UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Instalación y Configuración de Firewall Unidad Administrativa (DGAE)

INFORME DE ACTIVIDADES PROFESIONALES

Que para obtener el título de

Ingeniero en Computación

P R E S E N T A

López López José Manuel

ASESOR DE INFORME

M. C. Alejandro Velázquez Mena

Ciudad Universitaria, Cd. Mx., 2018

Dedico este proyecto y todo lo que él representa al motor de mi existencia mi hija Alina Yelitza López Salas, quien se ha convertido en el motivo más importante de mi vida personal y profesional; a mis padres que hicieron posible la persona que hoy represento, muchas gracias.

Agradecimientos

A mi hermana Laura por su apoyo incondicional y que es mi brazo derecho en la vida.

A mi esposa Mónica que siempre ha confiado en mí a pesar de todos los obstáculos que hemos pasado.

A mi amigo y hermano: Antonio Amezcua que me ha brindado su amistad, su apoyo y sus consejos desde que éramos niños.

A mi amigo y hermano: Eduardo Miranda que me ayudó a crecer profesionalmente y siempre estuvo ahí para apoyarme.

A mi familia y amigos que me han visto crecer y que jamás me dejaron solo.

Al M.C. Alejandro Velázquez Mena por su apoyo incondicional para la realización de este proyecto.

A la UNAM por darme la oportunidad de convertirme en un Ingeniero.

Índice de Contenido

Introducción

La Dirección General de Administración Escolar (DGAE) se caracteriza por proporcionar información de carácter administrativo y escolar referentes a la UNAM. Su parte administrativa funge como Departamento de Personal, Bienes y Suministros, Contabilidad y Presupuesto, Certificación y Control Documental, Asuntos Escolares del Posgrado, Gestión Estratégica y Primer Ingreso, y Registro Escolar.

La Unidad Administrativa de la Dirección General de Administración Escolar provee de información personalizada de todos los empleados que laboran en la misma, información referente a los presupuestos y contabilidad que se destinan a la DGAE, parte del registro (Primer Ingreso) para los aspirantes a ingresar a la UNAM y de manera importante los Planes de Estudios de todas las carreras impartidas en la UNAM.

El presente informe de actividades profesionales corresponde a la implementación de las Tecnologías de la Información para salvaguardar y proteger información sensible perteneciente a la Dirección General de Administración Escolar (Unidad Administrativa) ubicada en el edificio Rectoría planta baja.

Con la aparición de nuevas amenazas informáticas se ha vuelto indispensable contar con mecanismos de seguridad para proteger información sensible del personal que labora dentro de la institución, y como tal, información referente a la dependencia ante posibles intentos de robo de información.

La Unidad Administrativa ha venido trabajando en un ambiente no controlado por algún mecanismo de seguridad o de monitoreo que pudiese dar información relevante sobre el estado de la red y de los posibles intentos o éxitos de intrusión hacia la misma.

Se presentó la iniciativa de mejorar la infraestructura de red del nodo principal junto con la instalación de un firewall capa 3; ello representaría una mejora en cuanto a la

velocidad de transmisión de datos entre los equipos de usuario y la protección de la red de datos.

Para la implementación del proyecto propuesto se hizo uso de dispositivos y software privativo, combinado también las mejores opciones en cuanto a software libre se refiere; con esto se consiguió una infraestructura adecua a las necesidades básicas en temas de seguridad informática de bajo coste.

Definiendo el alcance, el presente trabajo se divide en:

Capítulo 1. Organigrama.

Capítulo 2. Descripción de Proyectos.

Capítulo 3. Reporte de Instalación y Configuración de Firewall en Rectoría.

Capítulo 4. Resultados.

Capítulo 1. Organigrama

En este capítulo presento la estructura interna de la Dirección General de Administración Escolar, sus funciones como parte de la Universidad Nacional Autónoma de México y el cargo que represento en ella.

Las principales funciones de la Dirección General de Administración Escolar se resumen en:

- Dirigir, coordinar y realizar las actividades de administración escolar en la institución, en base a la normatividad y a los requerimientos de servicio.
- Ejercer las facultades que le confiere la Legislación Universitaria en relación con la administración escolar.
- Definir e implementar los programas para el cumplimiento de los servicios de administración escolar.
- Proporcionar a la comunidad universitaria los servicios de administración escolar, desde el ingreso hasta la titulación.

Para el cumplimiento de sus metas, la DGAE se encuentra organizada en diferentes subdirecciones las cuales cumplen con un propósito específico; en la *Figura 1.1* se muestra dicha organización. Es en parte responsabilidad de la Subdirección de Diseño de Proyectos (SDP) implementar mecanismo de seguridad para:

- Dirección de Gestión Estratégica y Primer Ingreso
- Dirección de Certificación y Control Documental
- Unidad Administrativa

En la *Figura 1.2* se muestra el organigrama de la SDP y mi función como parte de la subdirección.

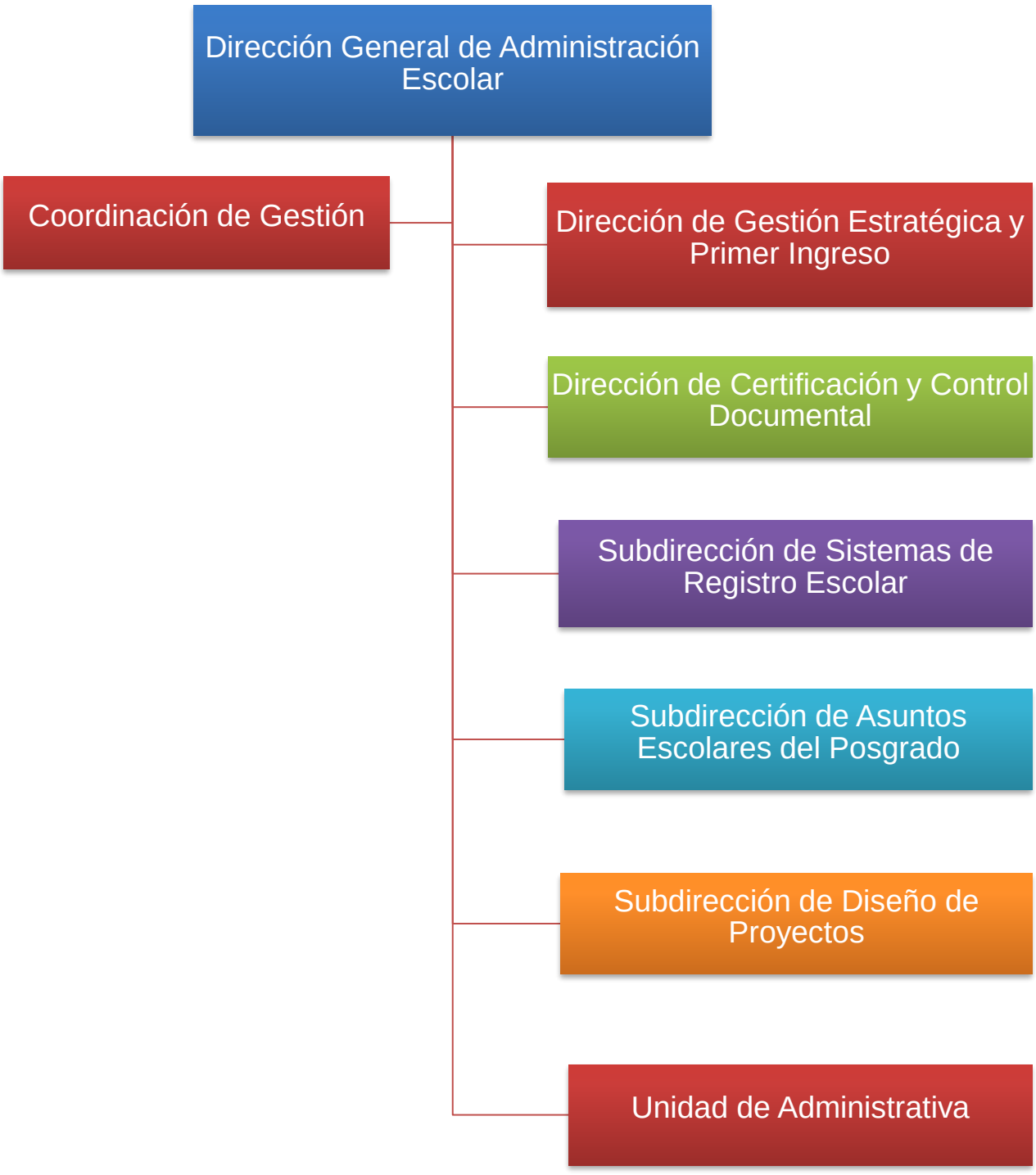

Figura 1.1 - Organigrama Dirección General de Administración Escolar.

