UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Actualización de IOS (Sistema Operativo) de equipos de acceso en sitios remotos

TESIS PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN TELECOMUNICACIONES

P R E S E N T A :

CARLOS JIMÉNEZ HERRADA

ASESOR:
DR. VICTOR RANGEL LICEA

2013

Dedicatorias

A mi familia por siempre apoyarme durante todo mi ciclo escolar hasta poder culminar con mi carrera profesional, un logro que comparto con cada uno de mis seres queridos.

A todos mis maestros que con sus conocimientos dentro y fuera del aula me motivaron para poder terminar mis estudios.

A la Universidad Nacional Autónoma de México, y especialmente a la Facultad de Ingeniería por darme las mejores lecciones de vida para poder desarrollarme en el ámbito profesional y personal.

Agradecimientos

A la DGAPA UNAM por el apoyo recibido en el proyecto de investigación PAPIIT: IN114713 “Diseño y análisis de algoritmos de calendarización en redes LTE y WiMAX”.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por el apoyo recibido en el proyecto CONACYT 105279 “Diseño de técnicas de reservación de capacidad en redes inalámbricas de banda ancha móviles”.

Índice General

Índice de Figuras

Capítulo I

Introducción

1.1 Instituto Federal Electoral

El IFE (Instituto Federal Electoral)[1] es el organismo público autónomo responsable de cumplir con la función del Estado de organizar las elecciones federales, es decir las referentes a la elección de Presidente de la República y de los Diputados y Senadores que integran el Congreso de la Unión. [1]

El Instituto Federal Electoral es un organismo de carácter permanente e independiente en sus decisiones y funcionamiento, cuenta con personalidad jurídica y patrimonio propios.

En su integración participan el Poder Legislativo, Partidos Políticos Nacionales y Ciudadanos.

Para el desempeño de sus funciones, cuenta con un cuerpo de funcionarios integrados en un Servicio Profesional Electoral y en una rama administrativa.

Su sede central está ubicada en el Distrito Federal y ejerce sus funciones en todo el territorio nacional con 32 delegaciones, una en cada Estado y 300 subdelegaciones, una en cada distrito electoral uninominal.

[1]En la página 64 se agrega el glosario de términos.

El Instituto Federal Electoral tiene a su cargo en forma integral y directa todas las actividades relacionadas con la preparación, organización y conducción de los procesos electorales federales, así como aquellas que resultan consecuentes con los fines que la ley le fija. [1]

Entre sus actividades fundamentales se pueden mencionar las siguientes:

- Capacitación y educación cívica.
- Geografía electoral.
- Derechos y prerrogativas de los partidos y agrupaciones políticas.
- Padrón y listas de electores.
- Diseño, impresión y distribución de materiales electorales.
- Preparación de la jornada electoral.
- Cómputo de resultados.
- Declaración de validez y otorgamiento de constancias en la elección de diputados y senadores.
- Regulación de la observación electoral y de las encuestas y sondeos de opinión.
- Administración del tiempo que corresponde al Estado en radio y televisión.

Dentro de la estructura del Instituto existe una unidad en materia de informática, tiene el nombre de Unidad de Servicios de Informática (UNICOM) la cual tiene dentro de sus actividades los siguientes rubros:

- Automatizar procesos que permitan eficientar el desarrollo de las tareas que deben cumplir las diversas áreas del Instituto, a través del desarrollo, implementación y actualización de sistemas y servicios informáticos.

- Administrar y operar la infraestructura de cómputo y comunicaciones que soporta los servicios y sistemas que se encuentran disponibles a través de la Red Nacional de Informática, la cual permite comunicar a todo el personal que labora en el

Instituto a nivel nacional, así como difundir información a la ciudadanía.

- Coordinar y ejecutar el desarrollo y operación del Programa de Resultados Electorales Preliminares PREP, el cual permite difundir de forma inmediata a la ciudadanía, los resultados preliminares de las elecciones federales.

- Proporcionar capacitación, asesoría y soporte técnico a todo el personal del Instituto, en el uso de sistemas, servicios y equipos de cómputo. [2]

El área donde desempeño mis funciones es UNICOM particularmente en la Subdirección de Comunicaciones que es la encargada de la administración de la red del IFE a nivel nacional tanto en datos, voz y video.

1.2 Redes de datos

Las redes han evolucionado a lo largo de los últimos años, al principio se refería el término "redes" a actividades como correos electrónicos, navegar por internet o usar los servicios de mensajería instantánea. Hoy en día las redes han evolucionado con enorme velocidad, ahora no sólo brindan los servicios básicos como la compartición de datos y de impresoras, ahora también soportan aplicaciones tan robustas como la videoconferencia. En estos tiempos es difícil encontrar que alguien únicamente quiera compartir información dentro de la misma área de oficina, el reto es conectar varias redes entre sí para que se pueda compartir información relevante de una organización a través de la red.

En la Figura 1.1, se observa una red de LAN (*Local Area Network*) básica que es interconectada por medio de un *Hub*. Esta red tiene un dominio de colisión y un dominio de *broadcast*.

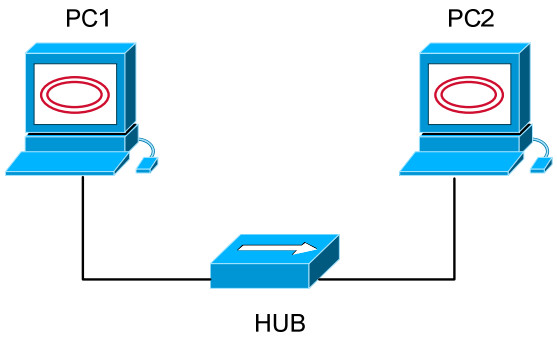


Figura 1.1: Red básica.

El dominio de colisión se refiere a que en este segmento de red los paquetes pueden colisionar o chocar entre sí, esto puede provocar pérdida de paquetes y por consiguiente una red de baja calidad.

El dominio de broadcast es el segmento de red donde todos los disposivitos conectados a ese segmento pueden recibir un *broadcast*. El *router* es el dispositivo que limita los dominios de *broadcast*.

La red de la Figura 1.1 es relativamente sencilla y sólo es funcional para redes pequeñas ya que es propensa a colisiones. Para evitar las fallas en la red conforme ésta crezca, esta red puede partirse o segmentarse. Esto se logra utilizando dispositivos tales como *switches*, *routers* y *bridges*. La Figura 1.2 muestra una red que ha sido segmentada con un *switch*, en cada puerto de *switch* se tiene un solo dominio de colisión.

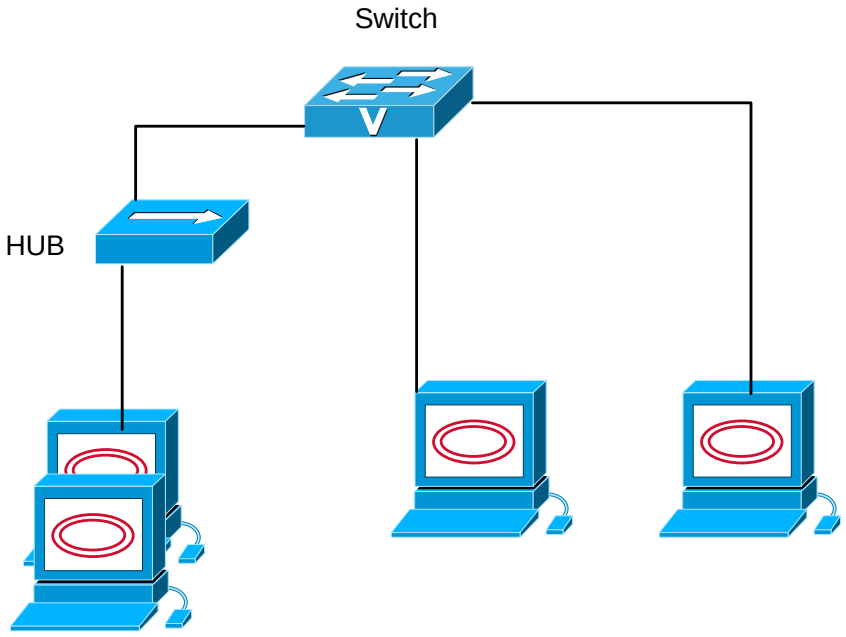


Figura 1.2: Red segmentada

Aun cuando la red ha mejorado sigue existiendo un solo dominio de *broadcast* por lo que la red puede sufrir de tormentas de *broadcast*, bajo ancho de banda o colisiones al agregar *hubs* en la topología. El agregar un *switch* hizo la red más eficiente comparado con la Figura 1.1, entendiendo que el agregar este dispositivo a la red supone mayor costo ya que los *switches* tienen un costo considerablemente mayor que los *hubs*.

Para poder romper o eliminar un dominio de *broadcast* se necesita un *router*, el cual es un dispositivo que proporciona un direccionamiento lógico y provee lo que se conoce como *switcheo* de paquetes. Los *routers* también tienen la funcionalidad de filtrar paquetes utilizando listas de

acceso, y cuando los *routers* conectan dos o más redes usando el direccionamiento lógico (Ipv4 o Ipv6) a esto se le llama interconexión de redes.

Los *switches* no fueron creados para la interconexión de las redes, su principal función es hacer que una red LAN funcione adecuadamente optimando su desempeño por medio de funcionalidades propias del dispositivo. Por default, un *switch* crea varios dominios de colisión pero un solo dominio de *broadcast*. Los *routers* crean un dominio de *broadcast* por cada interfaz del dispositivo.

La Figura 1.3 muestra una red más estable y funcional. Cada *host* está conectado a su propio dominio de colisión, y el *router* ha creado dos dominios de *broadcast*, además de dar conectividad a los servicios WAN (*Wide Area Network*).

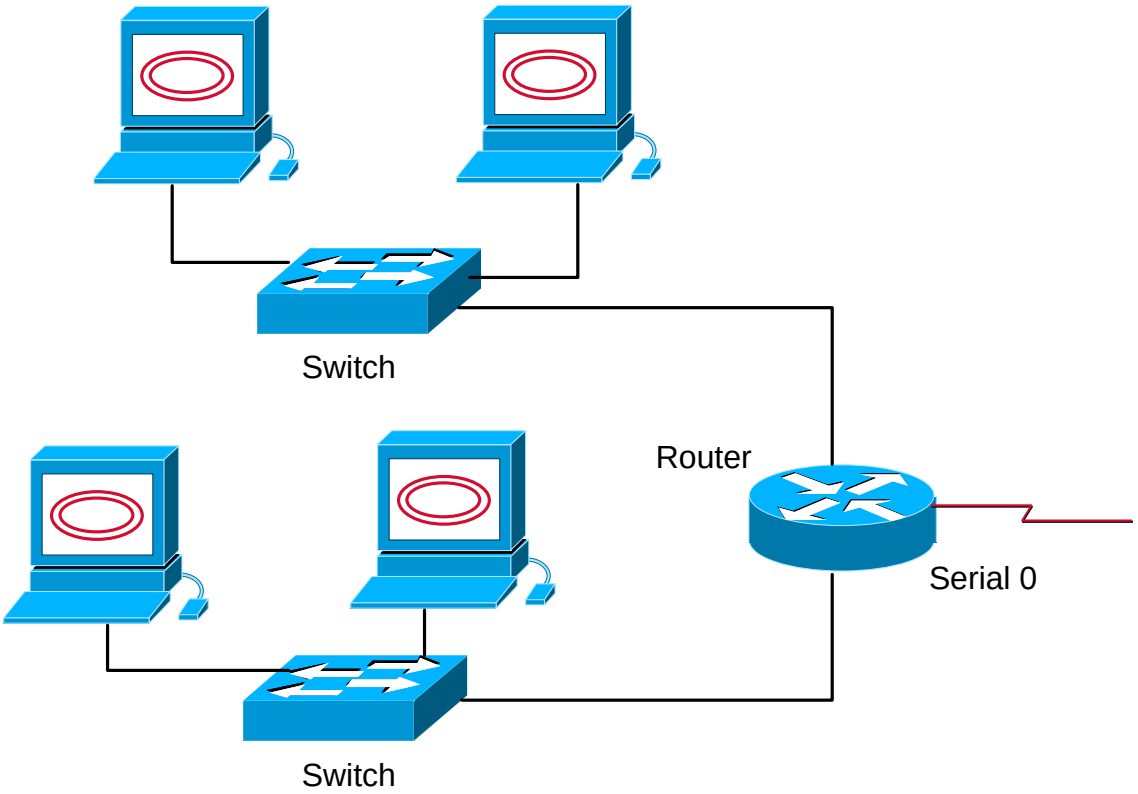


Figura 1.3: Red con dos dominios de broadcast.

Romper o separar los dominios de *broadcast* es importante ya que cuando un servidor o un *host* envían un mensaje de *broadcast*, todos los dispositivos en la red deben de leer y procesar estos paquetes aun cuando éstos finalmente sean descartados por el equipo, para poder evitar estas tormentas de *broadcast* se utiliza un *router*.

Entre las ventajas que se tiene al utilizar *routers* en una red está el que por default no reenvía *broadcast* a otras redes y que a diferencia del *switch* que envía datos utilizando la dirección mac, el *router* lo hace utilizando direcciones lógicas.

Los *routers* son en realidad *switches*, se les puede llamar *switches* de capa 3. Los *switches* convencionales actúan en capa 2, aunque existen *switches* que actúan ya sea en capa 2 y/o capa 3. [3]

1.3 Definición del problema

El Instituto Federal Electoral tiene varios Órganos Delegacionales que se encuentran distribuidos a nivel nacional, dentro de cada uno de ellos es imprescindible que existan una conexión a la red nacional del Instituto.

En cada Órgano Delegacional se tiene un enlace a través de un proveedor de servicios que interconecta a la red local con la red central del Instituto, dentro de la red local se utilizan equipos de acceso de la marca Cisco. Tanto los equipos de acceso como los equipos de capa 3 son administrados vía remota desde las Oficinas Centrales ubicadas en el Distrito Federal.

La Subdirección de Comunicaciones, área a la que yo pertenezco, tiene la administración de la red nacional y por consiguiente de los equipos que la conforman. Como parte del robustecimiento de la red se propuso una mejora en todos los equipos de acceso a nivel nacional.

La conexión remota de los equipos de acceso se realizaba anteriormente mediante el protocolo Telnet, el cual es un protocolo de red que tiene como función acceder a un equipo remotamente mediante línea de comandos. El mayor problema con este protocolo es la seguridad, ya que los nombres de usuarios y la contraseña que se utilizan para acceder a un equipo remoto (ya sea un equipo Linux, cisco, etc.) son enviados a través de la red como *texto plano*, esto implica que fácilmente, estos datos críticos pueden ser obtenidos por alguien que espíe el tráfico en la red, por lo que es una vulnerabilidad que debe de eliminarse hoy en día.

Se planteó cambiar la forma de acceso a dichos equipos utilizando el protocolo SSH (*Secure Shell*), este protocolo funciona de forma similar al Telnet con una gran diferencia, utiliza técnicas de cifrado de datos que permite que la información que viaja por el medio no esté legible, de esta manera la información del usuario, contraseña o lo que se escribe durante la sesión no pueda ser descubierta de una forma sencilla.

