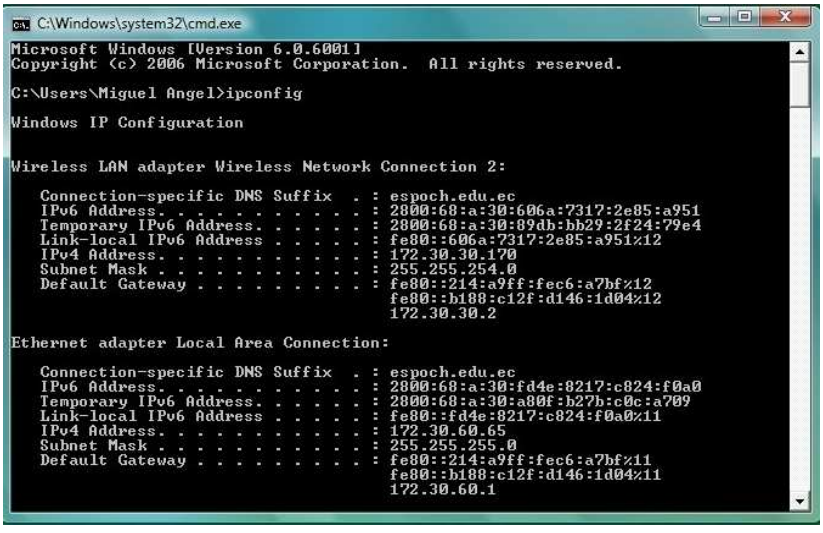

Figura III-101 Prueba del Servidor DHCP con Doble Pila IPv4/IPv6

3.3.6 Servicio de Transferencia de Archivos

Resolución del servidor FTP en doble pila IPv4/IPv6.