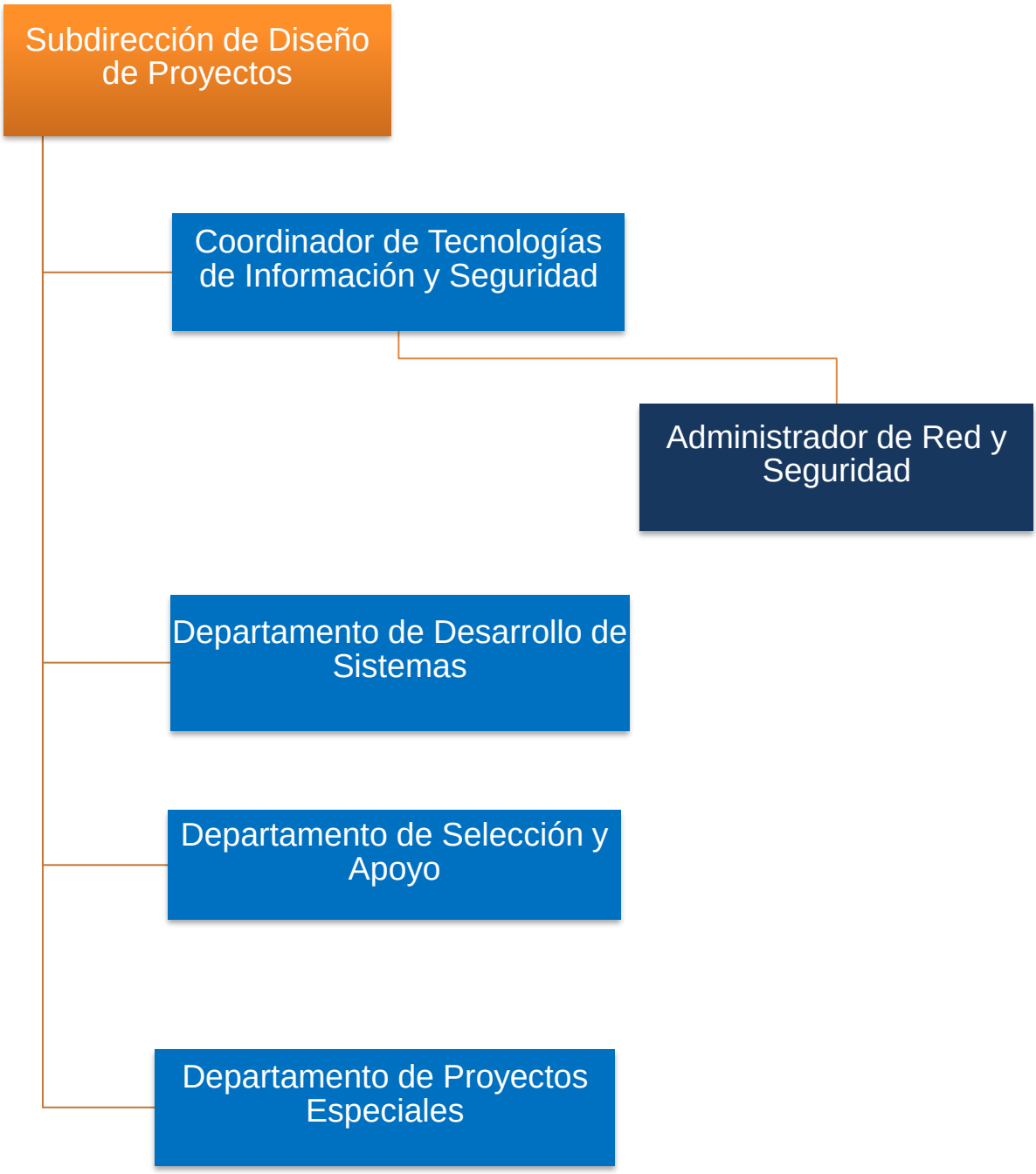

Figura 1.2 - Organigrama Subdirección de Diseño de Proyectos.

Capítulo 2. Descripción de Proyectos

En este capítulo expongo, en una breve descripción, los proyectos de mayor relevancia que realicé durante los 6 años de trabajar para la Subdirección de Diseño de Proyectos (DGAE), los cuales cumplieron con su objetivo para el cumplimiento de las metas de la dependencia.

2.1 Introducción

La Subdirección de Diseño de Proyectos a través de su portal *https://www.dgae.unam.mx* es la encargada de brindar información referente a:

- Convocatoria para los concursos de selección
- Examen COMIPEMS
- Ingreso a Iniciación Universitaria
- Ingreso a Licenciatura por Pase Reglamentado
- Resultados de los concursos de selección
- Trámites y Servicios Escolares en General
- Ubicación de Dependencias de la UNAM
- Venta de Guías y Planes de Estudios

Información que ha crecido en demanda cada año, esto ha llevado consigo el buscar la mejora de los sistemas informáticos para satisfacer las necesidades de la dependencia y cumplir los objetivos que la Dirección General de Administración Escolar ha planteado.

2.2 Firewall-Balanceador de carga para tráfico de red

2.2.1 Antecedentes

La Subdirección de Diseño de Proyectos a través de su portal *https://servicios.dgae.unam.mx* ofrece información referente a las convocatorias para el ingreso a la UNAM mediante la aplicación de examen.

En el portal se presentan las formas de registro para ser acreedor de un lugar al examen, así como también presenta los resultados de dicho examen para los aspirantes que cumplieron con el proceso de selección.

En los últimos años la demanda de aspirantes al registro ha venido en crecimiento, lo cual generaba una carga de tráfico en red mayor a lo esperado; esto sobrepaso las capacidades del único servidor que soportaba el dominio para atender dicha demanda.

2.2.2 Objetivo

Incrementar la cantidad de servidores que alojen el dominio *https://servicios.dgae.unam.mx* para poder brindar una mejor respuesta a los aspirantes en los eventos de ingreso a la UNAM.

Implementar un firewall capaz de filtrar las conexiones de red hacia lo servidores mediante un conjunto de reglas definidas.

Configurar el balanceo de peticiones de red hacia los puertos http (80) y https (443) de los servidores mediante el firewall; éste a su vez debe ser escalable si es que más equipos fuesen necesarios agregar a soportar el dominio.

2.2.3 Desarrollo e Implementación

La primera etapa la desarrollé identificando versiones de software del servidor principal donde se aloja el dominio.

La segunda etapa consistió en la adquisición de nuevos servidores, en donde la opción elegida derivó en reutilizar servidores que dejaron de ser productivos cuando sus servicios que brindaban (correo de alumnos de bachillerato) dejaron de ser administrados por la SDP. Estos servidores variaban del principal en cuanto a la arquitectura del procesador y memoria ram se refiere.

La tercer y cuarta etapa la llevé a cabo con el mantenimiento preventivo, correctivo de los servidores y la instalación de software en ellos similar al servidor principal. En la *Tabla 2.1* se muestran las características de los servidores con los que contaba para implementar el proyecto.