Para poder configurar este protocolo en los equipos de acceso se requiere validar si el sistema operativo con el que cuentan tiene la capacidad para

soportarlo. Debido a que los equipos de acceso no contaban con la versión de software necesaria para la implementación de SSH como método de acceso remoto, se encontró la necesidad de actualizar la versión del sistema operativo IOS (*Internetwork Operating System*) en todos los equipos de acceso.

Para la actualización del sistema operativo se necesita encontrar las versiones de IOS que soporten la funcionalidad de SSH, se requiere validar que la memoria *flash* tenga la capacidad para albergar la nueva imagen IOS, que la memoria RAM sea la mínima requerida y el procedimiento más seguro para realizar la sustitución del IOS.

El procedimiento a utilizar para cada modelo de equipo de acceso puede variar, un error durante la actualización puede provocar que el equipo deje de funcionar. Dado que las actualizaciones deberán realizarse de forma remota también debe de considerarse procedimientos de contingencia, los cuales nos ayudarán a responder a cualquier problema que pudiera presentarse.

1.4 Objetivo

Actualizar el sistema operativo (IOS) en equipos de acceso que se encuentren en los diferentes Órganos Delegacionales que tiene el Instituto Federal Electoral a nivel nacional y, que debido a su importancia, se requiere que la afectación en el acceso a red en dichos sitios sea el mínimo.

1.5 Estructura de tesis

La tesis se encuentra dividida en siete capítulos, el primero se encarga de explicar de manera resumida lo que es el IFE y su función como Instituto Electoral; una breve introducción de las redes de datos de manera general, la definición del problema y el objetivo del presente trabajo.

El segundo capítulo abordara el modelo de referencia OSI, la topología de red de manera general con la que se trabaja en el Instituto, y una breve explicación de la forma en que se lleva a cabo la conectividad entre los diferentes inmuebles del IFE.

El tercer capítulo presenta los componentes que conforman un equipo Cisco, el sistema operativo (IOS), y el proceso de arranque y su verificación, esto último parte importante debido a que podemos detectar si el IOS arrancó correctamente o algo ha fallado.

El cuarto capítulo presenta los modelos de equipos a intervenir así como la verificación de la memoria de los mismos, esto para poder definir qué procedimiento aplicar. Así mismo se justifica el uso de FTP (*File Transfer Protocol*) en este proyecto y el uso de los scripts.

El quinto capítulo detallará los dos procedimientos de actualización que se utilizaron en este proyecto, desde cómo determinar qué procedimiento usar, hasta la explicación de los *scripts* utilizados para la automatización de dichas actualizaciones. Así mismo en este capítulo se presenta el procedimiento de contingencia, el cual nos ayudará en dado caso de fallar el procedimiento de actualización seleccionado.

El quinto capítulo presentará los resultados obtenidos, y la configuración que se tiene que aplicar para poder habilitar el protocolo SSH, que fue el objetivo de este proyecto.

El sexto capítulo contiene las conclusiones sobre el trabajo realizado.

Capítulo II

Topología de red y protocolos

Es importante conocer la topología de red en una organización para poder realizar alguna mejora no sólo de sistema operativo, sino también mejoras en la velocidad, eficiencia, redundancia, etc.

En este capítulo se presenta la topología de red del Instituto y los protocolos que intervienen para que los diferentes sitios se encuentren interconectados entre sí. Así mismo es importante explicar de manera resumida el modelo de referencia OSI el cual es un modelo base para las comunicaciones hoy en día.

2.1 Modelo de referencia OSI

Cuando las primeras redes comenzaron a operar los equipos de cómputo de diferentes marcas no se podían comunicar entre sí, ya que cada fabricante de equipo seguía sus propios estándares y métodos para la comunicación entre sus equipos. Debido a esta problemática se creó el modelo OSI (*Open Systems Interconnection*), para que los fabricantes siguieran determinados métodos para fabricar dispositivos de red y software, para que los equipos sin importar su fabricante pudieran comunicarse entre sí.

El modelo de referencia OSI describe como los datos en una red pasan de un equipo a otro a través de capas específicas.

Un *modelo de referencia* es un modelo conceptual de cómo debe darse la comunicación. Tiene todos los procesos necesarios para llevar a cabo una

comunicación efectiva y divide esos procesos en grupos lógicos llamados *capas*. Cuando un sistema de comunicación está diseñado de esta manera, es conocido como una *arquitectura en capas*.

El modelo OSI es jerárquico, y los mismos beneficios y ventajas pueden aplicar a cualquier modelo en capas. El principal propósito de todos estos modelos, especialmente el modelo OSI, es permitir a diferentes fabricantes de dispositivos de red interoperar entre sí.

Una de las mejores funciones de las especificaciones del modelo OSI es la capacidad de asistir en la transferencia de datos entre *host* disparejos, por ejemplo, una transferencia de datos entre un *host* Linux y un Windows o Mac.

El modelo OSI no es un modelo físico, es más bien una serie de directrices que los desarrolladores de aplicaciones pueden utilizar para crear e implementar aplicaciones que funcionan en la red. También proporciona una estructura para crear e implementar estándares de redes, dispositivos y esquemas de interconexión de red.

El modelo OSI tiene siete diferentes capas, divididas en dos grupos. Las tres capas superiores definen como las aplicaciones dentro de los equipos de los usuarios finales se comunicarán entre sí y con los usuarios. Las restantes cuatro capas inferiores definen como los datos son transmitidos de punto a punto. La Figura 2.1 muestra las funciones de las tres capas superiores, y la Figura 2.2 muestra las funciones de las cuatro capas inferiores.

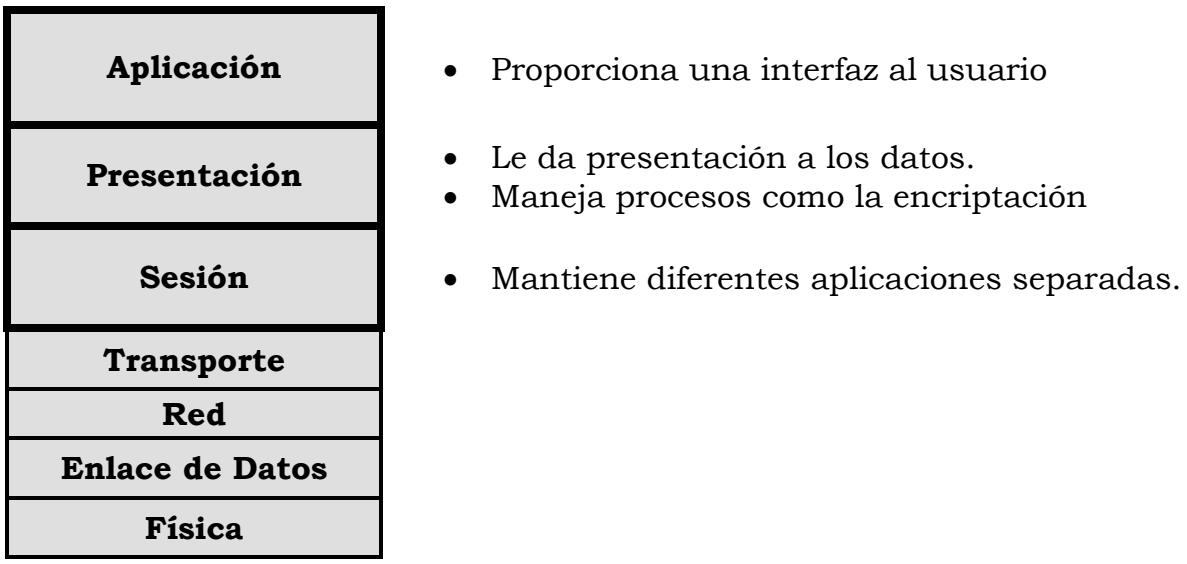
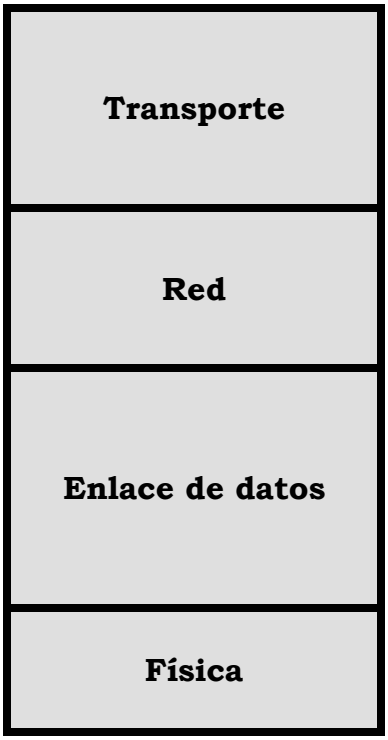


Figura 2.1: Modelo OSI parte I.

Al revisar la Figura 2.1, se comprende que el usuario interactúa con la computadora en la capa de Aplicación y también que las capas superiores son responsables de la comunicación de aplicaciones entre *hosts*. Las capas superiores desconocen sobre como la información llega de un punto a otro, de esto se encargan las capas inferiores.

En la Figura 2.2 muestra como las cuatro capas inferiores definen como los datos son enviados a través de un medio, ya sea un cable físico o por medio de *switches* y *routers*. Estas capas también determinan como reconstruir un paquete de un *host* emisor hacia alguna aplicación en el host receptor. [4]



- Proporciona envío de información usando protocolos orientados a conexión y protocolos no orientados a conexión.
- Proporciona corrección de errores antes de retransmitir.
- Proporciona direccionamiento lógico, que utilizan los routers para la determinación de la ruta más adecuada para la información.
- Combina paquetes en bytes y bytes en tramas.
- Proporciona el acceso al medio usando direcciones MAC.
- Realiza detección de errores mas no corrección.
- Transporta bits entre dispositivos.
- Especifica voltajes, velocidad en el medio y la configuración de los cables.

Figura 2.2: Modelo OSI parte II.

2.2 Topología de red

Estamos acostumbrados a visualizar una red de datos como la comunicación que existe entre dos computadoras solamente, sin conocer todo lo que implica que ambos equipos se puedan comunicar por medio de la red.

En el capítulo anterior se mostraron algunos diseños de red sencillas las cuales implicaban la conectividad de los equipos por medio únicamente de una red LAN, hasta ahí es fácil entender la forma en que se comunican

debido a que solo un equipo intermedio los separa (en ese caso fue un *switch* o un *hub*), pero ahora si la red crece y en ella intervienen equipos de capa 3 (como *switches* de capa 3 o *routers*) la complejidad para entenderla o para poder configurarla crece. A continuación se muestra una parte de la topología de red del Instituto y se explicarán algunos de los puntos más importantes de la misma, esto nos ayudará a entender cómo funciona una Red Nacional.

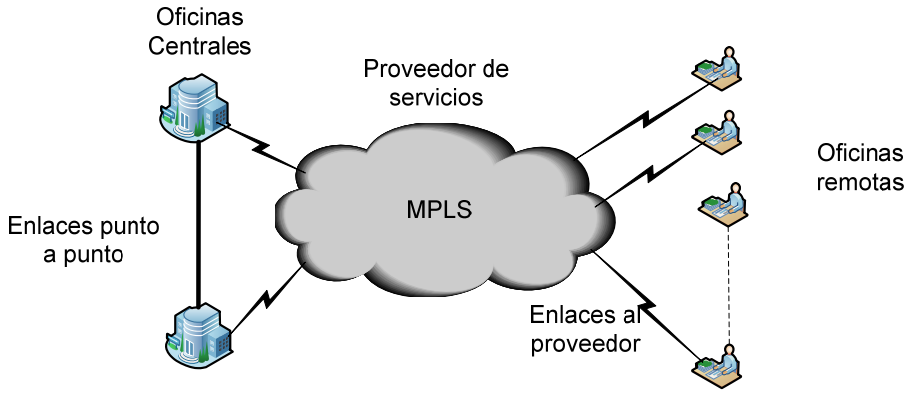


Figura 2.3: Topología del Instituto

En la Figura 2.3 se muestra un segmento de la topología del IFE, en este diagrama se muestran varias oficinas a nivel nacional interconectadas entre sí. Como se puede observar existen diferentes tipos de oficinas y de enlaces para su conexión.

Para edificios o inmuebles de oficinas centrales se utilizan enlaces punto a punto, donde la comunicación entre ambos lados es directa, la configuración y administración de este tipo de conectividad es sencilla, siempre y cuando no se abuse de un número muy grande de conexiones de este tipo.

Para las diversas oficinas o sedes remotas con las que se cuenta se utiliza una conexión a un proveedor de servicios de telecomunicaciones, este proveedor también tiene conectividad con las demás oficinas del Instituto y se encarga de distribuir todas las redes entre sí para que exista conectividad desde cualquier punto, y no existan sitios aislados.

La conectividad de red no es propia del Instituto, es decir, aun para los enlaces punto a punto no existe un enlace de fibra óptica o enlace de cobre que sea propiedad del IFE, toda la conectividad es a través de un proveedor de servicios debido a lo costoso de tener este tipo de infraestructura y, también por la naturaliza del IFE en donde algunas oficinas se cambian de domicilio cada determinado tiempo.

Para que toda la red pueda estar interconectada entre sí se utilizan protocolos de ruteo, estos protocoles permiten encontrar un ruta para hacer llegar el paquete a su destino. Estos protocolos con la debida configuración permiten que la red se distribuya y se conozca en todos los equipos interconectados del Instituto, de esta manera todos los segmentos de red pueden ser alcanzados desde un punto central de administración.

2.3 Protocoles de ruteo.

Un protocolo de ruteo dirige un paquete al destino requerido mediante una referencia de rutas adquiridas por toda la red, donde dicho protocolo se encuentra en funcionamiento.

Para cierto tipo de redes, normalmente pequeñas, no es necesario utilizar un protocolo de ruteo, se pueden configurar en los equipos de capa 3 rutas estáticas, en donde explícitamente se le indica al equipo como debe tratar los paquetes dependiendo del destino indicado. Pero cuando la red se hace más grande y compleja la utilización de rutas estáticas se debe reducir al mínimo o utilizarlas únicamente para agilizar o para hacer más eficiente la red, pero se recomienda utilizar uno o varios protocolos de ruteo para poder tener toda la organización conectada entre sí.

Dentro de su alcance se tienen dos grupos de protocolos de ruteo IGP (*Interior Gateway Protocol*) y EGP (*Exterior Gateway Protocol*). Para poder definirlos se requiere conocer el significado de Sistema Autónomo.

2.4 Sistema Autónomo (IGP y EGP)

Un Sistema Autónomo es un conjunto de ruteadores y de redes IP que se encuentran bajo una administración en común, respetando las mismas políticas y reglas. Entendiendo este concepto los protocolos IGP son aquellos que funcionan dentro de un Sistema Autónomo, y los EGP son aquellos que interconectan Sistemas Autónomos entre sí. [5]

En el IFE se utilizan tanto IGP como EGP para la conectividad con todos los sitios, dentro del Instituto para comunicar a las redes internas se utiliza un IGP, y parte de la conectividad con el Proveedor de Servicios es a través de un EGP, esto garantiza que todos los sitios se encuentren interconectados.