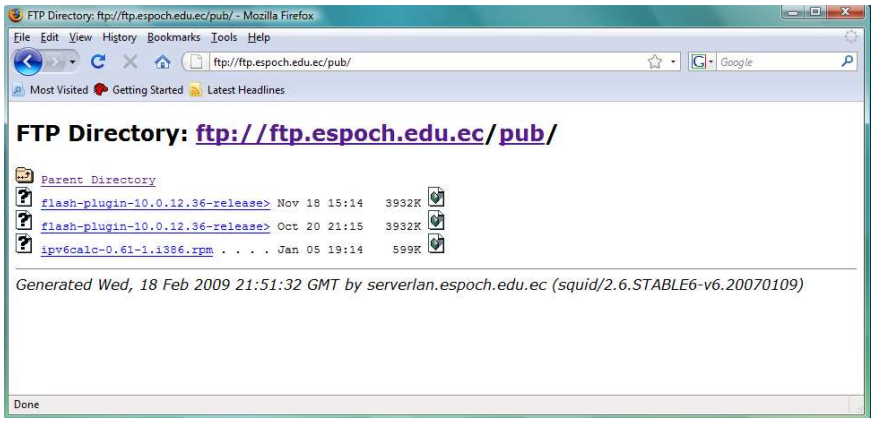

Figura III-102 Prueba del Servidor FTP con Doble Pila IPv4/IPv6

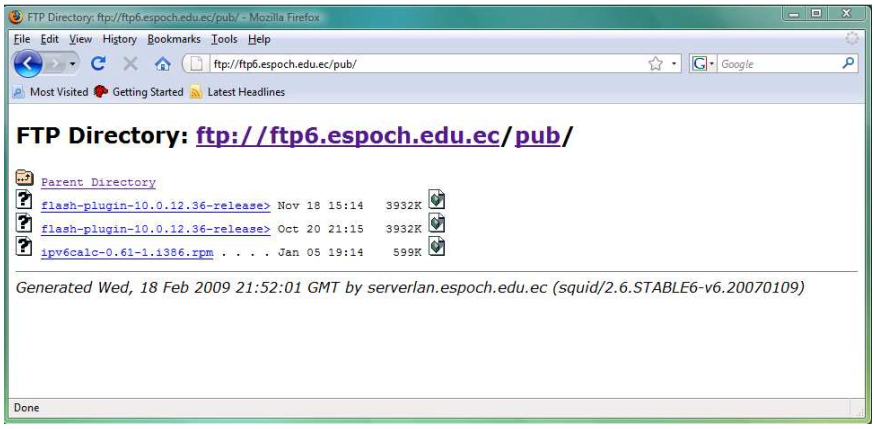

Figura III-103 Prueba del Servidor FTP con Doble Pila IPv4/IPv6

3.3.7 Pruebas de Conexión al Internet Avanzado

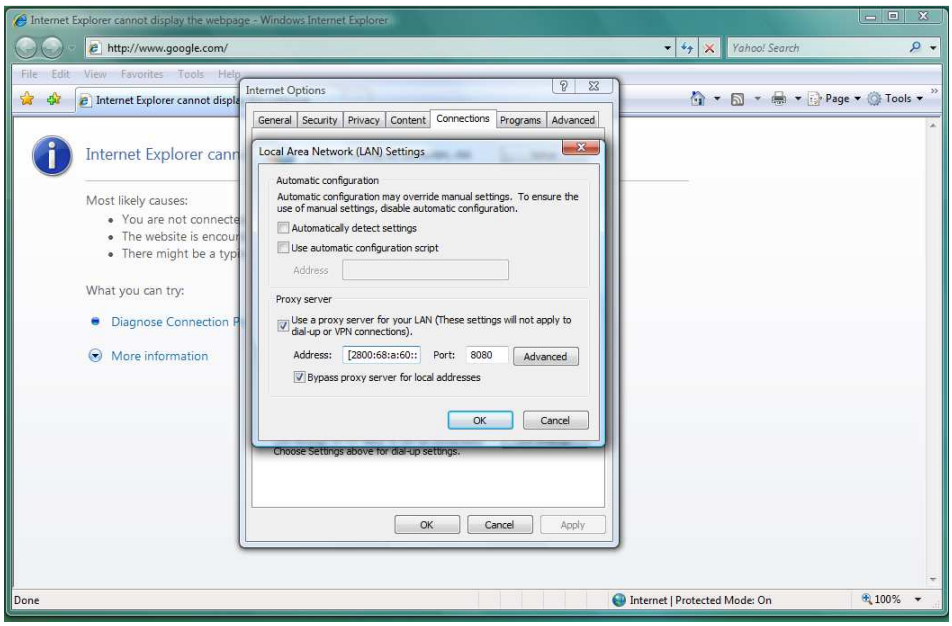

Figura III-104 Configuración del Proxy en Internet Explorer

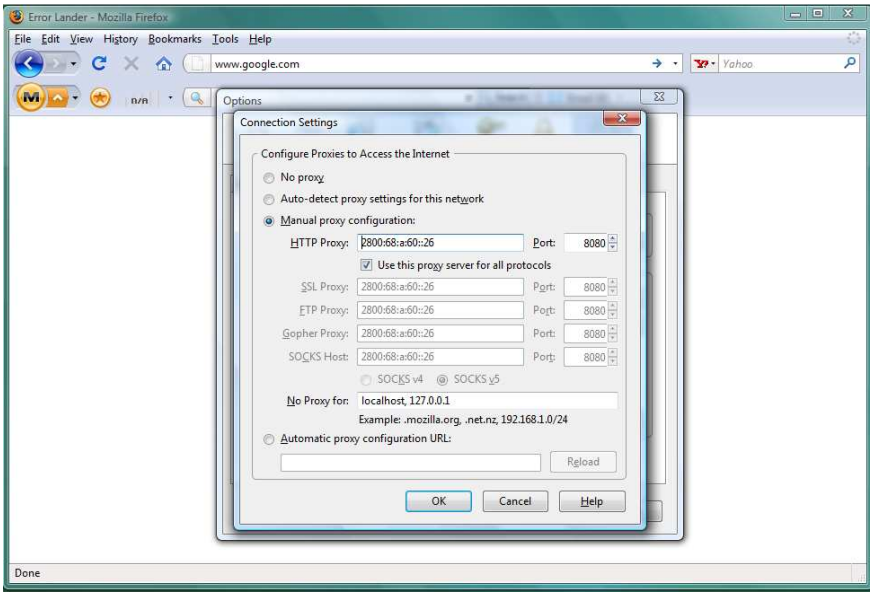

Figura III-105 Configuración del Proxy en Mozilla FireFox

Figura III-106 Accediendo al Web Site www.ipv6.org con Internet Explorer

Figura III-107 Accediendo al Web Site www.ipv6.org con FireFox

3.4 ANALISIS DE RENDIMIENTO

El análisis que se realizo fue tanto en los protocolos de IPv4 e IPv6 en los cuales se utilizo el parámetro de tiempo de respuesta, el mismo que es obtenido mediante el comando ping, que a continuación se detalla.