Tabla 2.1 - Tabla de especificaciones de los servidores para balanceo.

Servidor	Modelo	Sistema Operativo	Dirección IP
Principal	SUN SPARC V240	Solaris 10	Homologada
2	SUN SPARC V890	Solaris 10	Homologada
3	SUN X4200 (Intel)	Solaris 10	Homologada
4	SUN X4200 (Intel)	Debian 6	Homologada
5	SUN X4200 (Intel)	CentOS 5	Homologada
6	Armada (Intel)	Debian 6	Homologada

En la quinta etapa del proyecto presenté el esquema del balanceo de red a manejar (ver *Figura 2.1*) y los medios por los cuales los desarrolladores harían similar sus códigos fuente en todos los servidores.

Al término de la presentación del proyecto con los responsables de los departamentos involucrados, se optó por el uso de la aplicación *rsync* para la copía de sus codigos fuente, en la parte web, de todos los servidores.

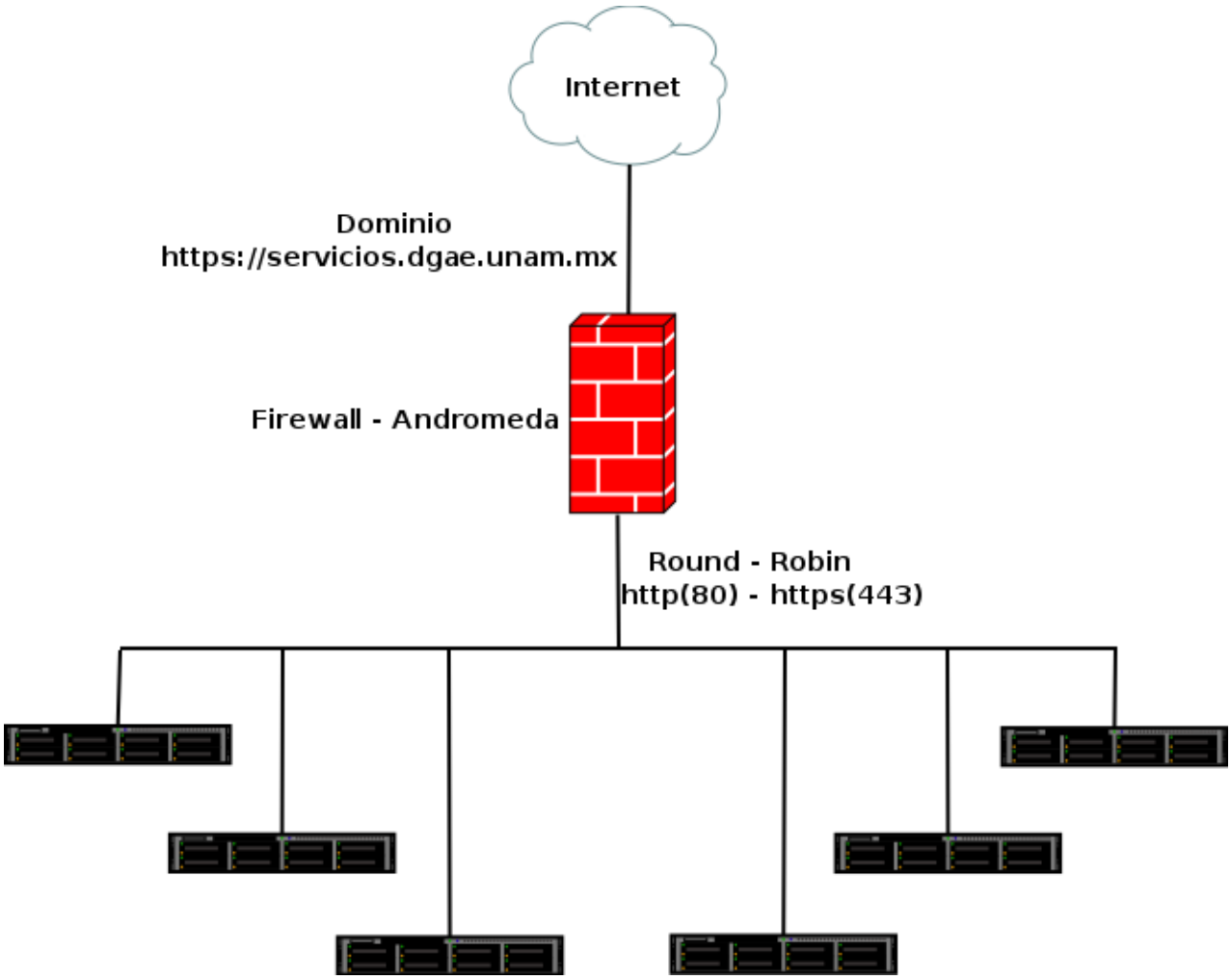

Figura 2.1 - Esquema de balanceo de red propuesto como solución al proyecto.

Dentro de la sexta etapa opté por instalar OpenBSD como Sistema Operativo para implementar el firewall-balanceador; en la *Tabla 2.2* están presentes las actividades que realicé para el cumplimiento de esta etapa.

Tabla 2.2 - Tabla de actividades para la instalación y configuración del firewall-balanceador.

Componente	Descripción
Características del equipo	Con base en la demanda de usuarios por evento, se destinó un equipo de cómputo con las posibles mejores características para el firewall-balanceador.
RAID 1	Configuré 2 discos duros con capacidad de 250GB cada uno en espejo.
Software y red	Instalé paquetes necesarios para la administración del firewall; creé usuarios y configuré direcciones IP a cada interface de red.
Reglas	Creé el archivo principal de reglas generales y de balanceo.
Pruebas de balanceo	Implementé un laboratorio de pruebas simulando el escenario real.
Seguridad	Aseguré las configuraciones de los paquetes instalados en el Sistema Operativo.

La séptima etapa la realicé configurando Apache HTTP Server Project en cada uno de los servidores a balancear; realicé la copía de los archivos necesarios que componen al certificado SSL del servidor principal a los demás servidores.

En la etapa final implementé el escenario real, realicé las últimas pruebas antes de salir a producción y al obtener resultados positivos el firewall-balanceador entro en producción.

2.2.4 Resultados y Conclusiones

La puesta en producción del proyecto la realicé un día antes del registro de aspirantes correspondiente al Concurso de Selección de Febrero 2011; como resultado del monitoreo realizado, se confirmo que las conexiones de red hacia los puertos http (80) y https (443) eran balanceada por round-robin hacia los 6 servidores disponibles para brindar el servicio.

La carga por cada servidor en términos de memoria ram y procesador era menor a la media esperada, las respuestas hacia los aspirantes no se vieron interrumpidas en ningún momento, el firewall no presentó sobrecarga al manejar un ancho de banda de 120Mb/s.

Una parte importante derivada del proyecto fue la detección de conexiones masivas a los servidores para descargar un archivo en formato pdf; la decarga continua de este archivo representaba para los servidores web una entrega de 40GB de información que reflejaba un consumo de ancho de banda no deseado. La solución consistió en la programación de un script en bash que fuese capaz de detectar a partir de las bitácoras del firewall estas conexiones y crear una regla que bloqueara las direcciones IP origen de la conexión.

Los resultados fueron positivos al optimizar los servicios que la Dirección General de Administración Escolar ofrece para uno de los eventos más importantes de la UNAM.

Actualmente la implementación referida sigue en producción, dejando la facilidad de incluir más equipos al balanceo si ello lo requiriese.

2.3 Administración, Monitoreo y Presentación de Recursos TI de la Subdirección de Diseño de Proyectos (SDP)

2.3.1 Antecedentes

Con la implementación de nuevas tecnologías para la mejora de servicios que ofrece DGAE a través de la Subdirección de Diseño de Proyectos, se ha tenido un incremento sustancial en cuanto a dispositivos de red, equipos de seguridad perimetral, servidores dedicados y equipos de usuarios se refiere.