Gracias a la red nacional del IFE se puede realizar una administración centralizada de los equipos de comunicaciones para fines de monitoreo y mejoras a la red.

Capítulo III.

Características de equipos

Para actualizar equipos de acceso de forma remota se deben de tener todos los elementos necesarios para garantizar que durante el procedimiento de actualización, las fallas o errores sean los mínimos

En este capítulo se explicará los componentes de los equipos, el proceso de arranque de los mismos que es importante para saber si la imagen se carga correctamente o no, si existe algún problema después de la actualización y saber que IOS está ejecutándose en los equipos.

3.1 Componentes de un equipo Cisco (Memoria, CPU, IOS)

Los componentes de un *router* o *switch* son muy similares, como cualquier PC, un equipo cisco contiene los siguientes elementos:

- Unidad Central de Procesos (CPU).
- *Random-Access Memory* (RAM).
- *Read-Only Memory* (ROM).

CPU

El CPU ejecuta las instrucciones del sistema operativo, tales como la inicialización del sistema, funciones de *ruteo* y funciones de *switcheo*.

RAM

La memoria RAM guarda las instrucciones y la información necesaria para ser ejecutada por el CPU. Esta memoria es usada para guardar los siguientes componentes:

- **Sistema Operativo.** El IOS de Cisco (*Internetwork Operating System*) es copiada en la RAM durante el arranque del equipo.

- **El archivo de configuración.** Este es el archivo de configuración que guarda los comandos de configuración que el IOS del *router* o *switch* está usando en ese momento. Con algunas excepciones, todos los comandos configurados en el equipo son guardados en el archivo de configuración, conocido como "*running-config*".

- **Tabla de ruteo.** Este archivo guarda la información acerca de redes conectadas directamente o remotas. Es usada para determinar la mejor ruta para transmitir el paquete.

- **ARP cache.** Este cache contiene el mapeo de direcciones IP con direcciones MAC, es similar al *ARP cache* en una PC.

- **Buffer de paquetes.** Los paquetes son temporalmente guardados en un búfer cuando son recibidos en una interfaz o después de ser enviados por una interfaz.

La memoria RAM es volátil y pierde todo su contenido cuando el equipo es reiniciado o apagado. Sin embargo, el equipo posee algunas áreas de almacenamiento permanente tales como son la ROM, flash y NVRAM.

ROM

La memoria ROM es una forma de almacenaje permanente. Los dispositivos Cisco utilizan la ROM para almacenar lo siguiente:

- Las instrucciones de *bootstrap*.
- Software de diagnóstico básico.
- Versiones más primitivas del sistema operativo (IOS).

La memoria ROM utiliza *firmware*, que es un *software* que está incrustado o instalado en el circuito integrado. El *firmware* incluye el *software* que generalmente no necesita ser modificado o actualizado, como son las instrucciones de *bootstrap*. La memoria ROM no pierde su contenido cuando el equipo es reiniciado o apagado.

Memoria Flash

La memoria *flash* es la memoria no volátil de la computadora que puede ser eléctricamente guardada o borrada. La memoria *flash* es usada para el almacenaje permanente del sistema operativo, Cisco IOS. En casi todos los modelos Cisco, el IOS está permanentemente guardado en la memoria flash y copiada en la RAM durante el arranque del equipo, donde es ejecutada por el CPU. Algunos modelos obsoletos de equipo Cisco el IOS está ejecutándose directamente desde la memoria *flash*. La memoria *flash* consiste en tarjetas SIMMs o PCMCIA, las cuales pueden ser actualizadas para aumentar el tamaño de la memoria *flash*.

NVRAM

La memoria NVRAM (*Nonvolatile RAM*) no se pierde cuando se apaga el equipo. Esto es contradictorio con la mayoría de tipos de memoria RAM, como la DRAM, que requiere continuamente energía para poder mantener la información. La NVRAM es utilizada por el Cisco IOS como un almacenaje permanente para el archivo de configuración inicial (*startup-config*). Todos los cambios de configuración son guardados en el archivo *running-config* en la RAM, y con algunas excepciones, son implementadas inmediatamente por el IOS. Para salvar esos cambios en caso de que el equipo se reinicie o sea apagado, el archivo *running-config* debe ser copiado a la memoria NVRAM, donde es guardada con el nombre de archivo *startup-config*. La memoria NVRAM mantiene dicho archivo aun cuando el equipo es reiniciado o se apague.

3.2 Internetwork Operating System (IOS)

El software del sistema operativo usado en equipos Cisco es conocido como *Internetwok Operating System* (IOS). Como cualquier sistema operativo en cualquier equipo de cómputo, el Cisco IOS administra los recursos en *hardware* y *software* del equipo, incluyendo asignación de memoria, procesos, seguridad y el sistema de archivos. El Cisco IOS es un sistema operativo multitarea que está integrado en *ruteo*, *switcheo*, interconexión y funciones de telecomunicaciones.

Aunque el Cisco IOS pareciera ser el mismo en varios equipos (*routers, switches*), existen diferentes imágenes IOS. Una imagen IOS es un archivo

que contiene el sistema operativo IOS completo para un equipo o modelo de equipo en específico. Cisco crea varios tipos diferentes de imágenes IOS, dependiendo tanto del modelo como de las funcionalidades que tenga la imagen. Normalmente, mientras más funcionalidades tenga una imagen de IOS, más grande será la imagen en tamaño, y por lo tanto se requerirá de más memoria *flash* y más RAM para poder almacenar y ejecutar el IOS.

Como con otros sistemas operativos, el Cisco IOS tiene su propia interfaz de usuario. Aunque algunos dispositivos tengan una interfaz de usuario gráfica (GUI), la interfaz de línea de comandos (CLI) es el método más común para configurar equipos Cisco.

En el arranque del equipo, el archivo *startup-config* que se encuentra en la NVRAM es copiada a la RAM y almacenada como el archivo *running-config*. El IOS ejecuta los comandos de configuración que se encuentran en el *running-config*. Cualquier cambio que realice el administrador de la red es guardado en el *running-config* y es inmediatamente ejecutado por el IOS.

3.3 El proceso de arranque de un equipo Cisco.

Existen cuatro principales fases en el proceso de arranque de un equipo:

1. El proceso de POST.
2. Carga del programa *Bootstrap*.
3. Ubicación y ejecución del sistema operativo Cisco IOS.
4. Ubicación y ejecución del archivo de configuración inicial o el ingreso vía modo set up.

1. El proceso de POST.

El POST (*Power-On Self Test*) es un proceso común que está presente en casi cualquier computadora durante el proceso de arranque. El proceso POST es usado para probar el *hardware* del equipo. Cuando el *router* o *switch* es encendido, *software* en el chip de ROM conduce el POST. Durante esta auto-prueba, el equipo ejecuta un diagnóstico desde el ROM hacia varios componentes de *hardware* incluyendo el CPU, RAM y NVRAM. Cuando el proceso POST ha finalizado, el equipo ejecuta el programa *bootstrap*.

2. Carga del programa *Bootstrap*.

Después del POST, el programa *bootstrap* es copiado del ROM a la RAM. Una vez en la RAM, el CPU ejecuta las instrucciones en este programa. La

función principal del programa *bootstrap* es localizar el Cisco IOS y ejecutarla en la RAM.

3. Ubicación y ejecución del sistema operativo Cisco IOS.

El IOS está normalmente guardado en la memoria *flash*, pero también puede ser almacenado en otros lugares como un servidor TFTP (*Trivial File Transfer Protocol*).

Si una imagen IOS no es localizada, una versión muy básica y limitada de IOS es copiada de la ROM a la RAM. Esta versión de IOS es utilizada para diagnosticar cualquier problema y puede ser utilizada para cargar o ejecutar una versión completa de IOS en la RAM.

Un servidor TFTP es usando generalmente como un servidor de respaldo de IOS pero también puede ser utilizado como el principal servidor donde se almacenan y ejecutan las imágenes IOS.

4. Ubicación y ejecución del archivo de configuración inicial o el ingreso vía modo set up.

Después de que el IOS es cargado, el programa bootstrap busca el archivo de configuración inicial, conocido como *startup-config*, en NVRAM. Este archivo tiene previamente guardados comandos y parámetros de configuración tales como:

- Direcciones en interfaces.
- Información de *ruteo*.
- *Passwords*.
- Cualquier otra configuración guardada por el administrador de red.

Si el archivo de configuración inicial, *startup-config*, es localizada en la NVRAM, éste es copiado a la RAM como el archivo de configuración actualmente operando, *running-config*.

Si el archivo de configuración inicial no existe en la memoria NVRAM, el equipo buscará un servidor TFTP.

Si el archivo de configuración inicial, *startup-config*, es encontrado en la NVRAM, el IOS lo ejecuta en la RAM como *running-config* y ejecuta los comandos uno a uno. El archivo *running-config* contiene direcciones de las interfaces, procesos a iniciar, *passwords* u otras características o funcionalidades.

Si el archivo de configuración inicial no es encontrado, el equipo muestra el *prompt* para entrar al modo *setup*. El modo *setup* muestra al usuario una serie de preguntas para la configuración básica del equipo, normalmente este modo no es utilizado por los administradores de la red.

3.4 Verificación del proceso de arranque de un equipo Cisco.

El comando *show versión* puede ser usado para verificar y realizar un *troubleshooting* básico de los componentes de *hardware* y *software* del equipo. El comando *show versión* muestra información de la versión de Cisco IOS que se encuentra ejecutando en el equipo, la versión del programa *bootstrap* e información acerca de la configuración del *hardware*, incluyendo la cantidad de memoria.

En la figura 3.1 se muestra la salida que da el comando.

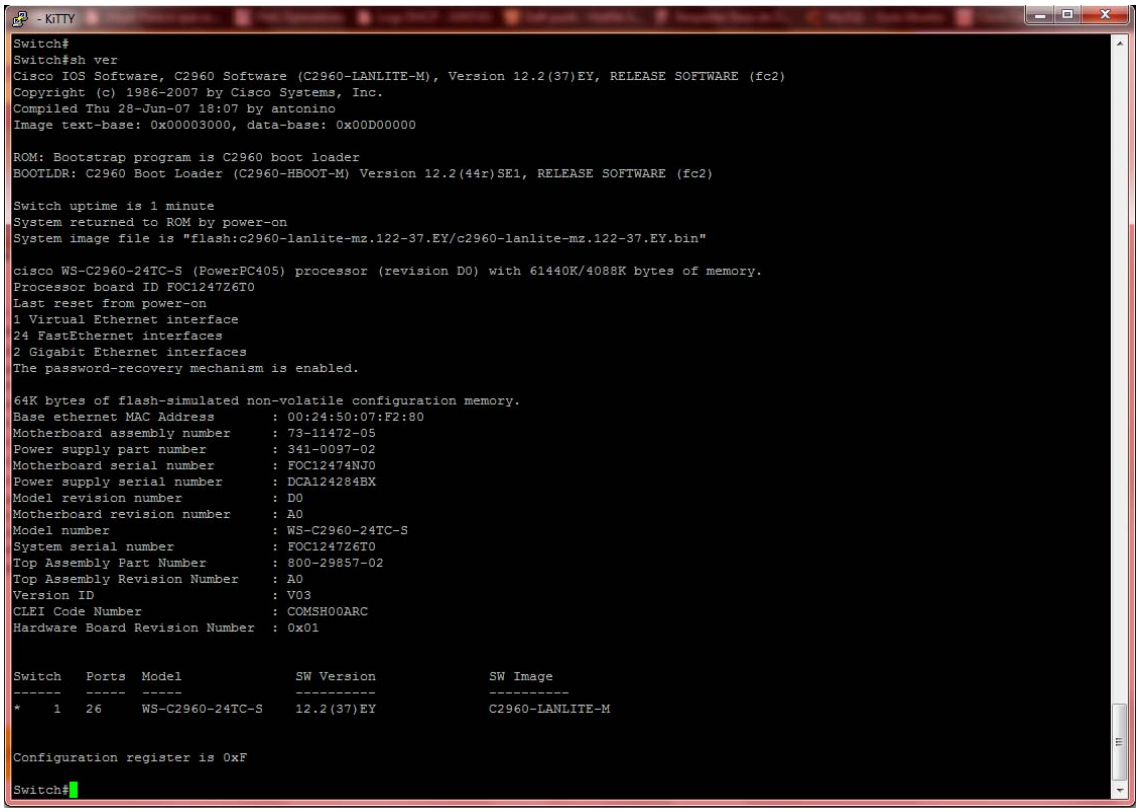


Figura 3.1: Comando *Show version*

Del comando de la Figura 3.1 se puede obtener la siguiente información:

Versión de IOS.

Cisco IOS Software, C2960 Software (C2960-LANLITE-M), Version 12.2(37)EY, RELEASE SOFTWARE (fc2)

Esta es la versión de Cisco IOS que está ejecutándose en RAM y que está siendo utilizada por el *switch*.

Programa en ROM del bootstrap.

ROM: Bootstrap program is C2960 boot loader
BOOTLDR: C2960 Boot Loader (C2960-HBOOT-M) Version 12.2(44r)SE1, RELEASE SOFTWARE (fc2)

Esta es la versión del software de sistema *Bootstrap*, el cual está almacenado en la memoria ROM, además que fue utilizada durante el arranque del *switch*.

Localización de la imagen IOS.

System image file is "flash:c2960-lanlite-mz.122-37.EY/c2960-lanlite-mz.122-37.EY.bin"

Esta es la ubicación y el nombre completo de la imagen de Cisco IOS que se ejecutó durante el proceso de arranque.

CPU y la cantidad de memoria RAM.

Cisco WS-C2960-24TC-S (PowerPC405) processor (revision D0) with 61440K/4088K bytes of memory.

La primera parte de la línea muestra el tipo de CPU del *switch*. La última parte de la línea muestra la cantidad de memoria DRAM.

Para determinar la totalidad de memoria DRAM en el equipo, se suman ambos dígitos. En este ejemplo, el switch 2960 tiene 61,440 KB (kilobytes) de espacio libre DRAM usada para el almacenaje temporal del Cisco IOS y de otros procesos de sistema. Los otros 4088 KB está dedicada a la memoria de paquetes. La suma de los dos es 65528K, o 64 megabytes (MB) del total de memoria DRAM.

Interfaces.

1 Virtual Ethernet interface
24 FastEthernet interfaces
DCC *Gigabit Ethernet interfaces*

Esta sección muestra las interfaces físicas en el equipo.

Cantidad de memoria NVRAM.

64K bytes of flash-simulated non-volatile configuration memory.

Se muestra la cantidad de memoria NVRAM necesaria para guardar el archivo de *startup.config*. [6]

Capítulo IV

Metodología

Conociendo cómo obtener la memoria, IOS actual y demás características de los equipos, lo que sigue es analizar la mejor metodología para realizar las actualizaciones, dado que son varios los equipos propuestos para su actualización se tiene que buscar la manera de automatizar los procedimientos.