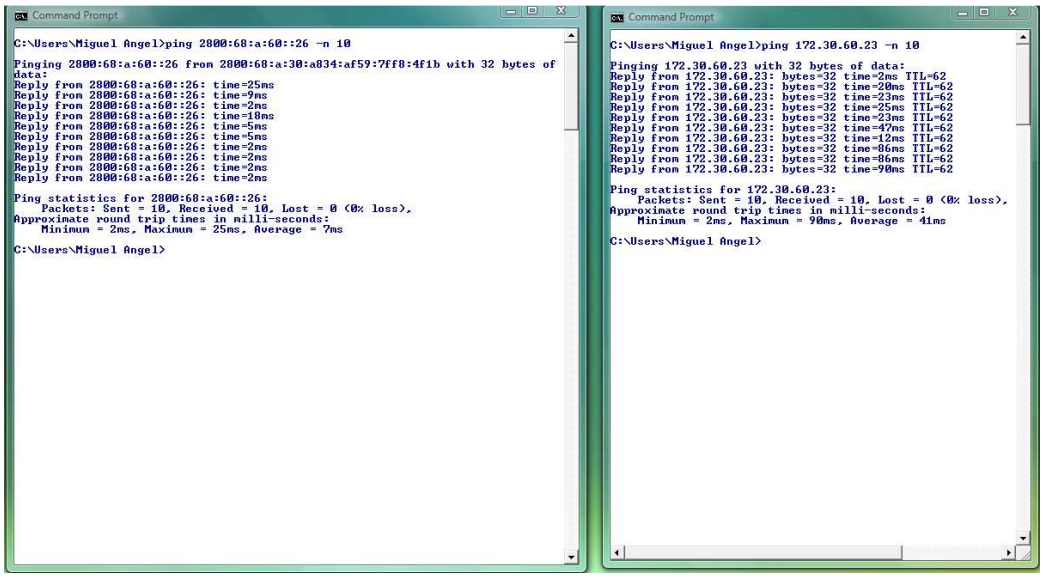

Figura III-108 Ping con 10 solicitudes de eco

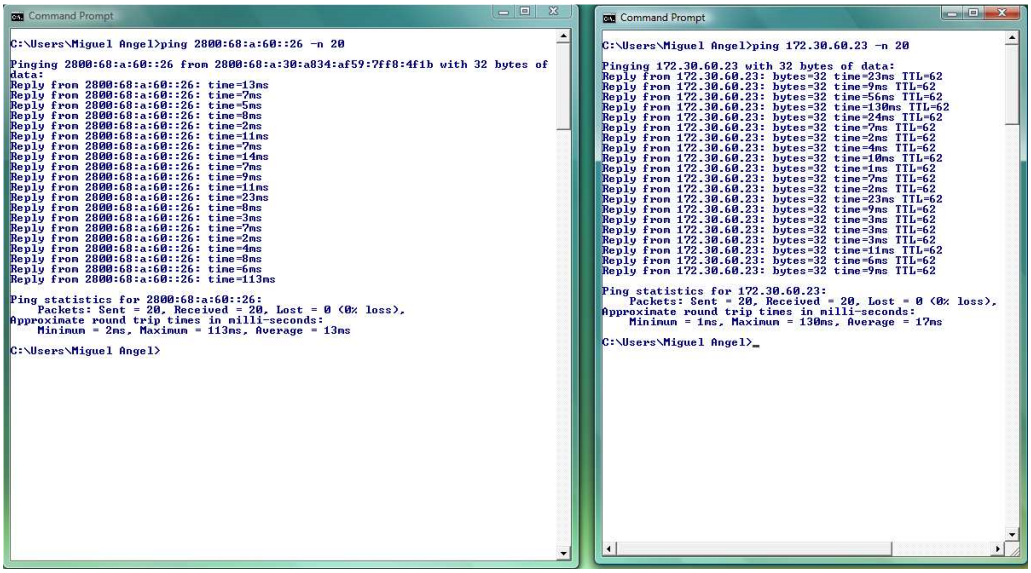

Figura III-109 Ping con 20 solicitudes de eco

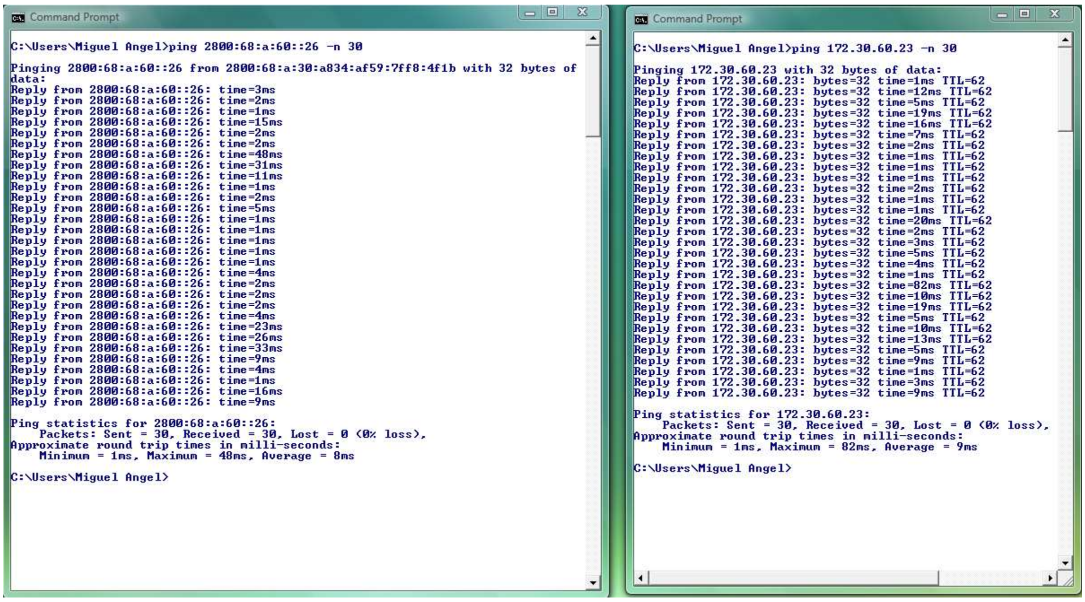

Figura III-110 Ping con 30 solicitudes de eco

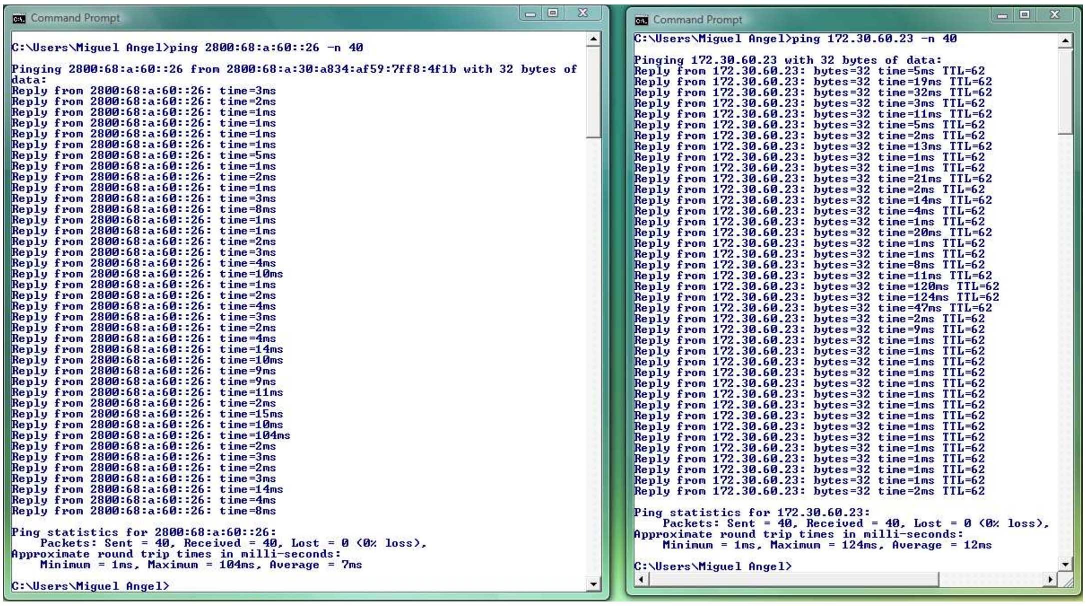

Figura III-111 Ping con 40 solicitudes de eco

Con la información obtenida anteriormente se pudo realizar un análisis comparativo entre los dos protocolos. Los tiempos de respuesta utilizados para el análisis son los tiempos promedios, los cuales nos ayudo a concluir que el protocolo IPv6 es más rápido respecto al protocolo IPv4.