Si bien es cierto que el incremento está justificado, también es cierto que ello ha venido a demandar más recursos humanos para su administración.

Durante los últimos años, la administración de la infraestructura de red que compone a la SDP ha consistido en la revisión manual por individual de cada activo, ello se ha convertido en una tarea que demanda mucho más tiempo y que dificulta una temprana detección de algún posible fallo en los equipos.

La cantidad de usuarios que dependen indirectamente del departamento Coordinación de Tecnologías de Información, Comunicación y Seguridad asciende aproximadamente a 300; dato importante porque cada usuario hace uso de una dirección IP y se ha hecho necesario tenerlos registrados; misma situación con los activos que componen la infraestructura de red SDP.

2.3.2 Objetivos

Implementar mecanismos de monitoreo para una temprana detección de fallos en los activos.

Desarrollar un medio eficiente, presentable, de fácil acceso y digital para el registro de los activos que componen la infraestructura TI SDP.

2.3.3 Desarrollo e Implementación de Cacti

Para la elección de las herramientas a utilizar, realicé un registro de los activos más críticos; después creé un informe con el mayor número de incidentes ocurridos por equipo para así definir qué recursos se iban a monitorear.

Los resultados obtenidos se pueden observar en la *Tabla 2.3*.

Tabla 2.3 - Activos y recursos de infraestructura de red SDP para monitoreo.

Activos	Incidencias frecuentes	Recurso qué monitorear
Switches y Routers	• Ancho de Banda • Interconexión con dispositivos • Uso de VLAN	• Ancho de Banda • Interconexión • VLAN
Firewalls	• Conexiones de Red no establecidas • Bajo Rendimiento	• Ancho de Banda • CPU • RAM • HDD
Servidores	• Bajo Rendimiento • Capacidad máxima de escritura en Disco Duro	• CPU • RAM • HDD • Ancho de Banda
UPS	• Ninguno	• Ninguno
Aire acondicionado	• Ninguno	• Ninguno
PC de usuarios	• Ancho de Banda	• Ancho de Banda

De protocolos y herramientas existentes de Software Libre, las que más satisfacían las necesidades para la implementación del proyecto fueron:

- SNMP (Simple Network Management Protocol)
- RRDtool (round-robin database tool)
- Software Cacti

Cacti por su robustez de integrar RRDtool y SNMP en su funcionamiento, fue la herramienta adecuada para graficar los recursos de los equipos y SNMP al ser multiplataforma no tenía problema de integrarse con los Sistemas Operativos con los que se contaba.

Una vez elegido el equipo que alojaría el proyecto, pasé a realizar las actividades que se muestran en lista en la *Tabla 2.4*.

Tabla 2.4 - Actividades de instalación y configuración para el servidor de monitoreo

Componente	Descripción
RAID 1	2 discos duros (100 GB) en espejo 2 discos duros (300 GB) en espejo
Sistema Operativo	Instalación básica, configuración de usuarios, configuración de red
Entorno web	Instalación: • MySQL • Open SSL • Apache • PHP
Cacti	Instalación: • Net-SNMP • RRDtool • Cacti

Seguridad	• Cerrando puertos no necesarios • Configuración de conexiones ‘ssh’ • Uso de archivos ‘htaccess’ • Uso de Certificados SSL

Realicé las pruebas necesarias para validar las herramientas instaladas; paso siguiente hice la configuración de Cacti vía web; en la *Figura 2.2* se muestra el cuadro de diálogo para el ingreso al aplicativo.

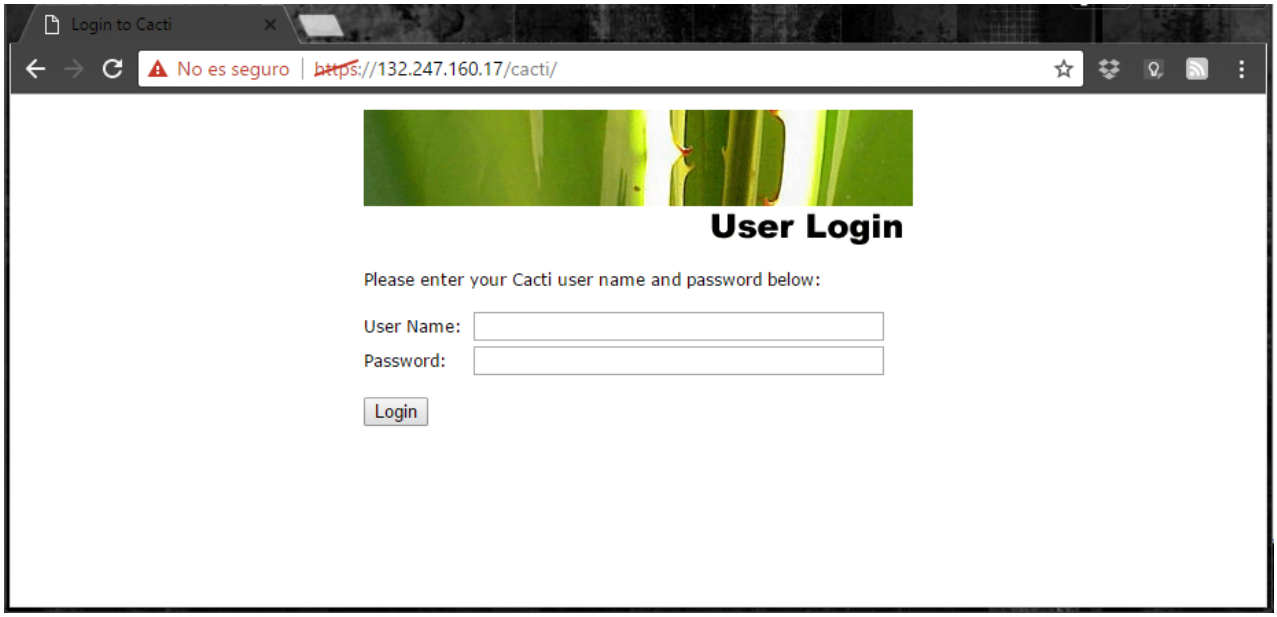

Figura 2.2 - Cuadro de diálogo Cacti vía web para su configuración

La configuración del aplicativo se realiza en mayor parte en automático, ingresando o corroborando datos como las rutas de los ejecutables de los que hará uso,

zona horaria, si se trata de una nueva instalación o una actualización y el cambio de contraseña del usuario administrador por una más robusta. En la siguiente *Figura 2.3* se visualiza la consola de administración donde se agregan los equipos a monitorear y así poder comenzar a graficar los recursos requeridos.