Dentro de esta automatización encontraremos elementos como servidor FTP, *scripts* y procedimientos que nos ayudarán a realizar esta actividad de cambio de IOS, y nos permitirá tener más controlados y acotados los casos de falla que pudiesen presentarse.

En este capítulo se mostraran los requerimientos previos para poder seleccionar el método de actualización más adecuado para cada caso.

4.1 Modelos y capacidad de memoria importante en los equipos de acceso.

En la familia Cisco existen múltiples modelos de equipos de acceso, existen modelos que sólo trabajan en capa 2, es decir que su función básica es el *switcheo* mediante la dirección física del equipo o MAC (*Media Access Control*); existen otros modelos los cuales son capaces de trabajar en capa 2 o en capa 3 según convenga. Es importante conocer el modelo del equipo para poder buscar una imagen IOS adecuada, así como la cantidad de memoria *flash* y de memoria RAM.

La cantidad de memoria *flash* será muy importante en la definición del procedimiento a seguir para el cambio de IOS, esto es porque en algunos modelos de equipos de acceso la memoria *flash* es tan pequeña que sólo puede albergar una sola imagen IOS, en otros modelos la memoria *flash* puede albergar a uno o más imágenes de IOS.

A continuación se muestra un equipo donde la memoria *flash* puede albergar más de una imagen IOS:

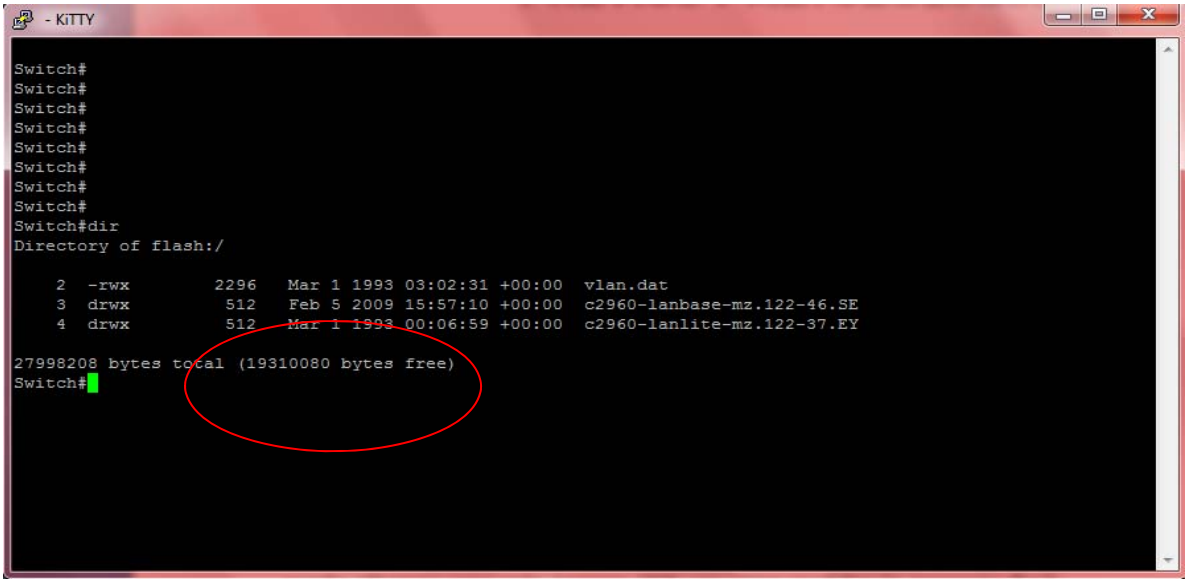


Figura 4.1 Memoria flash grande.

En la Figura 4.1 se muestra que la memoria *flash* disponible es de aproximadamente 28MB y todavía quedan disponibles cerca de 20MB, eso nos indica que aun cuando la imagen IOS sea más grande que la actual aún queda espacio de sobra para poder albergarla.

En la Figura 4.2 se muestra un equipo donde la memoria *flash* únicamente puede albergar una sola imagen ya que su capacidad de almacenamiento es baja, es fácil deducir debido a que del total de 8MB disponibles en la memoria ya están ocupados 5MB aproximadamente por lo que no queda espacio disponible adicional.

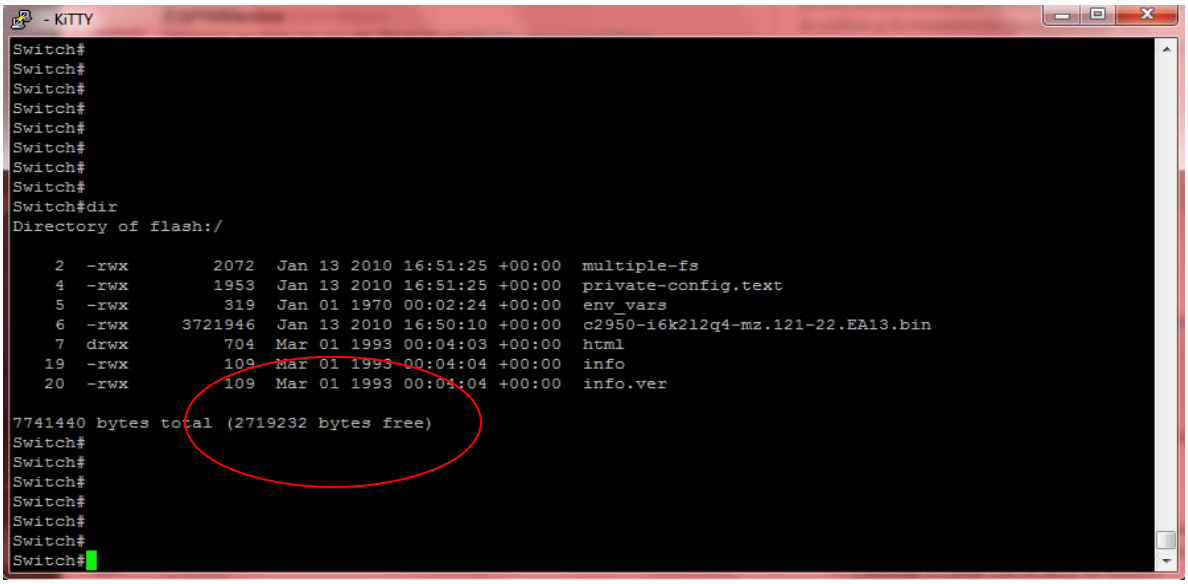


Figura 4.2 Memoria flash reducida.

Observando estos dos casos, es importante recalcar que no se puede seguir el mismo procedimiento de actualización de IOS para ambos equipos. El equipo que tiene un tamaño de memoria lo suficientemente grande para albergar más de una imagen IOS tiene una posibilidad menor de falla en la actualización, que un equipo donde sólo puede almacenar una imagen IOS a la vez.

En el primer caso, si se carga una imagen IOS errónea o corrupta en el equipo de acceso que impida que encienda correctamente, éste se puede intervenir para que cargue la otra imagen IOS que tiene disponible en la memoria *flash*. Esto no sucede en el segundo caso, primero se tiene que eliminar la imagen IOS que actualmente está ejecutando el equipo, en ese momento mientras este energizado no existirá ningún problema ya que el sistema operativo está ejecutándose en la memoria RAM, pero ¿qué pasa si durante el proceso de copiado de la nueva imagen hacia el equipo existiera una falla de energía eléctrica, o si la imagen estuviera corrupta o no fuera la imagen que le corresponde a dicho equipo? En estos casos al reiniciarse o iniciarse el equipo no tendría una imagen IOS válida para poder arrancar, y por consiguiente, el equipo no funcionaría durante ese lapso. Esto es una problemática considerando que los equipos que se van a actualizar se encuentran en lugares remotos, además que no existe personal calificado en dichos sitios para poder recuperar el sistema.

Para este proyecto se tienen tres procedimientos:

- Procedimiento donde los equipos tienen la capacidad de almacenar las dos imágenes IOS (la actual y la nueva).

- Procedimiento donde los equipos sólo tienen la capacidad de almacenar una imagen IOS.
- Procedimiento de contingencia.

Es importante considerar un procedimiento de contingencia cuando se requiere actualizar un equipo, sistema, *hadware*, etc, esto es porque en este tipo de actividades críticas, un error humano o un error del sistema o del equipo puede provocar que la actualización no sea exitosa. En cualquier ambiente computacional o de comunicaciones donde hoy en día lo más importante es la disponibilidad, todos los procesos de cambios y mejoras en cualquier sistema o infraestructura deben tener un procedimiento de "*roll back*" o de contingencia que nos permitirá restablecer el sistema o la infraestructura en caso de alguna falla. Debido a esto, en este proyecto se realizó un procedimiento de contingencia el cual aun cuando es de suma importancia su elaboración, la idea es no utilizarlo en absoluto.

4.2 Servidor FTP.

Para el envío de la imagen IOS a los equipos Cisco normalmente se utiliza el protocolo TFTP (*Trivial File Transfer Protocol*), el cual es muy similar al FTP (*File Transfer Protocol*) pero la diferencia entre los dos es que el segundo es orientado a conexión lo que permite que los paquetes lleguen sin errores y completos.

Para este procedimiento se utilizará el envío de las imágenes IOS mediante el protocolo FTP, esto debido a que los sitios afectados están en sitios remotos y necesitamos que la imagen viaje a través de la red sin la posibilidad de que se corrompa o que llegue incompleta. Se utilizará un servidor FTP instalado, con un usuario y contraseña para poder acceder a los equipos.

4.3 Scripts

Como se mencionó anteriormente se requiere intervenir varios equipos en la red, por lo que el hacerlo uno por uno disminuye la cantidad de equipos que se pueden actualizar en un determinado tiempo. Debido a ello para este proyecto se utilizaron *scripts* utilizando *expects* y programas sencillos en el lenguaje de programación Perl.

Los *scripts* utilizados varían dependiendo de cada caso, es decir, dependiendo de la cantidad de memoria *flash* se realizaron *scripts* específicos, esto porque se siguen dos procedimientos para ambos casos como ya se comentó anteriormente.

Capítulo V

Procedimiento de actualización de IOS

Como se abordó en los capítulos anteriores es importante definir dos procedimientos de actualización de IOS dependiendo de la capacidad de la memoria *flash* de cada equipo. Es el punto medular para decidir qué tipo de procedimiento se va a seguir.

Es importante seguir paso a paso dichos procedimientos ya que de esta manera será más complicado que ocurra alguna eventualidad.

Se explicará de manera detallada los pasos de cada procedimiento, esto es, para que se entienda que es lo que sucede en cada uno de ellos. Los procedimientos aun cuando sean destinados para personal del área de telecomunicaciones, también pueden y deben, ser entendidos por un personal con poca experiencia en los equipos.

En este capítulo se mostrarán los tres tipos de procedimientos que se crearon para este proyecto, por efectos de seguridad no se agregaron direcciones IP, *passwords* ni imágenes IOS válidas en dichos procedimientos.

5.1 Procedimiento para actualización de sistema operativo en equipos de acceso (Switches) en Órganos Delegacionales donde hay espacio en la memoria flash para al menos dos imágenes.

5.1.1 Conectarse vía remota al equipo.

Se debe conectar vía remota al equipo para verificar el modelo del mismo, y si el espacio en la memoria *flash* es suficiente para albergar dos imágenes IOS. El modelo es importante ya que se realizaron programas para automatizar la actualización, pero debido a que se valida la integridad de la imagen una vez enviada vía red, entonces los programas realizados están hechos a la medida dependiendo del modelo de *switch*.

Telnet X.X.X.X

5.1.2 Modelo del equipo.

Se determina el modelo del equipo utilizando el comando ***show versión.***

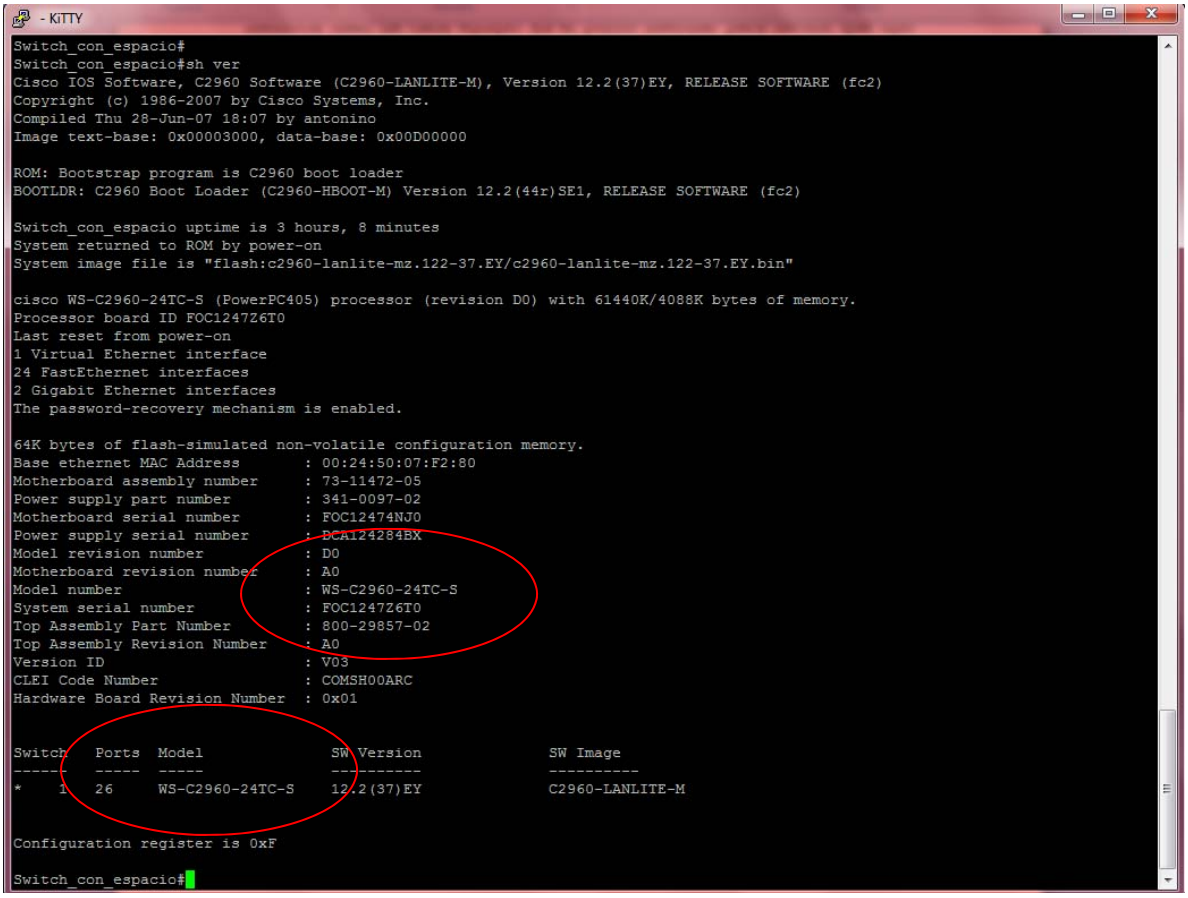


Figura 5.1: Modelo de equipo I.