Número de solicitudes	IPv6 [mseg]	IPv4 [mseg]
10	7	41
20	13	17
30	8	9
40	7	12

Tabla III-16 Análisis de IPV4/IPv6

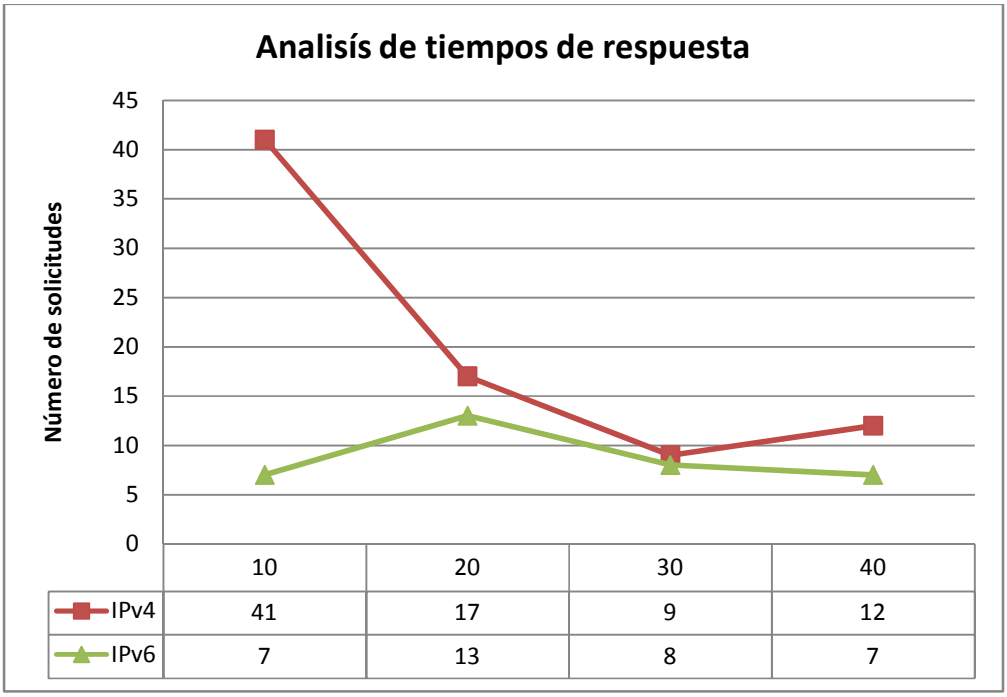

Figura III-112 Análisis Comparativo de IPv4/IPv6

3.5 RESULTADOS

Los servicios de Web, Mail, Ftp, Proxy, Dns y Dhcp, están implementados con el mecanismo de transición Dual Stack (Doble Pila), lo que permitirá que los usuarios puedan acceder a los servicios de la Intranet de la Escuela Superior Politécnica de Chimborazo indiferente a la versión del protocolo IP. El impacto del protocolo IPv6 será mínima, ya que los servicios será transparentes para la los usuarios.

CONCLUSIONES

- ✓ De acuerdo al análisis realizado entre los diferentes mecanismo de transición, se opto por seleccionar la implementación de Dual stack o doble Pila, debido a que no implica trabajar con dos redes paralelas, por lo que resulto el mecanismo más optimo y transparente para la ESPOCH.
- ✓ Los servicios de la Intranet de la Escuela Superior Politécnica de Chimborazo como son: Web, Mail, Ftp, Proxy, Dns y Dhcp fueron implementados con el mecanismo de transición DUAL STACK o Doble Pila.
- ✓ Los usuarios de la Intranet de la Escuela Superior Politécnica de Chimborazo podrá acceder al Internet Avanzado mediante IPv4 e IPv6.
- ✓ El tiempo de respuesta en IPv6 es mejor, debido a que la fragmentación se realiza en el nodo origen y el reensamblado se realiza en los nodos finales, y no en los routers como en IPv4.
- ✓ Lo más importante no es si las direcciones IPv4 se terminan antes o después, sino que IPv6 es una puerta a la innovación, a la integración de servicios y tecnologías, de manera que contribuya al desarrollo de nuestra sociedad.