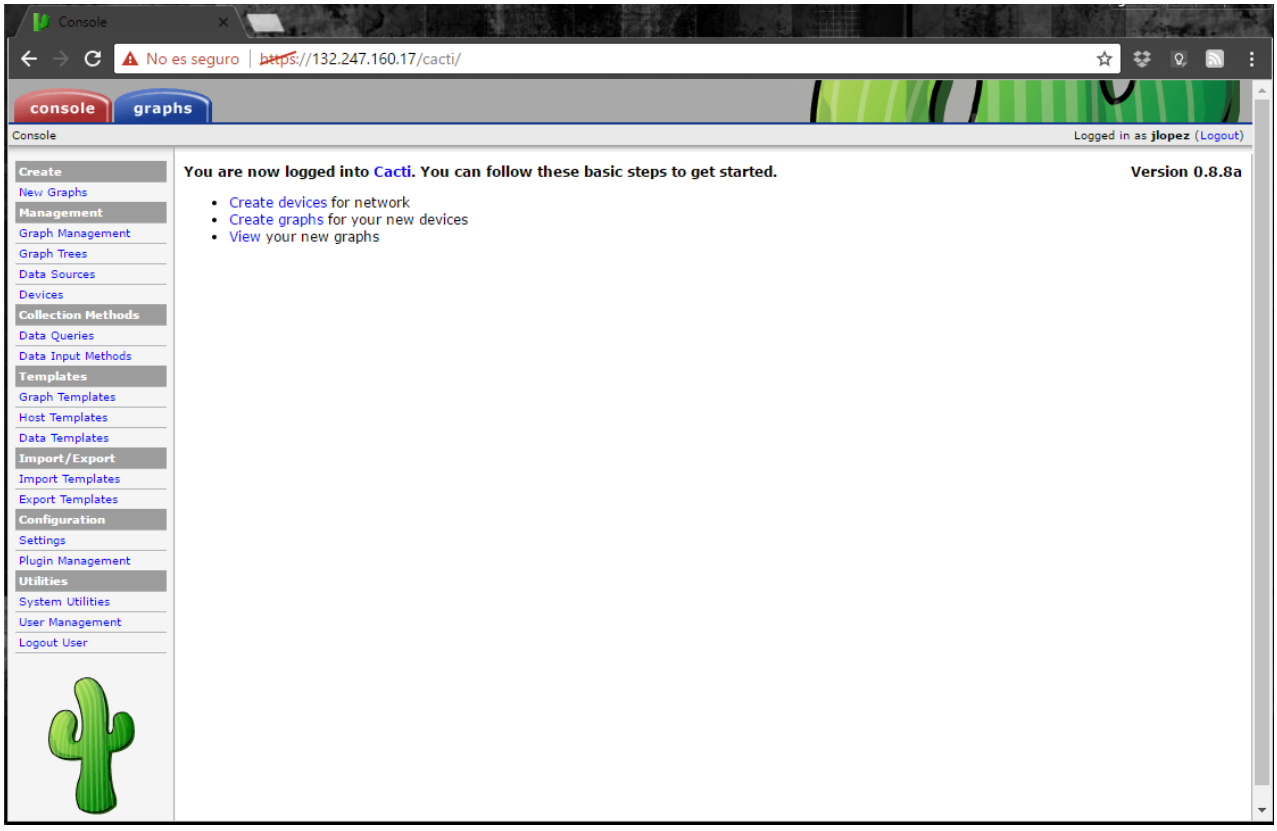

Figura 2.3 - Consola de administración del software Cacti

Lo siguiente consistió en la instalación y configuración del agente SNMP en los equipos a monitorear; esto representó un reto ya que, al tener diferentes arquitecturas de procesador, diferentes Sistemas Operativos y versiones obsoletas de librerías, en la mayoría fue necesario la compilación del código fuente de Net-SNMP.

En la *Figura 2.4* y *Figura 2.5* podemos observar cómo Cacti grafica los recursos de procesador, memoria ram, espacio en disco duro y ancho de banda utilizados por uno de los servidores residentes dentro de la infraestructura de red SDP.

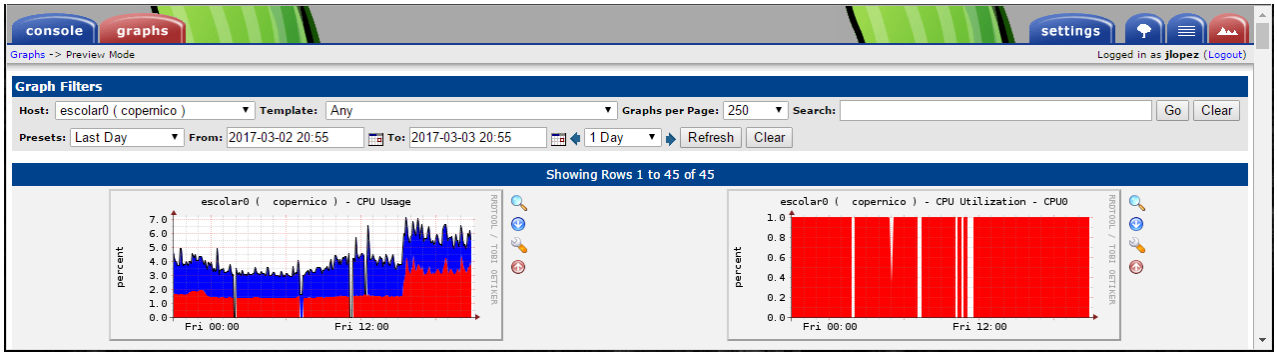

Figura 2.4 - Gráficas del uso de procesador

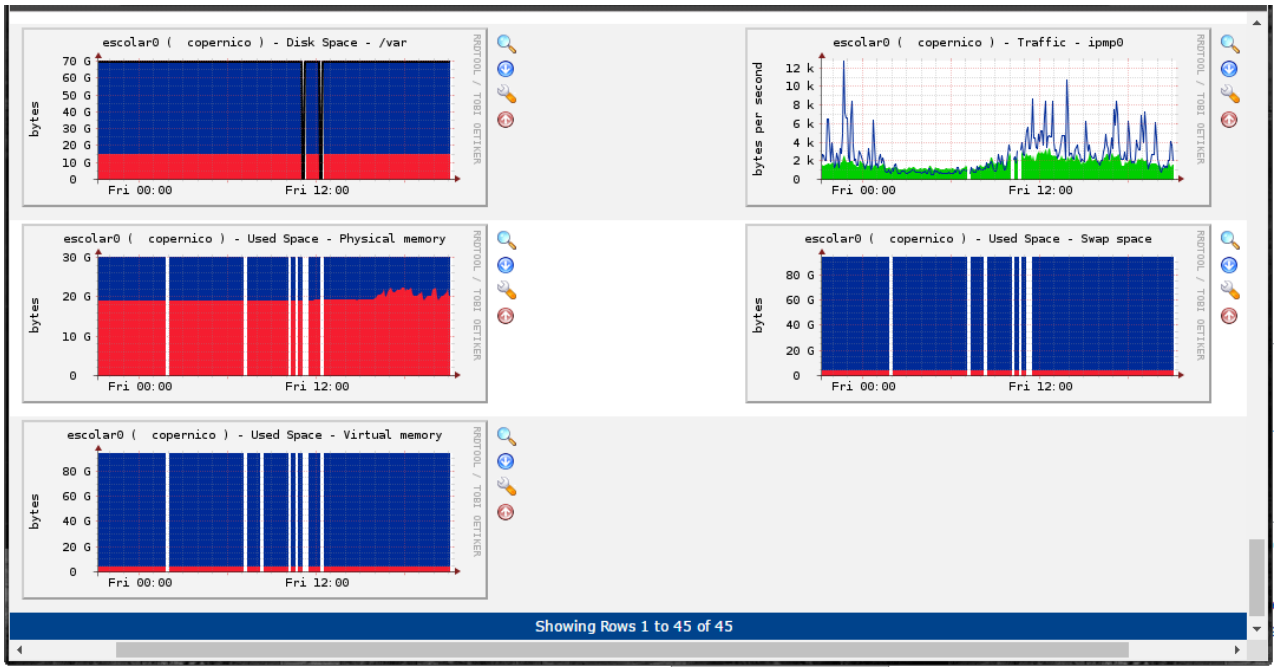

Figura 2.5 - Gráficas memoria ram, disco duro y ancho de banda utilizados

2.3.4 Desarrollo e Implementación SiteAdmin

Para la parte del registro de equipos dentro de la infraestructura de red, direcciones IP y usuarios, creé un sitio web en donde fuese posible llevar el control de altas, bajas y cambios de estos.

El sitio web lo desarrollé haciendo uso de las tecnologías MySQL, Apache y PHP instalados en el servidor donde Cacti se encuentra; en la *Tabla 2.5* se muestra la distribución implementada en el sitio.

Tabla 2.5 - Estructura sitio web SiteAdmin

Sección	Descripción
Inicio	Página principal y presentación del sitio web
Seguridad	Relación de equipos de seguridad perimetral y de red
Red	Relación de puertos (switches, routers) y segmentos de red
Servidores	Descripción, alertas por servidor
Rango de direcciones IP	Relación usuario - dirección IP
Actividades	Proyectos y actividades

Realicé un levantamiento a mano de cada equipo existente en la red, así como sus características e información que debían presentarse dentro del sitio web SiteAdmin.