Por salida del comando mostrada en la Figura 5.1 el modelo del equipo es WS-C2960-24TC-S.

5.1.3 Escoger el script dependiendo del modelo de equipo.

Para este proyecto se utilizaron dos *scripts*, el primero realiza el copiado de la imagen vía FTP y además comprueba que dicha imagen haya llegado completa e íntegra. El segundo *script* indica al equipo de acceso que debe bootear con la nueva imagen IOS y posteriormente se reinicia.
Se pone un ejemplo de nomenclatura de los scripts:

Modelo	Scripts a usar
WS-C2960-24TC-S	script_2960_24_TC_S.pl, script_reload_2960_24_TC_S.pl

5.1.4 Verificar que la nueva imagen IOS tenga el espacio suficiente en la memoria flash del equipo a intervenir.

Para verificar el espacio que se tiene en la memoria *flash* se utiliza el comando ***show flash:*** tal como se muestra en la Figura 5.2.

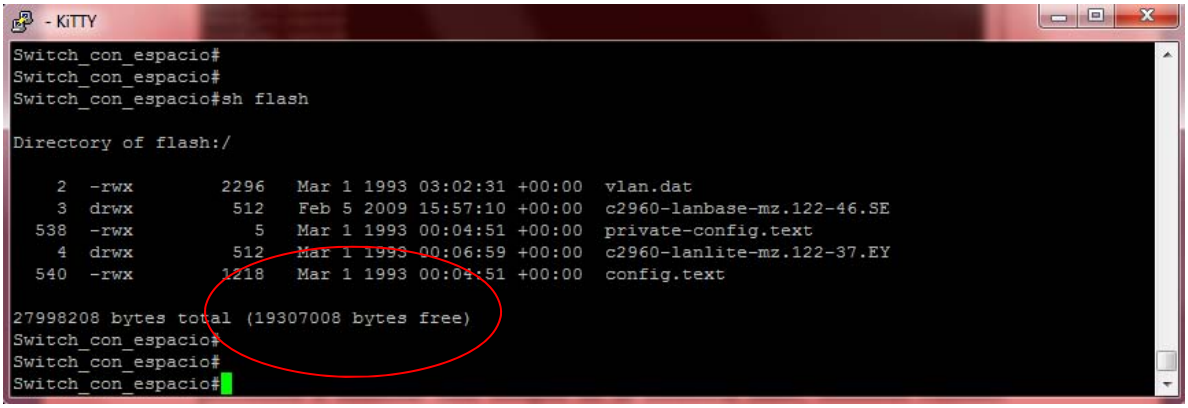


Figura 5.2: Espacio libre I.

El espacio libre que se observa en la Figura 5.2 es de 19MB aproximadamente, y tomando en cuenta el tamaño total de la memoria se observa que la imagen IOS actual ocupa poco espacio. En caso de que no exista espacio suficiente debido a que existan múltiples imágenes IOS se procederá a eliminar una imagen de la memoria *flash*, teniendo cuidado que no sea la imagen con la cual arrancó el equipo.

Para revisar con qué imagen el switch inició se utiliza nuevamente el comando ***show versión*** como se observa en la Figura 5.3.

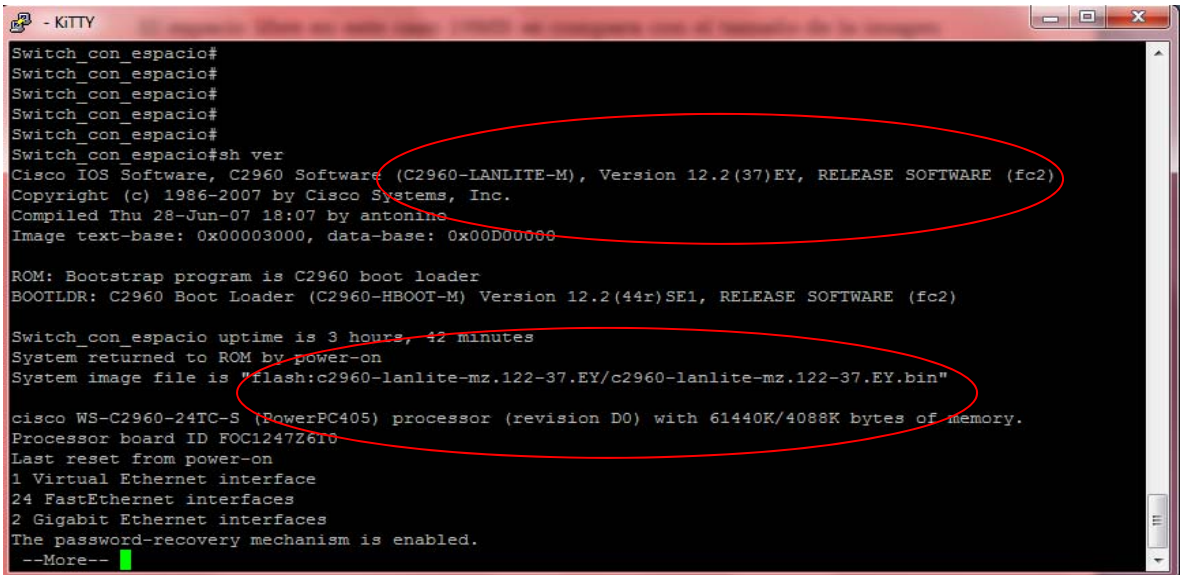


Figura 5.3: Imagen de arranque I

5.1.5 Explicación y ejecución de los scripts

Para la ejecución de los *scripts* se considera que la máquina que tiene acceso a los equipos (*switches*) tiene instalado el sistema operativo Linux, ya que los scripts creados se ejecutan bajo ese sistema operativo.

Estos *scripts* están hechos para que se puedan intervenir varios equipos en cascada. Por lo que el primer paso para ejecutarlos es crear una lista con las direcciones IP de todos los dispositivos a intervenir, esta lista deberá estar guardada en la misma ruta donde se encuentren los *scripts*. El nombre del archivo se llamará **switch.txt,** este nombre puede variar, es una variable que se manejará dentro del programa. Ejemplo:

[root@linux_update_ios]$ more switch.txt
10.X.X.X
10.Y.Y.Y
10.Z.Z.Z

Posteriormente, ya teniendo la lista de los equipos a intervenir se ejecuta el primer script de la siguiente forma:

[root@linux_update_ios]$ perl script_2960_24_TC_S.pl

A continuación se mostrará el contenido del programa, y se explicaran los bloques del mismo para entender cómo funciona.

[root@linux_update_ios]$ more script_2960_24_TC_S.pl

```
#!/usr/bin/perl –w

open(DATOS,"switch.txt");
open(IN,">>log.txt");
while (<DATOS>) {

@a=<DATOS>;

}
for($i=1;$i<@a;$i++){
   print IN `./update_2960_24_TC_S.exp $a[$i]`;
}
close(DATOS);
close(IN);
```

Dentro de este programa se manejan dos archivos, el archivo *switch.txt* el cual contiene las direcciones IP de los equipos a intervenir, y el archivo *log.txt*. El primer archivo únicamente es leído por el programa, a diferencia del segundo el cual guardará el resultado obtenido en la ejecución del programa para que al final se pueda revisar si existieron errores durante la ejecución.

```
Open(DATOS,"switch.txt");
open(IN,">>log.txt");
```

Se utiliza un ciclo *while* donde se guarda en la variable "@a" cada una de las direcciones IP de los equipos a intervenir.

```
While (<DATOS>) {

@a=<DATOS>;

}
```

Posteriormente en un ciclo *for* se ejecutará un expect por cada dirección IP guardada en el archivo *switch.txt.*

```
for($i=1;$i<@a;$i++){
   print IN `./update_2960_24_TC_S.exp $a[$i]`;
}
```

A continuación se dará una explicación de lo que hace el expect:

[root@linux_update_ios]$ more update_2960_24_TC_S.exp

```
#!/usr/bin/expect –f
spawn telnet [lindex $argv 0]
```

```
match_max 100000
expect “*Username:*”
send “cisco\n”
expect “*password:*”
send “cisco\n”
expect “\#”
set timeout -1
send “copy ftp://user:password@X.X.X.X/imagen.bin flash:\n”
expect “*?*”
send “\n”
expect “\#”
send “verify /md5 flash:imagen.bin
56788133a4f7d6e41d5e28133d49acf3\n”
expect “\#”
send “exit\n”
exit
```

Dentro de este programa expect se realizan los siguientes pasos:

- Conectarse al switch remoto utilizando la IP del archivo *switch.txt.*

```
spawn telnet [lindex $argv 0]
match_max 100000
expect “*Username:*”
send “cisco\n”
expect “*password:*”
send “cisco\n”
```

- Realizar la copia de la nueva imagen IOS llamada *imagen.bin* que se encuentra en el servidor FTP X.X.X.X cuyas credenciales de autenticación son “user” y “password.”

```
Send “copy ftp://user:password@X.X.X.X/imagen.bin flash:\n”
```

- Por último para validar que la imagen copiada en el equipo sea la correcta, se hace una prueba de integridad mediante el uso del algoritmo md5. Se obtiene el cheksum (es el número que se genera al aplicarle el algoritmo md5) primero de la imagen una vez descargada de la página de cisco, posteriormente se compara con el cheksum que se obtiene en el equipo donde se copió, esto se muestra en la siguiente línea del programa:

send "verify /md5 flash:imagen.bin 56788133ª4f7d6e41d5e28133d49acf3\n"

Finalizando la ejecución del *expect* para cada uno de los equipos a intervenir, a continuación queda validar mediante el archivo *log.txt* que las imágenes IOS copiadas a dichos equipos sean válidas, es decir, que pasen la prueba de integridad.

El ejemplo siguiente muestra cuando una imagen es copiada correctamente, visualizando el archivo log.txt:

Switch_con_espacio#verify /md5 flash:imagen.bin 50c64101d9a30e8e13bc52dea0851e65
…….
…….
……..Done!
Verified (flash:imagen.bin) = 50c64101d9a30e8e13bc52dea0851e65

Si sale este mensaje quiere decir que el archivo se copió correctamente, ahora si en dado caso que se encontrara algún error, el log mostraría algo como lo siguiente:

%Error verifying flash:imagen.bin
Computed signature = 50c64101d9a30e8e13bc52dea0851e65
Submitted signature = 50c64101d9a30e8e13bc52dea0851e64

Esto indicaría que la imagen en algún momento fue modificada y, por lo tanto si nosotros utilizáramos esta imagen para actualizar el equipo podríamos provocar una falla en dicha actualización y dejar sin red al sitio más tiempo del planeado. Por esa razón si se encontrará este último mensaje en alguno de los equipos intervenidos se deberá borrar de la memoria *flash* dicha imagen, y se tendrá que realizar nuevamente la transferencia del archivo utilizando el mismo programa en perl.

En el caso donde las imágenes fueron copiadas correctamente, entonces lo único que resta para que el switch arranque o inicie con el nuevo Sistema Operativo es reiniciándolo, para lo cual se creó otro script.

[root@linux_update_ios]$ perl script_reload_2960_24_TC_S.pl

A continuación se mostrará el contenido del programa, y se explicaran los bloques del mismo para entender cómo funciona.

[root@linux_update_ios]$ more script_reload_2960_24_TC_S.pl
#!/usr/bin/perl –w

```
open(DATOS,"switch.txt");
open(IN,">>log1.txt");
while (<DATOS>) {

@a=<DATOS>;

}
for($i=1;$i<@a;$i++){
   print IN `./reload_2960_24_TC_S.exp $a[$i]`;
}
close(DATOS);
close(IN);
```

Igualmente que en el script anterior se manejan dos archivos, el archivo *switch.txt* el cual contiene las direcciones IP de los equipos a intervenir, y el archivo *log1.txt*.

```
open(DATOS,"switch.txt");
open(IN,">>log.txt");
```

Se utiliza un ciclo *while* donde se guarda en la variable "@a" cada una de las direcciones IP de los equipos a intervenir.

```
While (<DATOS>) {

@a=<DATOS>;

}
```

Posteriormente en un ciclo *for* se ejecutará un expect por cada dirección IP guardada en el archivo *switch.txt.*

```
for($i=0;$i<@a;$i++){
   print IN `./reload_2960_24_TC_S.exp $a[$i]`;
}
```

A continuación se dará una explicación de lo que hace el expect:

[root@linux_update_ios]$ more reload_2960_24_TC_S.exp

```
#!/usr/bin/expect –f
spawn telnet [lindex $argv 0]
match_max 100000
expect "*Username:*"
send "cisco\n"
```

```
expect “*password:*”
send “cisco\n”
expect “\#”
send “conf t\n”
expect “\#”
send “boot system flash:/imagen.bin\n”
expect “\#”
send “exit\n”
expect “\#”
send “wr\n”
expect “\#”
send “reload\n”
expect “*confirm*”
send “\n”
```

- Conectarse al switch remoto utilizando la IP del archivo *switch.txt.*

```
spawn telnet [lindex $argv 0]
match_max 100000
expect “*Username:*”
send “cisco\n”
expect “*password:*”
send “cisco\n”
```

- Cambiar la variable *boot* del equipo, esta variable posee la ubicación de la imagen flash con la que el equipo debe de arrancar cuando es iniciado.

Send “boot system flash:/imagen.bin\n”

- Por último se realiza el reinicio del equipo para que cargue la nueva imagen IOS.

Send “reload\n”

Ahora sólo queda esperar el reinicio del equipo, en dado caso que no levantara el equipo correctamente se tendrá que revisar el archivo log1.txt para verificar dónde estuvo el error, en dado caso de no encontrarlo se tendrá que aplicar el procedimiento de contingencia para poder recuperar el equipo.

5.2 Procedimiento para actualización de sistema operativo en equipos de acceso (Switches) en Órganos Delegacionales donde NO hay espacio en la memoria flash para al menos dos imágenes.

Este procedimiento es muy similar al anterior, con la principal diferencia que no todos los pasos se automatizaron, como ya se ha comentado este procedimiento es más riesgoso debido a que el equipo de acceso se queda sin la imagen del Sistema Operativo durante unos minutos, por lo que se prefiere realizar las cuestiones más críticas sin la ayuda de *scripts*. La automatización siempre simplifica el trabajo, pero en ocasiones es mejor escoger otro método que aunque un poco más tardado, nos garantizará tener el ambiente más controlado.

5.2.1 Conectarse vía remota al equipo.

Se debe conectar vía remota al equipo para verificar el modelo del mismo, y el espacio en la memoria *flash* que en este caso no es suficiente para albergar dos imágenes IOS al mismo tiempo. El modelo es importante ya que se realizaron programas para automatizar la actualización, pero debido a que se valida la integridad de la imagen una vez enviada vía red, entonces los programas realizados están hechos a la medida dependiendo del modelo de switch.