RECOMENDACIONES

- ✓ Para la adaptación de las aplicaciones a IPv6 se recomienda utilizar nombres en vez de direcciones para identificar un nodo, ya que al utilizar nombres se evita la diferenciación en el formato de presentación para las direcciones IPv4 e IPv6, además que los nombres siguen el mismo formato en ambos protocolos y su resolución a direcciones IP se delegará en el servicio de DNS (Domain Name Server), de tal forma que la implementación de IPv6 sea lo más transparente posible para el usuario.
- ✓ Es recomendable utilizar en las redes de la ESPOCH dispositivos y aplicaciones que estén realmente listos para trabajar tanto con IPv4 como IPv6, de tal forma que estos sean aprovechados en todo su potencial.

RESUMEN

Se implementó Dual Stack (IPv4/IPv6) dentro de la Intranet de la ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO en los servicios Web, Mail, Ftp, Proxy, Dns y Dhcp, con la finalidad de diversificar los servicios y al mismo tiempo estar preparado para el manejo de este protocolo para las redes avanzadas.

La implementación se realizó mediante mecanismo de transición Dual Stack o Doble Pila, es decir que éste permite tener dos stacks: stack IPv4 e IPv6 en un host; lo que permitió que el host pueda tomar decisiones de cuándo se deban hacer las conexiones con IPv4 o IPv6; basándose en la disponibilidad de conectividad con IPv6 y los registros de sistema de nombres de dominios (DNS), que son completamente independientes.

Ipv6 ofrece mayor espacio de direcciones, cambia de 32 a 128 bits, para soportar mayores niveles de jerarquías de direccionamiento y nodos direccionables, con direcciones unicast, multicast y anycast, es decir que broadcast no existe; tiene un formato de cabecera más flexible que en IPv4 para agilitar el encaminamiento, se agregan nuevas características de seguridad IPSEC que forma parte del estándar; auto-configuración de los nodos finales, que permite a un equipo aprender automáticamente una dirección IPv6 al conectarse a la red; movilidad incluida en el estándar, facilitando el cambio de red sin perder conectividad; permite calidad de servicio (QoS) y clase de servicio (CoS).

Con la implementación del mecanismo de transición Dual Stack se logró que los servicios trabajen con IPv4/IPV6, sin crear impacto para los usuarios, debido a que trabaja de manera transparente.

SUMMARY

Dual Stack (IPv4/IPv6) was implemented inside the Intranet of the "ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO" with different services such as: Web, Mail, Ftp, Proxy, Dns and Dhcp, with the purpose of diversifying the services and at the same time to be prepared for the handling this protocol for advanced nets.

The implementation was carried out by means of a Dual Stack transition or Double Pile, mechanism, it means, Dual Stack allows to have two stacks: stack IPv4 and IPv6 in a host. The host can make decisions about when the connections should be made with IPv4 or IPv6; based on the connection readiness with IPv6 and the system records related to names of domains (DNS), due to Dual Stack are totally independent.

Ipv6 offers bigger addresses space, it changes 32 to 128 bits, to support more levels of addressed hierarchies and nodes, using unicast, multicast and anycast addresses, it means "broadcast" disappear. The Ipv6 has a format more flexible header than the IPv4, to activate the routing, it adds new security IPSEC characteristics, which is part of the standard; automatic configuration last nodes, the equipment will learn an IPv6 address automatically when being connected to the net; and also it will have mobility included in the standard, facilitating the net change without losing connection; it allows quality (QoS) and class (CO).of service.

With the implementation of Dual Stack transition mechanism, the intranet services were achieved, using IPv4 / IPV6, without creating impact for the users, due to Dual Stack works in a transparent way.