El acceso al sistema lo restringí mediante el mecanismo básico de autenticación de Apache Server. En la *Figura 2.6* está la captura de pantalla de la página principal de SiteAdmin.

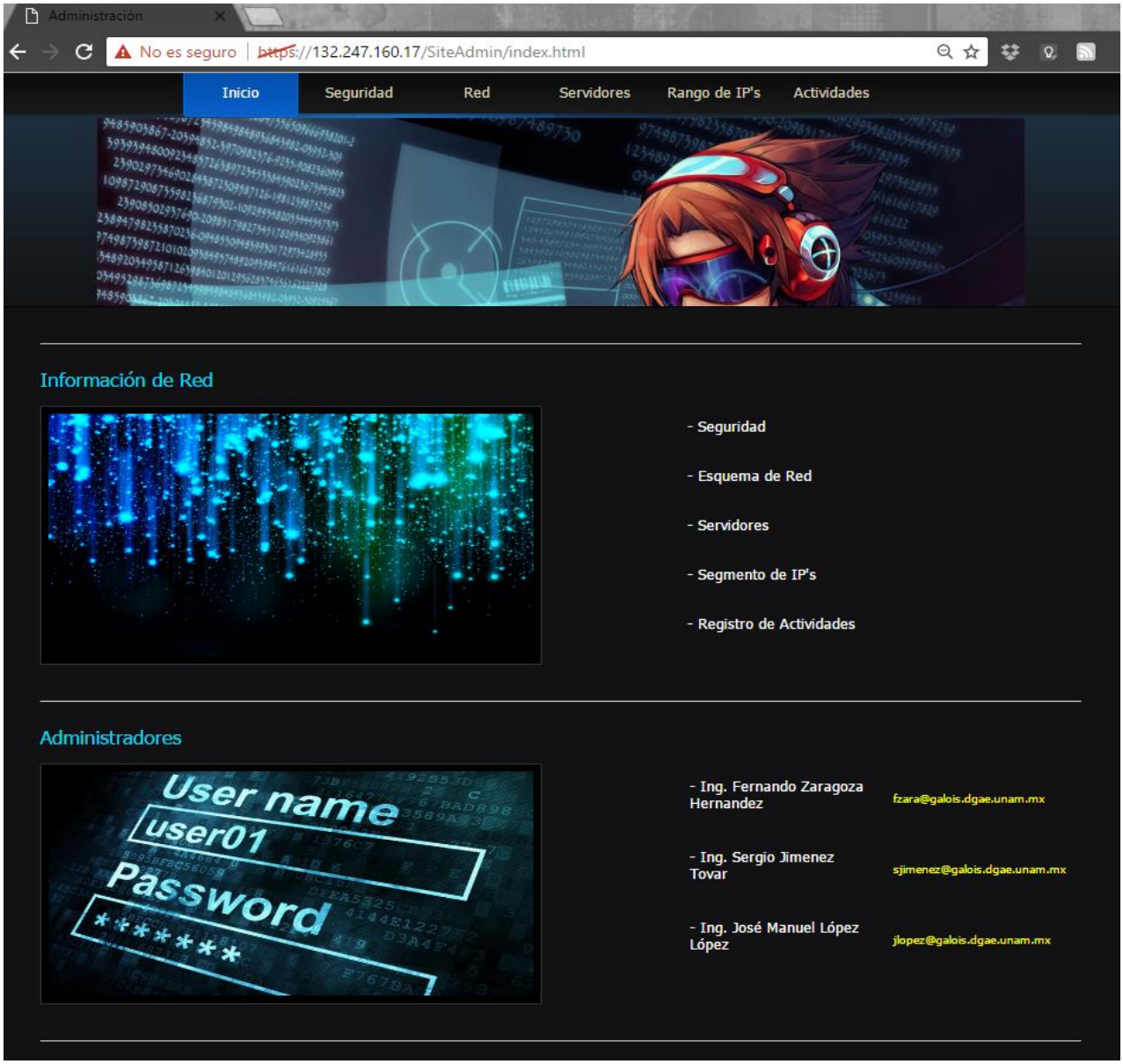

Figura 2.6 - Captura de pantalla página principal del sitio web SiteAdmin

Ahora bien, una vez que el sitio fue puesto en producción, el siguiente paso fue implementar una de las funcionalidades con las que el protocolo SNMP cuenta:

> Trap (Se genera por el agente para reportar condiciones y/o cambios anormales del equipo donde reside.)

El esquema por seguir para procesar los reportes usando Traps, consiste en la existencia de un servidor único que permite a los agentes residentes en los demás equipos enviar sus reportes hacia él y este a su vez se encarga de procesar estas Traps. El servidor encargado de realizar estas acciones corresponde al mismo que soporta SiteAdmin.

La información que llevan las Traps es de los cambios en porcentaje del uso de procesador y espacio en disco duro; cada vez que el procesador rebase el 70% y/o el disco duro llegue al 20% de su capacidad se genera el reporte en el equipo y se envía al servidor central para su procesamiento.

Para el procesamiento de las Traps que llegan sin formato programé un script en bash, el cual solo toma los datos de procesador y/o disco duro, les da formato, los guarda en una Base de Datos y crea un mensaje de texto con fecha, hora y datos del reporte para después enviarlo por correo electrónico a las cuentas de los interesados. Estas alertas también son visibles desde SiteAdmin; en la *Figura 2.7* observamos un ejemplo de una alerta ya procesada.

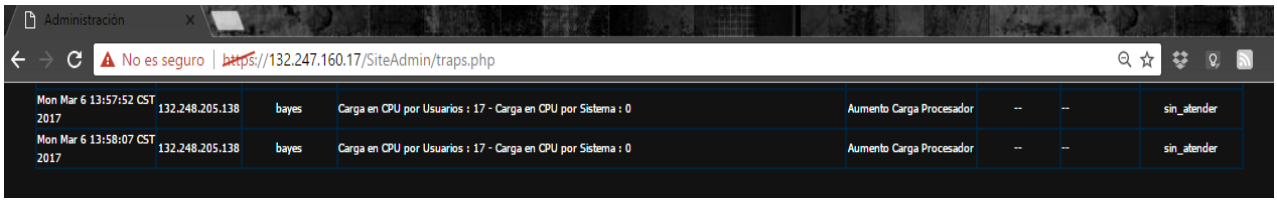

Figura 2.7 - Trap recibida, aumento de carga en procesador

2.3.5 Resultados y Conclusiones

Con la implementación de un sistema capaz de graficar el comportamiento de los equipos de infraestructura en red, se presentó un medio importante como referencia para conocer las capacidades de SDP en cuanto a hardware se refiere; esto generaría nuevas propuestas para la adquisición de equipo más robusto o en su caso el mejoramiento de las aplicaciones residentes en los servidores.

Esto incluye también el uso de ancho de banda total por parte de la SDP y las necesidades que ella pudiera requerir para solventar la gran demanda de usuarios en eventos en los que la DGAE participa.

Con el sitio web SiteAdmin, se logró concentrar y mejorar la forma en administrar altas, bajas o cambios de todos los equipos a cargo de la SDP, así como también la asignación de direcciones IP para los usuarios.

Es importante mencionar que una de las características en conjunto con Cacti y SiteAdmin es el manejo de las Traps, esto representa una alerta temprana ante cualquier incidente que pudiese ocurrir dentro de la red de datos o en los servidores, lo cual es importante conocer y tener registro de ello para si lo requiriese un análisis a futuro. A esto se suman las alertas que son enviadas en tiempo real al correo electrónico de los encargados.