Telnet X.X.X.X

5.2.2 Modelo del equipo.

Se determina el modelo del equipo utilizando el comando ***show versión.***

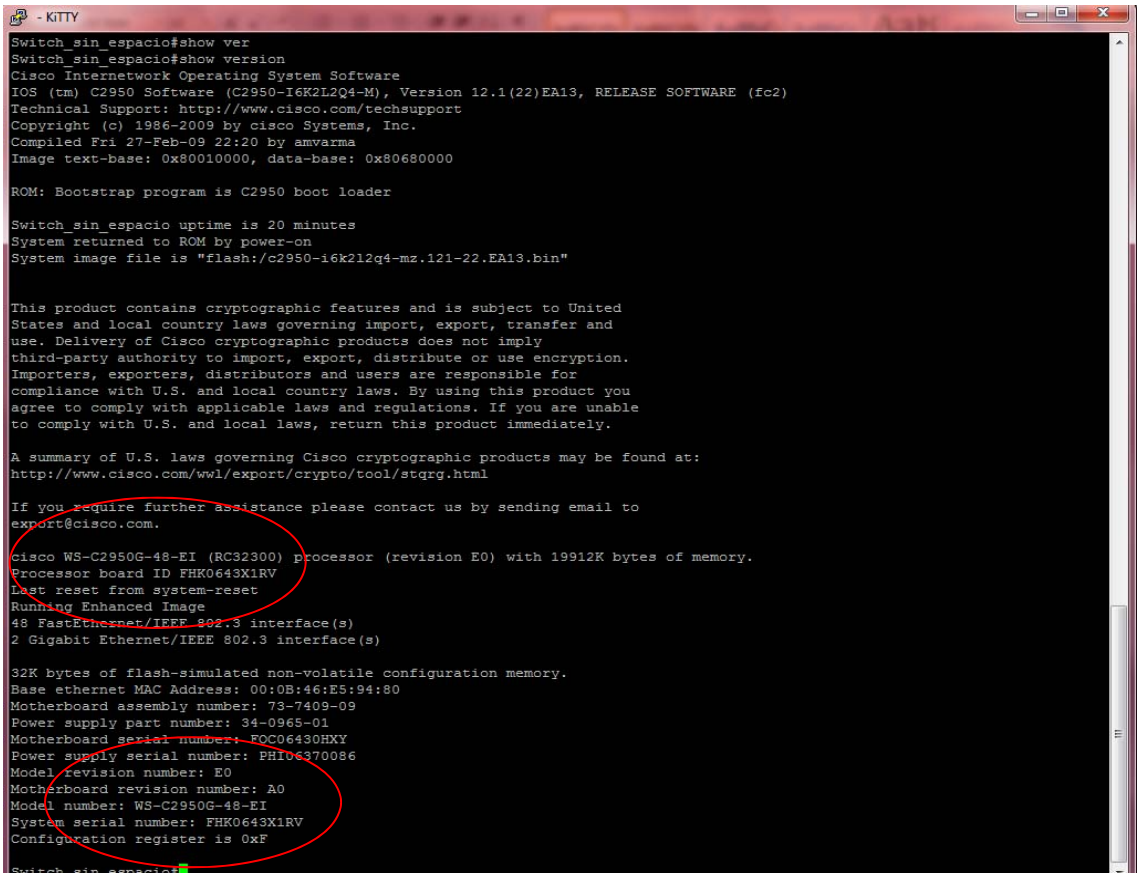


Figura 5.4: Modelo de equipo II.

Por salida del comando mostrada en la Figura 5.4 el modelo del equipo es WS-C2950G-48-EI.

5.2.3 Escoger el script dependiendo del modelo de equipo.

Para este proyecto se utilizó únicamente un *script*, el cual realiza la transferencia vía FTP al router del sitio, los demás pasos para su actualización se realizaran de manera manual dado que este caso es crítico ya que durante el proceso el *switch* a intervenir no tendrá en la memoria *flash* ninguna imagen IOS para su arranque, y por consiguiente cualquier falla de energía o reinicio del equipo fuera de tiempo podrá ocasionar que el sitio remoto que estamos interviniendo se quede sin los servicios de red por un tiempo prolongado.

Se pone un ejemplo de nomenclatura de los *scripts*:

Modelo	Scripts a usar
WS-C2950G-48-EI	script_send_2950G.pl

5.2.4 Explicación y ejecución de los scripts

Como en el procedimiento anterior se considera que la máquina que tiene acceso a los equipos de acceso tiene instalado el sistema operativo Linux, ya que los *scripts* creados se ejecutan bajo ese sistema operativo.

Este *script*s es de similares características que los anteriores, y está hecho para poder intervenir a varios equipos a la vez. Como ya se comentó anteriormente para que se pueda copiar la nueva imagen IOS se necesita espacio en la memoria, por lo que es imprescindible borrar la imagen con la cual arrancó el sistema. Este punto es muy relevante, durante esa transferencia de la imagen el *switch* puede perder el suministro de energía eléctrica, se puede perder el enlace WAN, entro otros problemas que pueden ocasionar que la transferencia no sea exitosa, por eso mismo para este procedimiento y específicamente para este *script* se realiza primero la transferencia de la imagen IOS al *router* del sitio. Posteriormente se hace la transferencia al *switch*, dado que la transferencia es local es menos probable que la imagen llegue corrupta o incompleta.

Para ésto usamos el *script*, únicamente para enviar la imagen al router, por lo que este programa se basa en una lista de direcciones IP de los *routers* de los sitios a intervenir, esta lista deberá estar guardada en la misma ruta donde se encuentren los *scripts*. El nombre del archivo se llamará **routers.txt,** este nombre puede variar, es una variable que se manejará dentro del programa. Ejemplo:

[root@linux_update_ios]$ more routers.txt
10.X.X.X
10.Y.Y.Y
10.Z.Z.Z

Posteriormente, ya teniendo la lista de los equipos a intervenir se ejecuta el primer script de la siguiente forma:

[root@linux_update_ios]$ perl script_send_2950G.pl

A continuación se mostrará el contenido del programa, y se explicaran los bloques del mismo para entender cómo funciona.

[root@linux_update_ios]$ more script_send_2950G.pl

```
#!/usr/bin/perl –w
open(DATOS,”routers.txt”);
open(IN,”>>log2.txt”);
while (<DATOS>) {
@a=<DATOS>;
```

```
}
for($i=1;$i<@a;$i++){
   print IN `./send_2950G.exp $a[$i]`;
}
close(DATOS);
close(IN);
```

Como en los programas anteriores se utilizan dos archivos, el de router.txt y el de log2.txt el cual guardará la ejecución de todo el script para poder validar algún error en el mismo.

Open(DATOS,"switch.txt");
open(IN,">>log.txt");

Se utiliza un ciclo *while* donde se guarda en la variable @a cada una de las direcciones IP de los equipos a intervenir.

While (<DATOS>) {

@a=<DATOS>;

}

Posteriormente en un ciclo *for* se ejecutará un expect por cada dirección IP guardada en el archivo *routers.txt.*

for($i=1;$i<@a;$i++){
*   print IN `./* send_2950G.exp *$a[$i]`;*
}

A continuación se dará una explicación de lo que hace el expect:

[root@linux_update_ios]$ more send_2950G.exp

```
#!/usr/bin/expect –f
spawn ssh –l cisco [lindex $argv 0]
match_max 100000
expect "*password:*"
send "cisco\n"
expect "\#"
set timeout -1
send "copy ftp:// user:password@X.X.X.X/imagen.bin flash:\n"
expect "*?*"
send "\n"
expect "\#"
```

```
send "verify /md5 flash:imagen.bin
2869d2c201e59258e23194498511a24e\n"
expect "\#"
send "exit\n"
exit
```

Como se puede observar el *expect* es muy similar que los usados anteriormente, la única diferencia notable es el método de conexión al sitio remoto que en vez de ser mediante el protocolo *telnet* ahora es mediante el protocolo *ssh.*

De la misma forma que en el anterior procedimiento se debe revisar el archivo de logs, en este caso llamado *log1.txt* para validar si la imagen se copió sin problemas en los equipos remotos y, en dado caso que se observara un error en el mismo se tendría que enviar el archivo nuevamente.

5.2.5 Borrado de imagen actual y actualización del equipo.

En este momento ya tenemos la imagen en el router, la cual ya comprobamos que llego de manera íntegra y completa, como en este caso el equipo de comunicaciones sólo puede albergar una sola imagen debido al tamaño de su memoria *flash* se tiene que borrar la imagen que tiene actualmente. Esta parte es la más riesgosa del procedimiento ya que al momento de eliminar la imagen que tiene actualmente el equipo, ya no tendrá un sistema operativo desde donde arrancar, mientras este encendido funcionará correctamente pero, si es reiniciado o si el suministro de energía eléctrica es interrumpido por un pequeño lapso el equipo no podrá arrancar.

Como una buena práctica se debe asegurar un respaldo de suministro eléctrico en los equipos críticos, por lo que para los equipos de comunicaciones se utiliza un UPS (*Uninterruptible Power Supply*) para solventar las variaciones de voltaje y la pérdida eventual del suministro eléctrico. El UPS debe de alguna manera estar monitoreado para conocer que sus baterías se encuentran saludables y que se encuentra funcionando correctamente. Además se debe verificar que el equipo que vamos a intervenir efectivamente se encuentra conectado al UPS.

Ya validado el correcto funcionamiento del UPS y que efectivamente el equipo se encuentra conectado en el mismo, se procede a eliminar la imagen IOS del equipo mediante el siguiente comando:

Switch_sin_espacio#dir
Directory of flash:/
2 -rwx 2774747 Mar 01 1993 00:03:23 c2950-i6q4l2-mz.121-12c.EA1.bin
3 -rwx 273 Jan 01 1970 00:01:39 env_vars
4 -rwx 1808 Mar 01 1993 00:02:58 config.text
5 -rwx 5 Mar 01 1993 00:02:58 private-config.text
7 drwx 832 Mar 01 1993 00:04:11 html
21 -rwx 110 Mar 01 1993 00:04:13 info
22 -rwx 110 Mar 01 1993 00:04:13 info.ver
7741440 bytes total (3245056 bytes free)

Ya con el nombre de la IOS se procederá a eliminarla con el comando *delete*.

Switch_sin_espacio#delete c2950-i6q4l2-mz.121-12c.EA1.bin
Delete filename [c2950-i6q4l2-mz.121-12c.EA1.bin]?
Switch_sin_espacio#
Delete flash: c2950-i6q4l2-mz.121-12c.EA1.bin? [confirm]

A continuación se realizará la copia de la imagen que se encuentra en el *router* hacia el *switch*, como es un paso sensible debido a que en estos momentos el *switch* se encuentra vulnerable, se realizarán estos pasos manualmente.

Para copiar la nueva imagen se utilizará el *router* local como servidor TFTP para esto se utilizará el siguiente comando, la imagen IOS dependerá del modelo del switch:

R1(Config)#tftp-server flash:/imagen.bin

Se realizará el copiado de la nueva IOS que se encuentra en el *router* al *switch* mediante el siguiente comando:

maqueta-C2950G-24-EI#copy tftp:imagen.bin flash:
Address or name of remote host []? 10.X.X.X **(Dirección del router)**
Destination filename [imagen.bin]?
Accessing tftp://10.0.X.X/imagen.bin…
Loading imagen.bin from 10.X.X.X (via VlanX):
!!!
[OK – 3721946/4096 bytes]
3721946 bytes copied in 43.144 secs (86268 bytes/sec)

Se deberá verificar la integridad de la imagen copiada mediante el comando **verify**, el número al final del comando es el md5 de la imagen original.

Verify /md5 flash:imagen.bin **2869d2c201e59258e23194498511a24e**

En esto momento ya se encuentra la nueva imagen IOS en el equipo, ahora antes de proceder a reiniciar se necesita cambiar la variable boot y al final se reinicia, tal y como muestran los siguientes comandos:

maqueta-C2950G-24-EI(config)#boot system flash:/imagen.bin
maqueta-C2950G-24-EI(config)#exit
maqueta-C2950G-24-EI#wr
Building configuration…
[OK]
maqueta-C2950G-24-EI#**reload**
Proceed with reload? [confirm]

Se esperará a que reinicie el equipo y se verificarán los resultados.

5.3 Procedimiento de contingencia en Órganos Delegacionales para recuperar el Sistema Operativo (IOS) de un switch debido a que la imagen está corrupta o es inexistente.

Este procedimiento se realizó para poder recuperar un equipo si alguno de los procedimientos anteriores no se siguió correctamente, o si existió un incidente que provocará que el equipo no arrancara con una imagen IOS válida. Dentro de este procedimiento se considera el apoyo de personal en sitio que nos facilite la conexión al equipo en contingencia.

Se debe de considerar al menos los siguientes insumos para poder recuperar el equipo remoto:

- Un equipo de cómputo el cual tenga acceso a la red.
- Se requiere en sitio un cable consola para poder conectar el equipo de comunicaciones con el equipo de cómputo. (En este caso los Órganos Delegacionales del Instituto cuentan con un cable consola para atender emergencias parecidas a esta).
- Personal en sitio debe de recibir vía correo electrónico, ftp, u otra vía la imagen válida del equipo que se encuentra en sitio.

En el caso de los sitios remotos el *router* que se tiene instalado en sitio tiene puertos *Ethernet*, es decir, tiene un *mini-switch* integrado. Existen también sitios donde existe más de un *switch*, por lo que no es necesario

configurar los puertos del *router*, en estos casos se conectará el *switch* adicional al *router* y a partir de ahí seguir los demás pasos del procedimiento.

5.3.1 Configuración del router para la conexión del equipo de cómputo

En este caso al perder el *switch*, todos los usuarios conectados al mismo perderán el acceso a red. Para este caso necesitamos conectar un equipo de cómputo a red para que a través de él podamos intervenir el equipo dañado.

Al tener dañado el *switch* procederemos a configurar los puertos *Ethernet* del *router* en su vlan correspondiente y con la configuración de puertos de acceso tal y como se muestra a continuación:

- Se creará la vlan XX en el router(según la VLAN que aplique):

router #vlan database
router (vlan)#vlan XX
VLAN XX added:
Name: VLANXX

- Se configurará el puerto interface FastEthernet0/X/0:

router(config)#interface FastEthernet0/X/0
router (config-if)#switchport mode access
router (config-if)#switchport access vlan XX
router (config-if)#spanning-tree portfast
router (config-if)#no shutdown

Ya configurado el puerto se procederá a conectar la computadora mediante un cable de red al mismo y se verificará que tenga salida a internet. Una vez validado el acceso a red se conectara el cable consola del equipo de cómputo al equipo de acceso a intervenir.

5.3.2 Configuración del equipo de cómputo para su acceso remoto.

Considerando que la persona en el sitio no tiene la experiencia necesaria para poder entrar al equipo se tendrá que configurar el equipo para su acceso remoto, se procurará que el equipo de cómputo tenga instalado el sistema operativo Windows XP ya que éste tiene por default instalado el programa *HyperTerminal*, programa que utilizaremos en este procedimiento. Se puede utilizar otro sistema operativo en el equipo de cómputo pero se tendrá que descargar el programa antes mencionado.