GLOSARIO

- ✓ FTP (File Transfer Protocol) :Protocolo Transferencias Archivos es un protocolo de red para la transferencia de archivos
- ✓ DNS: Domain Name Server es el servidor encargado de transformar los bonitos nombres que te aprendes en las direcciones IP que realmente les corresponden, de forma que cuando tu buscas una dirección en concreto tu ordenador contacta con tu DNS (que te proporciona tu ISP) y le pregunta por la IP si este no la conoce te manda que preguntes a su superior y este hará lo mismo hasta que lo encuentres o te des cuenta de que no existe ese ordenador, por eso los servidores están unidos jerárquicamente y cada cierto tiempo intercambian las modificaciones para tener las bases de datos lo mas actualizadas posibles.
- ✓ DAEMON (demonio): Son los servidores de UNIX y LINUX y pueden ser identificados porque su nombre termina siempre en d: httpd, smtpd, ftpd...
- ✓ DoS: Denial of Service, se trata de un ataque basado en conseguir que el ordenador se bloquee, esto se puede conseguir a través del conocimiento de algún BUG o simplemente en los servidores sobrepasando con creces el número de peticiones que puede soportar de forma que se bloquee, esta fue la técnica utilizada para los ataque a grandes portales a principios de año.
- ✓ FIREWALL (cortafuegos): Sistemas físicos o no que se encargan de seleccionar e impedir las conexiones que se realicen a un ordenador suele ser muy utilizados para proteger un servidor de ataque a puertos no autorizados aunque también se utiliza para impedir el acceso a una subred conectada a una red mientras la subred mantiene todas sus posibilidades dentro de la red.

- ✓ ISP: Internet Service Provider (Proveedor de servicios de Internet), es la entidad que se encarga de conectar a las personas que no pueden permitirse tener una conexión fija con Internet, y de proporcionarles todos los datos y servicios que deseen por un precio determinado o gratis.
- ✓ SCRIPT: Se llama script a todo conjunto de instrucciones que se guardan en un fichero y que son interpretadas tal y como están por un programa. En su época los archivos .bat fueron unos de los primeros script actualmente se utilizan en la web a través de Javascript o Jscript, los CGIs en Perl, las paginas en ASP y PHP3... y en el ordenador de casa subsisten los .bat y han aparecido miles para Windows, alguno de los nuevos virus como el I love you se programan en scripts de Windows. Además otros programas que soportan la automatización lo hacen a través de scripts como el mIRC, el Macromagic, el Automate...
- ✓ PARCHE: Es el archivo que realiza correcciones en un archivo ejecutable o en sus datos para eliminar BUGs Microsoft, cuando ha acumulado un gran número de ellos realiza un Service Pack de forma que elimina todos juntos puesto que la cantidad de parches que salen cada mes es inmensa y es muy complicado tener la absoluta certeza de haberlos instalado todos de forma que con los service pack se obtiene la seguridad de que todos los errores encontrados hasta la fecha de publicación están corregidos.
- ✓ PUERTO: Numero que identifica un socket de forma que todos los paquetes además de tener una dirección IP de destino tienen también un puerto de destino, el numero de puerto es fijo en el servidor y depende del servicio que preste (por ejemplo un servidor web esta en el puerto 80 siempre) y en el usuario depende de la conexión pues utiliza un número de puerto distinto para cada ordenador con el que se conecta.

- ✓ SERVIDOR: Programa encargado de la realización de una serie de tareas para toda una comunidad (una red o parte de ella) automáticamente a través de un determinado PUERTO.
- ✓ SOCKET (enchufe): Se trata del dispositivo lógico de la recepción y envío de datos en una red, todos los dispositivos físicos de un ordenador tienen una dirección lógica a donde enviar los datos y de donde se esperan las respuestas de la misma manera se utiliza un conjunto de dispositivos lógicos para controlar las conexiones con otros ordenadores como si fueran otros periféricos.
- ✓ PROXY: Ordenador encargado compartir internet y de guardar copias de los archivos más solicitados para reducir el tráfico de la red e impedir su colapso, se trata de caches de red.
- ✓ IPV4: Versión 4 del protocolo IP (Internet Protocol). Es el estándar actual de Internet para identificar dispositivos conectados a esta red.
- ✓ IPV6: El protocolo **IPv6** es una nueva versión de IP (Internet Prococol), diseñada para reemplazar a la versión 4 (IPv4)
- ✓ STACK: es una lista ordinal o estructura de datos en la que el modo de acceso a sus elementos
- ✓ QoS: Calidad de Servicio (Quality of Service, en inglés) son las tecnologías que garantizan la transmisión de cierta cantidad de datos en un tiempo dado (throughput). Calidad de servicio es la capacidad de dar un buen servicio. Es especialmente importante para ciertas aplicaciones tales como la transmisión de video o voz