Actualmente Cacti y SiteAdmin han venido operando por más de 3 años; estas implementaciones han venido creciendo en cuanto a datos se refiere, puesto que ahora se incluyen datos y gráficas de más dependencias de la DGAE como Unidad Administrativa en Rectoría, Dirección de Certificación y Control Documental, Dirección de Gestión Estrategia y Primer Ingreso, un servidor del Sistema Integral de Administración Escolar (SIAE), otro servidor de la Coordinación de Desarrollo Educativo e Innovación

Curricular (CODEIC) y por supuesto todo lo relacionado con la Subdirección de Diseño de Proyectos.

Considero importante mencionar que estos dos sistemas son accesibles desde cualquier equipo con acceso a Internet, esto claro se logra ingresando las credenciales correctas para poder realizar las consultas necesarias.

2.4 Migración de Firewalls de la Subdirección de Diseño de Proyectos

2.4.1 Antecedentes

Cuando hablamos de Seguridad Perimetral nos referimos a equipos de primer nivel para protección de los usuarios y de los activos que dan servicio, en este caso la red de la Dirección General de Administración Escolar.

Con la creciente demanda de usuarios, el ancho de banda en red de la Subdirección de Diseño de Proyectos ha tenido un incremento considerable al alojar la página principal de DGAE, lo cual para un equipo de seguridad perimetral con bajos recursos en hardware le ha sido muy difícil de manejar.

Este fue un problema que alcanzó a la SDP. La conexión principal entre los servidores web y las Bases de Datos se realizaba originalmente mediante 4 PCs y 3 servidores de rack como firewalls de bajos recursos:

- Equipos de computo:
    - Tarjetas de red con velocidades de solo 100Mb/s.
    - Memoria ram de 500MB.
    - Espacio en disco duro de 120GB.
    - Firewall Toolkit (sin soporte).
- Servidores de rack:
    - Tarjetas de red con velocidades de solo 100Mb/s.
    - Memoria ram de 500 MB.
    - Espacio en disco duro de 40GB.
    - Firewall Toolkit (sin soporte).

2.4.2 Objetivos

Diseñar e Implementar un nuevo esquema que permita reducir el número de equipos intermediarios para las conexiones de redes externas hacia los servidores internos.

Actualizar o sugerir una mejora para el Sistema Operativo a utilizar.

Implementar el protocolo SNMP en el nuevo esquema para reportar condiciones de procesador y espacio en disco duro; hacer uso de la herramienta Cacti para el monitoreo de los recursos.

Implementar un firewall para la protección de la red de datos de la Dirección de Gestión Estratégica y Primer Ingreso.

2.4.3 Desarrollo e Implementación

Para el diseño de un nuevo esquema unificado y robusto, me basé en las características, funcionalidades e importancia de cada firewall perimetral dentro de la infraestructura de red y en las necesidades de DGEPI (Dirección de Gestión Estratégica y Primer Ingreso) para la protección de su red de datos. En la *Figura 2.8* se presenta el esquema de red anterior SDP; esto fue el punto de partida para plantear un nuevo diseño.

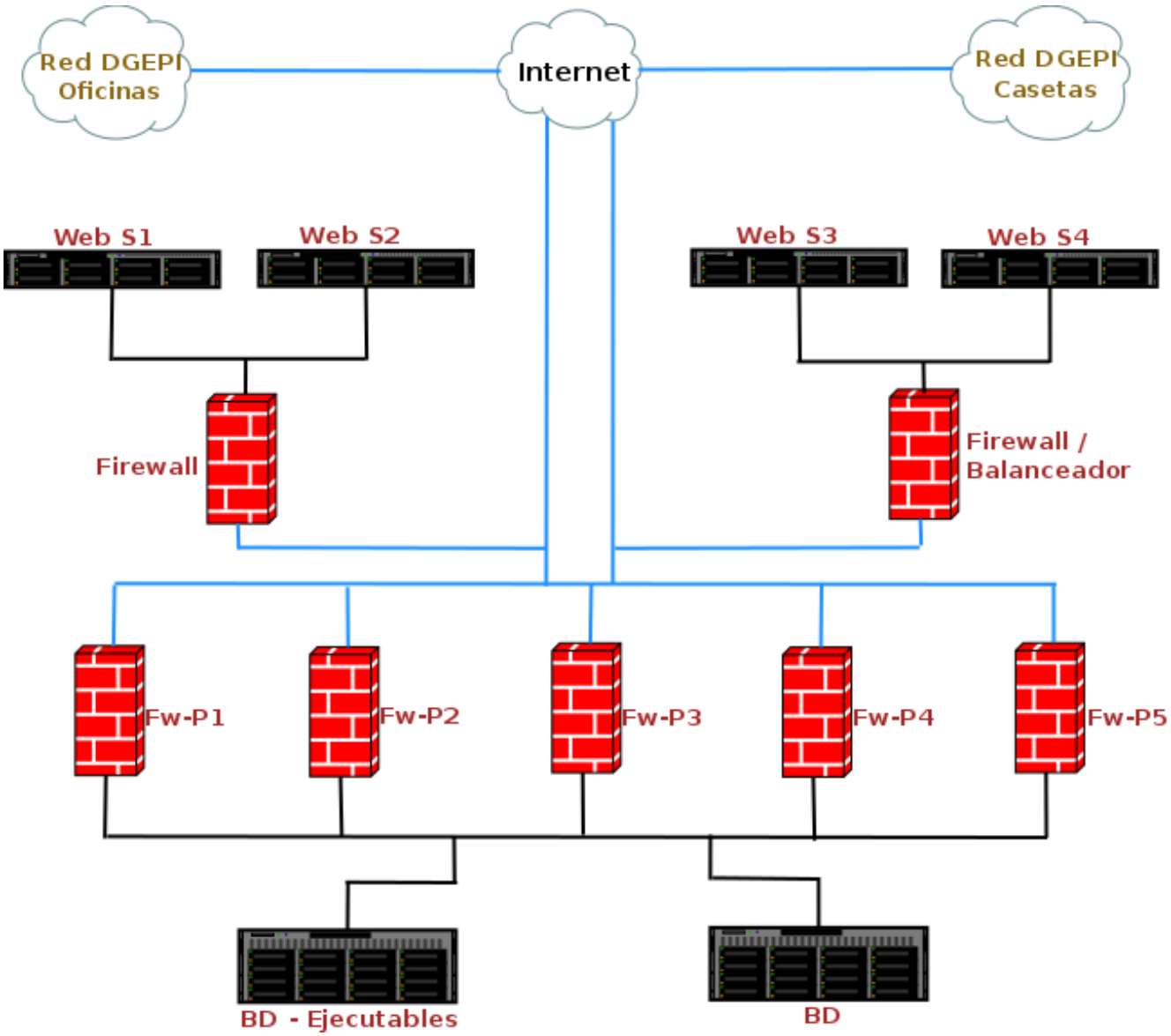

Figura 2.8 - Esquema de red anterior SDP

Un punto importante para el planteamiento del nuevo diseño fue mantener el concepto de red lógico manejado por los firewalls, esto es, la existencia de un equipo de seguridad perimetral para la protección de los servidores web y del balanceo de los correspondientes, un equipo de seguridad perimetral como intermediario entre las conexiones de servidores web, conexiones externas hacia los servidores internos que alojan las Bases de Datos. Además, la implementación de otro equipo de seguridad perimetral para la protección de la red de datos DGEPI.

Ahora bien, lo primero fue definir el Sistema Operativo con el que trabajaría, opté por OpenBSD debido a su robustez, seguridad y constantes actualizaciones; una vez definido el S.O. pasé a la planeación, diseño y presentación del nuevo esquema de red para SDP. En la *Figura 2.9* observamos el diseño que presenté y que aprobaron para la protección de los servidores internos, usuarios SDP y la red de datos (oficinas) DGEPI.