Existen varios métodos para acceder a un equipo remotamente, en este caso se manejarán dos de ellos. El primero es configurando un usuario con permisos de acceso remoto en el equipo, para ello se considerarán los siguientes pasos:

5.3.2.1 Configurar acceso remoto mediante escritorio remoto.

Creación de usuario para acceso remoto.

Cuando la PC esté en red, se le pedirá al personal de apoyo del Órgano Delegacional la creación de un usuario y *password* (*user*: cisco, *password*: cisco) con permisos de administrador. Para lograr esto se necesita ir a *Panel de Control*, y dentro de éste a cuentas de usuarios. Si pulsamos en esa opción se nos muestran varias opciones relacionadas con las cuentas de usuarios, entre ellas la de "c*rear una cuenta nueva*. Se escogerá esta opción y se llenarán la información que se pida correspondiente al nombre de la cuenta (cisco), *password* (cisco) y tipo de cuenta (administrador).

Configuración de acceso remoto (remote desktop)

Adicionalmente se le pedirá al personal de apoyo del Órgano Delegacional la habilitación del escritorio remoto. Para realizar esta habilitación se requiere ir a *Inicio>Panel de control>Sistema*, o bien mostrar las propiedades del sistema haciendo click con el botón derecho en el icono de *Mi PC*. Se pulsa en la pestaña *Remoto* y se marca la casilla *Permitir* que los usuarios se conecten de manera remota a este equipo, tal como se muestra en la Figura 5.5.

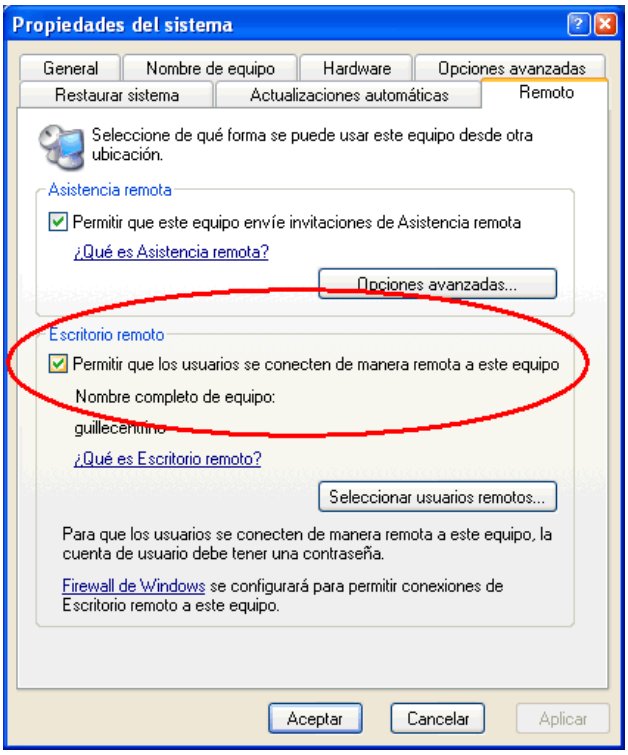


Figura 5.5: Acceso remoto.

Validación de servicios de red.

Finalmente se probará la conexión del escritorio remoto con las credenciales configuradas en el punto anterior, si es fallida la conexión, se revisarán los puntos anteriores.

La configuración del escritorio remoto anterior es para el sistema operativo Windows XP, para otra versión los pasos tendrán algunas modificaciones.

La otra opción para poder acceder al equipo de cómputo de forma remota es a través de un programa llamado *TeamViewer*, este método es más sencillo que el anterior ya que sólo se le pide al usuario remoto ejecutar un archivo y podemos acceder a él remotamente, a continuación se muestra cómo funciona.

5.3.2.2 Configurar acceso remoto mediante Teamviewer.

Paso 1. Se descarga de la página http://www.teamviewer.com/es/download/index.aspx el programa de *TeamViewer* versión completa, el cual se instala en el equipo de cómputo origen como se muestra en la Figura 5.6.

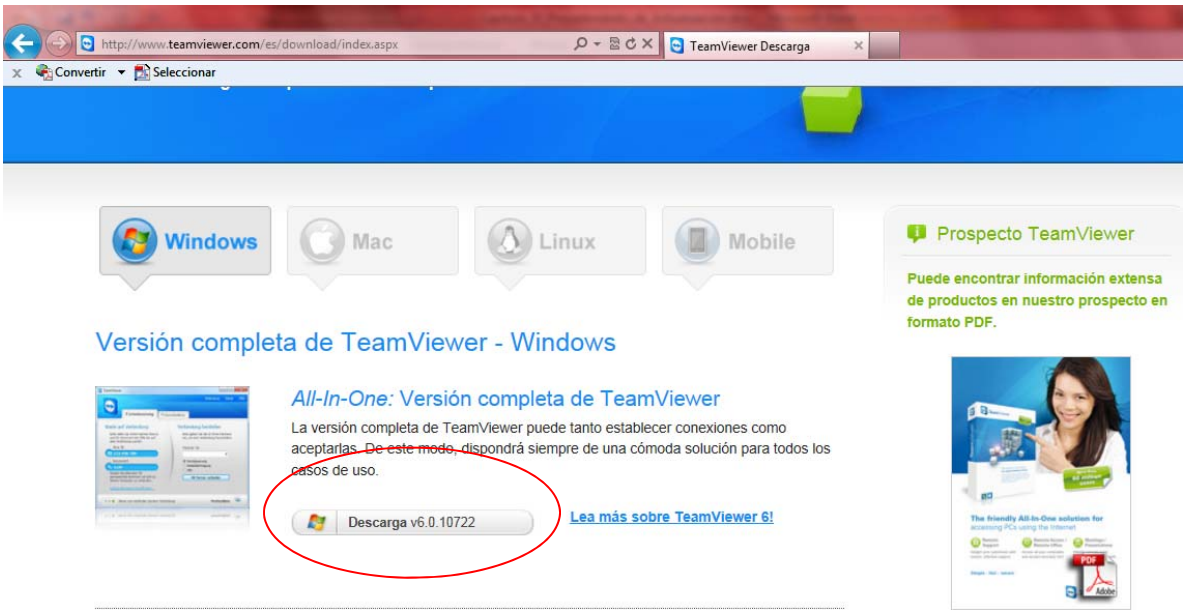


Figura 5.6: TeamViewer versión completa.

Paso 2. Ahora se procederá a descargar otra versión del TeamViewer la cual se le enviará vía correo electrónico o se le indicará a la persona de apoyo en sitio que la descargue, esta versión se ejecuta pero no se instala en el equipo de cómputo precisamente porque está destinada para los equipos remotos a conectarse, Figura 5.7.

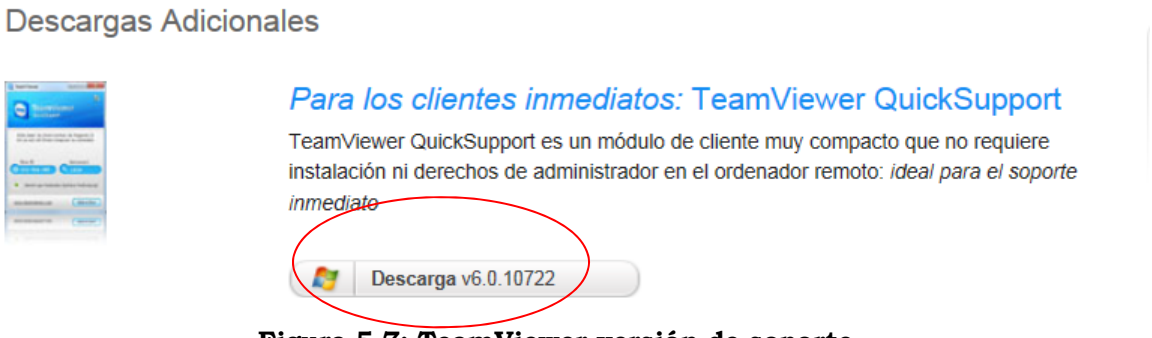


Figura 5.7: TeamViewer versión de soporte.

Paso 3. En la máquina origen se ejecutará el programa, como se muestra en la Figura 5.8. Se tienen dos campos, el primero dice *Permitir el control remoto*, este apartado sirve si queremos que alguien tome el control de nuestro equipo, para ello le tenemos que proporcionar los datos de ID y de contraseña; en este caso nosotros vamos a conectarnos al equipo remoto que es la parte de *Controlar un ordenador remoto*, en ese apartado necesitamos los datos de ID y de Contraseña los cuales nos proporcionará el personal en sitio ejecutando el archivo que previamente se les proporcionó, como se muestra en la pantalla de la Figura 5.9 el personal en sitio nos debe proporcionar los datos que vienen en los campos de *Su ID*

y *Contraseña*, precisamente estos datos son los que nos van a permitir podernos conectar remotamente al equipo de cómputo de apoyo.

Figura 5.8: TeamViewer origen.

Figura 5.9: TeamViewer remoto.

Paso 4. En este paso al ingresar los datos de ID y Contraseña podremos tomar el control del equipo remoto, y de esta manera poder intervenir el equipo que se encuentra dañado.

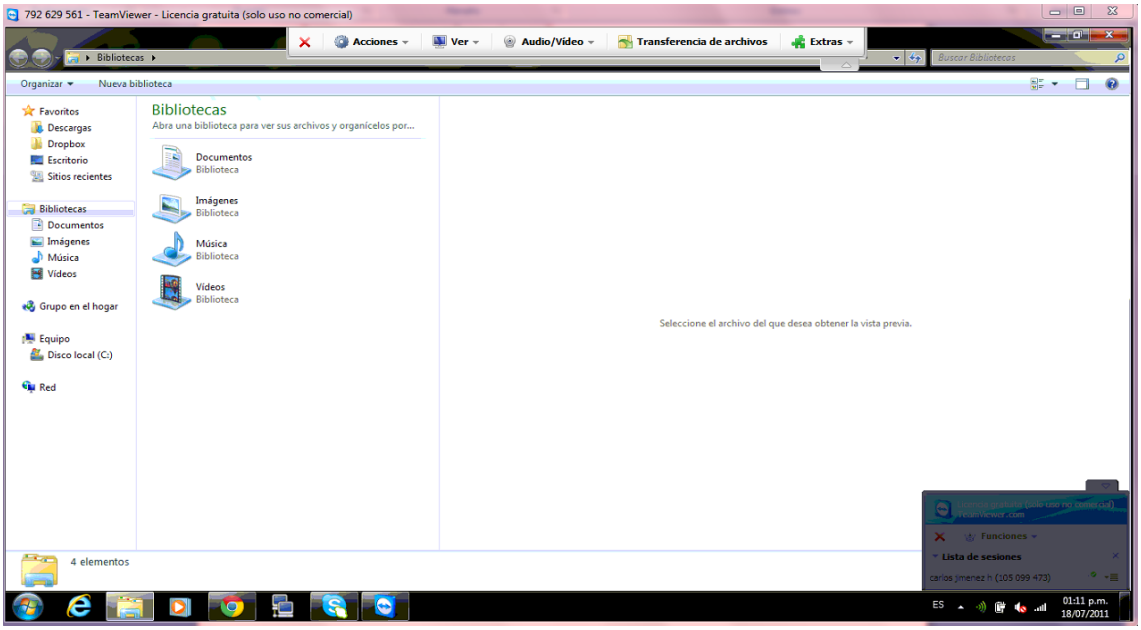


Figura 5.10 Equipo en control

5.3.3 Descarga de la imagen IOS

Ahora se debe de tener la imagen válida en la máquina remota por lo que se necesita hacer llegar por algún medio dicho archivo, en este proyecto se creó una página web sencilla donde se encontraban links de descarga para poder obtener las imágenes IOS que corresponden a los *switches* que se tenían planeado actualizar, de esta manera se facilita la descarga de las imágenes.

5.3.4 Recuperación

5.3.4.1 Conexión al switch mediante Hyperterminal

Se debe conectar el equipo de cómputo mediante el cable consola al *switch* utilizando los siguientes parámetros:

- ✓ Bits por segundo: 9600
- ✓ Bits de datos: 8
- ✓ Paridad: Ninguno
- ✓ Bits de parada: 1
- ✓ Control de Flujo: Ninguno

5.3.4.2 Problemas al arranque de un equipo de acceso (switch)

Cuando un *switch* experimenta problemas para arrancar el sistema operativo pueden ocurrir las siguientes condiciones:

a) El *switch* se reinicia constantemente

b) El *switch* muestra el *prompt* "*switch*:"

c) El mensaje de "*error loading flash*:" aparece.

Para el caso b) y c) el *switch* visualizará el prompt *switch*:
Para el caso a) donde el switch se reinicia constantemente se deben seguir los siguientes pasos, existen varios casos dependiendo el modelo de equipo, se ponen los casos más frecuentes:

i. Caso 1:

- ➢ Se desconecta el cable de potencia del equipo.

- Se mantiene presionado el botón MODE mientras se reconecta el cable de potencia al equipo.
- Se deja de presionar el botón MODE después de que el led STAT se apague.
- Finalmente se visualiza el prompt *switch*:

ii. Caso 2:

- Se desconecta el cable de potencia del equipo.
- Se mantiene presionado el botón MODE mientras se reconecta el cable de potencia al equipo.
- Se deja de presionar el botón MODE después de que el led SYST parpadea en color ámbar y se apaga.
- Finalmente se visualiza el prompt *switch*:

iii. Caso 3:

- Se desconecta el cable de potencia del equipo.
- Se mantiene presionado el botón MODE mientras se reconecta el cable de potencia al equipo.
- Se deja de presionar el botón MODE cuando se visualice el prompt *switch*:

Al haber ya accedido al prompt switch: se puede proceder con el procedimiento, de lo contrario revisar nuevamente los pasos anteriores.

5.3.4.3 Arranque de un equipo utilizando una imagen IOS en la memoria flash.

Ya estando con el prompt *switch*: se ejecutan los siguientes comandos:

switch: flash_init
switch: load_helper

Ahora con los comandos dir flash: se buscará una imagen en la memoria flash y se tratará de inicializarla con el comando:

boot flash:XXXXXX.bin

Si la carga no es exitosa o no hay una imagen que cargar se tendrá que copiar la imagen vía XMODEM.