BIBLIOGRAFIA

- ✓ HINDEN, R. and DEERING, S. Internet Protocol, Version 6 (IPv6) Specification

  http://www.ietf.org/rfc/rfc2460.txt

  199812

- ✓ THOMSON, S. et al. IPv6 Stateless Address Autoconfiguration

  http://www.ietf.org/rfc/rfc2462.txt

  199812

- ✓ LUCENT CONTA, A. and DEERING, S. Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6)

  http://www.ietf.org/rfc/rfc2463.txt

  199812

- ✓ J. MCCANN, et al. Path MTU Discovery for IP version 6

  http://www.ietf.org/rfc/rfc1981.txt

  199608

- ✓ PARTRIDGE, C. et al. Using the Flow Label Field in IPv6

  http://www.ietf.org/rfc/rfc1809.txt

  199506

- ✓ HINDEN, R. et al. IP Version 6 Addressing Architecture

  http://www.ietf.org/rfc/rfc2373.txt

  199807

- ✓ REKHTER, Y. et al. An Architecture for IPv6 Unicast Address Allocation

  http://www.ietf.org/rfc/rfc1887.txt

  199512

- ✓ HINDEN, R. et al. An IPv6 Aggregatable Global Unicast Address Format”

http://www.ietf.org/rfc/rfc2374.txt

199807

- ✓ S. THOMSON, et al. DNS Extensions to support IP version 6

  http://www.ietf.org/rfc/rfc1886.txt

  199512

- ✓ CRAWFORD, M. et al. Transmission of IPv6 Packets over Ethernet Networks

  http://www.ietf.org/rfc/rfc2464.txt

  199812

- ✓ TSIRTSIS, G. and SRISURESH, P. Network Address Translation - Protocol Translation (NAT-PT)

  http://www.ietf.org/rfc/rfc2766.txt

  200002

- ✓ K. TSUCHIYA, et al. Dual Stack Hosts using the Bump-In-the-Stack

  http://www.ietf.org/rfc/rfc2767.txt

  199812

- ✓ IAB AND IESG, IPv6 Address Allocation Management

  http://www.ietf.org/rfc/rfc1881.txt

  199512

- ✓ ALLMAN, M. et al. FTP Extensions for IPv6 and NATs

  http://www.ietf.org/rfc/rfc2428.txt

  199809

- ✓ NICHOLS, K. et al. Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers

  http://www.ietf.org/rfc/rfc2463.txt

199812

- ✓ Srisuresh, P. et al. IP Network Address Translator (NAT) Terminology and Considerations

  http://www.ietf.org/rfc/rfc2663.txt

  199908

- ✓ ELZ, R. UNIVERSITY OF MELBOURNE A Compact Representation of IPv6 Addresses"

  http://www.ietf.org/rfc/rfc1924.txt

  19960401

- ✓ Martínez Palet Jordi

  http://bjcu.uca.edu.ni/LibrosIsti/Tutorial_de_IPV6.pdf

- ✓ DUEÑAS BARRIOS JOEL, Implementación De Servidores Con GNU/Linux

  www.linuxparatodos.org

  200706

- ✓ Red Hat Manuals

  https://www.redhat.com/

- ✓ CentOS-5 Documentation

  http://www.centos.org/docs/5/

- ✓ WIKIPEDIA

  http://es.wikipedia.org/wiki/IPv6

- ✓ Understanding IP Addressing

www.3com.com/other/pdfs/infra/corpinfo/en_US/501302.pdf

- ✓ Tutorial de **IPv6**

  http://www.ipv6.unam.mx/documentos/Tutorial-IPv6.pdf

- ✓ Linux IPv6 HOWTO (en)

  http://tldp.org/HOWTO/Linux+IPv6-HOWTO/

- ✓ Instalación de IPV6 en plataformas Windows

  http://www.6sos.net/documentos/6SOS_Instalacion_IPv6_Windows_v4_0.pdf

- ✓ Instalación de **IPv6** en plataformas **Linux**

  http://www.6sos.net/documentos/6SOS_Instalacion_IPv6_Linux_v4_0.pdf

- ✓ Configuring IPv6 addresses

  http://www.linux.com/base/ldp/howto/Linux+IPv6-HOWTO/chapter-configuration-address.html

- ✓ IPv6 configuration on Linux operating systems

  http://forskningsnett.uninett.no/ipv6/IPv6hostslinux.html

- ✓ Diferencias entre DHCPv4 y DHCPv6

  http://docs.sun.com/app/docs/doc/820-2981/gefkw?l=es&a=view

- ✓ dhcp6s.conf(5) - Linux man page

  http://linux.die.net/man/5/dhcp6s.conf