5.3.4.4 Copiado de la imagen IOS vía Xmodem

Se verificará el espacio en la flash, si no lo tiene se borrará la imagen que no fue posible cargar u otra imagen que ocupe espacio innecesario. Esto se realiza con el comando:

delete flash:XXXXXX.bin

A. Para copiar la imagen en el *switch* se utiliza el comando copy xmodem: flash:filename, como el siguiente ejemplo:

switch: copy xmodem: flash:imagen.bin
Begin the Xmodem or Xmodem-1K transfer now…
CCC

B. Del menú de Hyperterminal que se encuentra en la parte superior se selecciona Transferir > Enviar archivo.

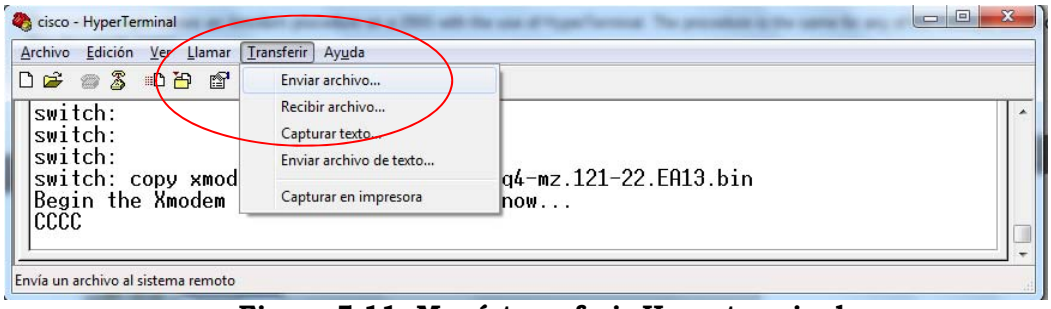


Figura 5.11: Menú transferir Hyperterminal.

C. Se escoge Xmodem como el protocolo y se selecciona el archivo de la carpeta donde se encuentre en la PC.

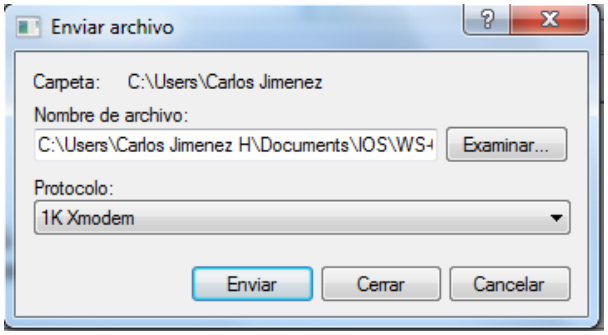


Figura 5.12: Enviar archivo Hyperterminal.

D. Se presiona el botón Enviar para que la transferencia comience.

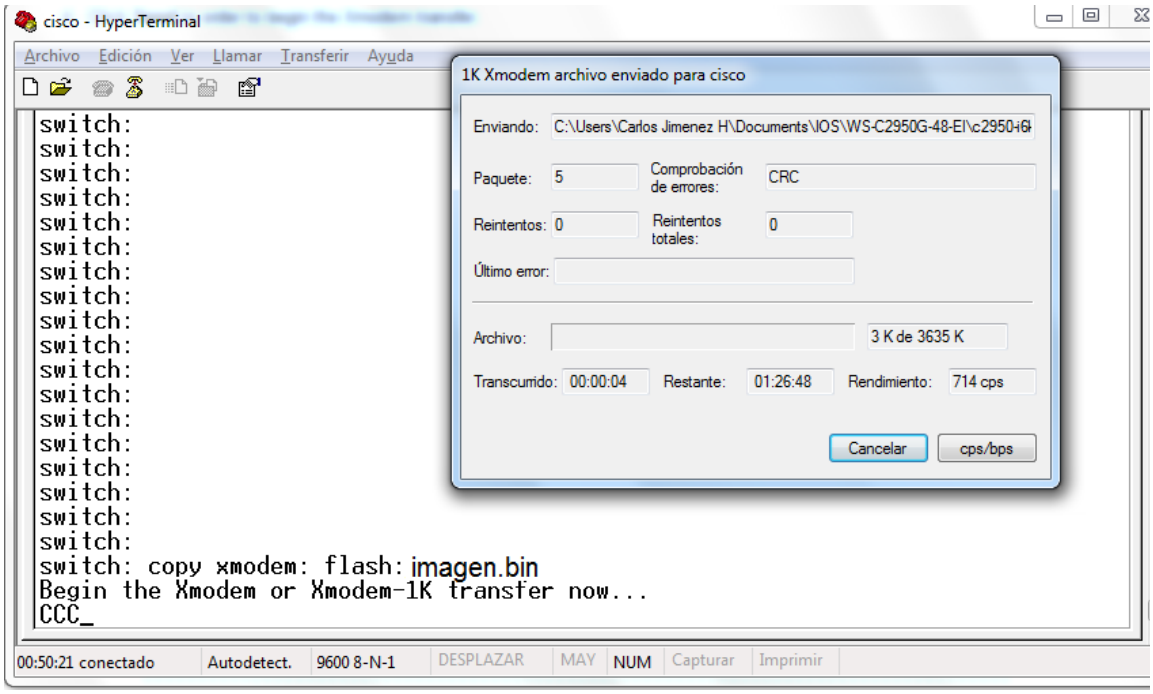


Figura 5.13: Transferencia Hyperterminal.

E. La transferencia comienza y puede tardar hasta dos horas dependiendo del tamaño de la imagen:

CCCCCCC.....
.....
.....
.....
.....
.....
File "xmodem:" successfully copied to "flash: imagen.bin "

F. Ya con la nueva imagen copiada en el equipo se procederá a cargarla con el comando boot:

switch: boot flash: imagen.bin
Loading "flash: imagen.bin "...#################################

Press RETURN to get started!
Switch>

Al finalizar se borrará la vlan creada en el router y se apagará el puerto FastEthernet0/2/0. También se eliminará la cuenta de usuario en la PC de apoyo en caso de haberse creado.

Capítulo VI.

Resultados

El proyecto en su totalidad fue exitoso, se logró actualizar el sistema operativo de más de 600 equipos remotos sin ningún contratiempo. Dado el éxito de los procedimientos utilizados es más fácil poder mantener actualizados los equipos de comunicaciones cada vez que se requiera, se pudo constatar que aun cuando este tipo de actividades son críticas, teniendo un procedimiento adecuado y con las debidas precauciones se puede obtener un resultado satisfactorio.

Gracias a estas actualizaciones se logró configurar el protocolo SSH en todos los dispositivos, lo cual fue uno de los objetivos primordiales por los que se diseñó este proyecto, a continuación se muestran las líneas de configuración necesarias para habilitar el protocolo SSH, comandos que no se encontraban disponibles en las versiones anteriores de IOS.

```
sw-maqueta(config-if)#ip domain-name ife.org.mx
sw-maqueta(config-if)#crypto key generate rsa
sw-maqueta(config-if)#1024
sw-maqueta(config-if)#ip ssh version 2
sw-maqueta(config)#line vty 0 15
sw-maqueta(config-line)#transport input ssh
```

Como parte importante en mi rol en este proyecto, fui partícipe en el diseño e implementación del mismo, se realizaron maquetas donde se probaron todos los procedimientos y se fueron perfeccionando en cada una de las pruebas realizadas. Se descargaron las imágenes IOS de cada uno de los modelos que se tienen, se investigó las vulnerabilidades de cada uno de estos archivos ya que en ocasiones no es recomendable instalar la

imagen más actualizada del Sistema Operativo, sino la versión más estable posible debido a que los equipos requieren estar operando en todo momento y es imprescindible validar que la imagen a instalar no nos vaya a traer problemas en un futuro.

Posteriormente se escogieron algunos sitios los cuales serían considerados *pilotos*, se escogieron los sitios más cercanos, a las Oficinas Centrales del Instituto para que, en caso de contingencia, nos pudiéramos desplazar a sitio para arreglar el desperfecto.

Una vez siendo exitosas las pruebas descritas anteriormente, realice un plan de trabajo donde se actualizarían de manera gradual todos los sitios que serían intervenidos.

A continuación se muestra una representación lógica de lo que comprende la Red Nacional del IFE.

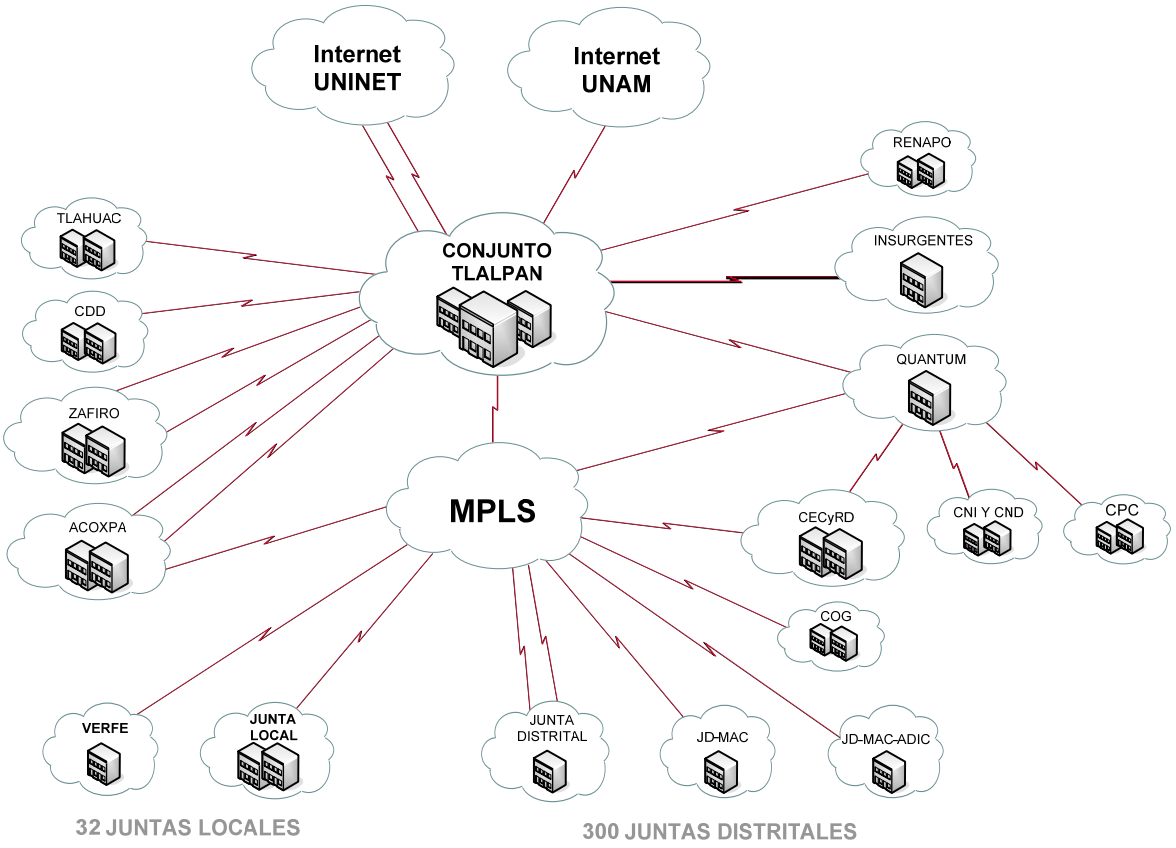


Figura 6.1: Red Nacional del IFE.

Como se observa en la Figura 6.1 la conexión a red de todos los inmuebles del IFE está distribuido ya sea por enlaces punto a punto, como son los inmuebles que se denominan oficinas centrales, los cuales comprenden los conjuntos de Tlalpan, CDD, Zafiro, Acoxpa, etc., y los inmuebles que se encuentran conectados mediante la red MPLS de un proveedor de servicios, entre estos sitios se encuentran todos los órganos delegacionales que existen a lo largo de la República Mexicana, como son las Junta Locales, Distritales, Módulos y Centros de Monitoreo. Todos los sitios se pueden comunicar entre sí y para fines de operación y mantenimiento de la red, todos los segmentos de red pueden ser administrados de manera centralizada. Para la conexión a internet todos los sitios deben tener conexión con los equipos principales de oficinas centrales, éstos a su vez tienen dos conexiones hacia dos proveedores diferentes para la salida a Internet. Debido a que la red del Instituto está interconectada entre sí, es como fue posible realizar todas estas actualizaciones de manera centralizada.

Los equipos que se intervinieron se encuentran en Juntas, Vocalías, Módulos y Centros de Monitoreo, los cuales al ser sitios lejanos a oficinas centrales en su momento de instalación de los equipos y por la urgencia de ponerlos a funcionar en red, sólo se configuró el protocolo Telnet para su acceso.

Capítulo VII.

Conclusiones

Es importante tener actualizado hoy en día los sistemas informáticos y de comunicaciones, esto debido a la vulnerabilidad que pueden tener ya sea por bugs, agujeros de seguridad, adicionar funcionalidades, entre otras. Para este proyecto se tuvo en mente cambiar el método de acceso en los equipos de comunicaciones de capa 2 a nivel nacional, anteriormente se encontraban con el protocolo de acceso Telnet y se quiso migrar a un protocolo más seguro como es el SSH. Para realizar esto se tiene que estudiar si el sistema soporta dicha funcionalidad, como no fue el caso se decidió actualizar el sistema operativo de todos los equipos para que pudieran soportar SSH.

Los procedimientos creados resultaron exitosos lo que nos permitió tener a todos los equipos actualizados, esto a parte de lograr que se cambiara el método de acceso remoto a los equipos, también permitió tener unos procedimientos que nos permiten actualizar de manera fácil y segura cualquier equipo de comunicaciones de la marca Cisco con sistema operativo IOS. De esta manera si existiera algún bug asociado a la imagen instalada, o si se encontrará en la página de Cisco que ciertas imágenes tienen importantes agujeros de seguridad, se puede actualizar sin problema los equipos modificando en menor medida los *scripts* y procedimientos que se realizaron para este proyecto.

Se tienen que hacer revisiones periódicas sobre la seguridad y estabilidad de cualquier equipo informático o de comunicaciones, la información día a día se ha vuelto crítica para poder mantener de manera exitosa cualquier tipo de negocio, es por eso que la seguridad en los últimos años ha tomado

mucha fuerza debido a la importancia de proteger dicha información. Las amenazas pueden ser externas e internas, y particularmente el área en la que laboro tiene que estar monitoreando constantemente la red para evitar cualquier tipo de vulnerabilidad. Se tiene que ser proactivo en vez de reactivo, es por eso que hay que estar actualizado constantemente en innovaciones tecnológicas que nos permitan mantener las comunicaciones seguras, estables, con calidad y disponibles en todo momento.

Referencias

[1] http://www.ife.org.mx/portal/site/ifev2/Que_es/
[2] http://www.ife.org.mx/portal/site/ifev2/Unidad_de_Servicios_de_Informatica/
[3] CCNA EXPLORATION 4.0
[4] ITU-T Rec. X.200 (1994 E)
[5] RFC1930
[6] Lammle Todd, CCNA Cisco Certified Network Associate Study Guide, Seventh Edition.

Glosario

Lista de acrónimos

EGP	Exterior Gateway Protocol
FTP	File Transfer Protocol
IFE	Instituto Federal Electoral
IGP	Interior Gateway Protocol
IOS	Internetwork Operating System
Ipv4	Internet Protocol Versión 4
Ipv6	Internet Protocol Versión 6
LAN	Local Area Network
MAC	Media Access Control
OSI	Open Systems Interconnection
POST	Power-On Self Test
SSH	Secure Shell
TFTP	Trivial Transfer Protocol
UNICOM	Unidad de Servicios de Informática
UPS	Uninterruptible Power Supply
WAN	Wide Area Network