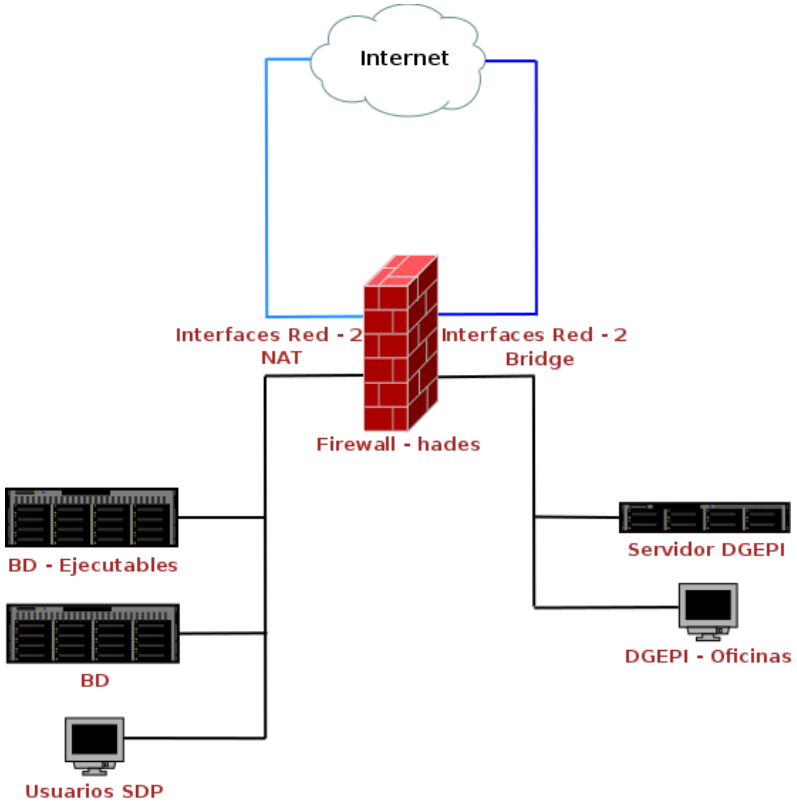

Figura 2.9 - Esquema de red y seguridad del proyecto para su implementación

De igual manera en la *Figura 2.10* presento el diseño aprobado e implementado para la protección de los servidores web y de balanceo, así como la red de datos (casetas) DGEPI.

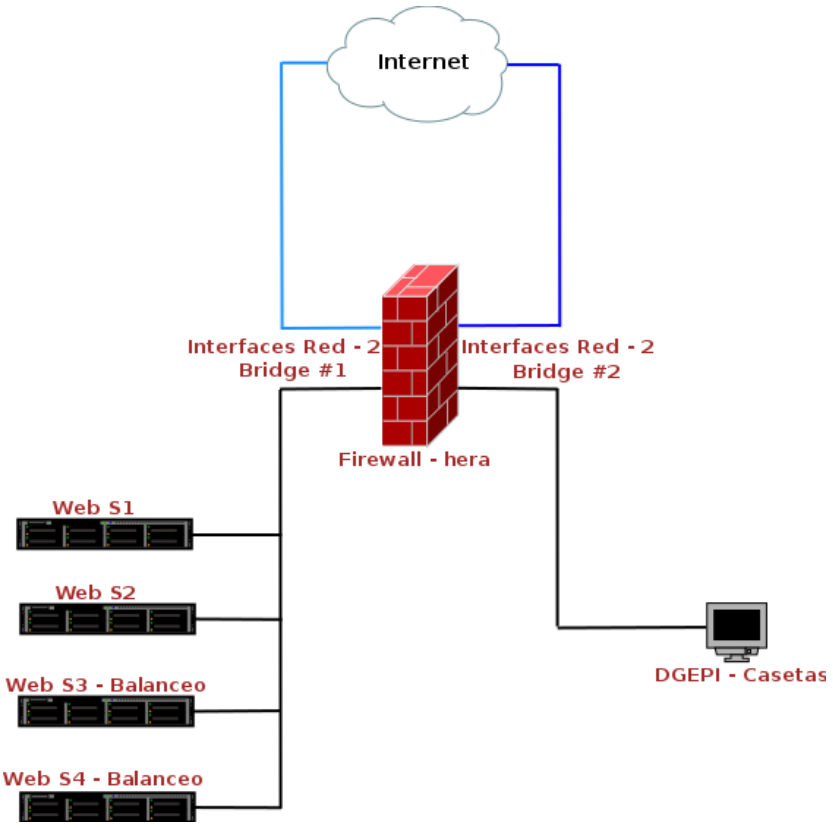

Figura 2.10 - Esquema de red y seguridad del proyecto para su implementación

Los esquemas mostrados recaen principalmente en las capacidades de hardware de los equipos utilizados, ello permitió agrupar todo el conjunto de reglas en solo dos firewalls; parte importante del proyecto recae en el uso de Virtual Local Area Netwrok (VLAN) utilizadas dentro de la infraestructura de red SDP.

En la *Tabla 2.6* hago un resumen general de las actividades que llevé a cabo para la instalación, implementación y pruebas de los nuevos equipos firewall perimetrales.

Tabla 2.6 - Actividades realizadas para migración de los firewalls SDP

Firewall perimetral	Actividad	Descripción
Hades	Características del equipo	• 128 GB en RAM • 64 procesadores virtuales • 4 interfaces de red (1000Mb/s)
	RAID 1	• 1 TB (2 HDD - 1 TB c/u) • 2 TB (2 HDD - 2 TB c/u)
	Sistema Operativo	• Instalación de utilerías, configuración de usuarios y direcciones Ip
	Reglas	• Configuración del archivo principal • Configuración y migración de reglas - NAT - Bridge
	Pruebas	• Pruebas de NAT - Conexiones externas - Conexiones de usuarios • Pruebas de Bridge - Conexiones usuario oficinas DGEPI
	Seguridad	• Asegurar los procesos del sistema operativo - Archivo de configuración ssh - OpenVPN

Hera	Características del equipo	• 64 GB en RAM • 64 procesadores virtuales • 4 interfaces de red (1000Mb/s)
	RAID 1	• 1 TB (2 HDD - 1TB c/u) • 2 TB (2 HDD - 2TB c/u)
	Sistema Operativo	• Instalación de utilerías, configuración de usuarios y direcciones IP
	Reglas	• Configuración del archivo principal • Configuración de Reglas - Bridge #1 - Bridge #2
	Pruebas	• Pruebas Bridge #1 - Conexiones de servidores web - Conexiones de servidores web balanceados • Pruebas Bridge #2 - Conexiones de usuarios casetas DGEPI
	Seguridad	• Asegurar los procesos del Sistema Operativo - Archivo de configuración ssh

El primer equipo en salir a producción fue ‘*hades’*, esto lo hice 3 semanas antes del Concurso de Selección Febrero 2016 en donde tuvo una gran participación; 2 meses después correspondió turno a ‘*hera’*, su participación relevante se dio en la entrega de resultados del mismo concurso.

2.4.4 Resultados y Conclusiones

Con la implementación de los nuevos firewalls los beneficios obtenidos fueron:

- Mejor administración de los equipos al solo contar con 2 de ellos.
- No se presentan problemas de recursos en hardware por parte de los equipos.
- El análisis de las bitácoras es más completo al concentrarlos en un solo archivo.
- El manejo de reglas es más eficiente al haber construido una configuración escalable.
- Se tiene un monitoreo constante de los nuevos firewalls con el uso de Cacti y SNMP.
- Se cuenta con mayor espacio en disco duro para conservar las bitácoras por más de un año para un posterior análisis
- Las interfaces de red soportan un ancho de banda de 1000Mb/s

Los equipos han trabajado por más de 1 año; año en el cual se han ido migrando las conexiones de más servidores, usuarios y dependencias de la UNAM a los nuevos firewall. Esto representa una mejora y un cambio importante para la SDP en cuanto a TI y seguridad nos referimos.

Otro punto que destacar es la implementación de una red privada virtual (Virtual Private Network, VPN) con OpenVPN; esto ha permitido, solo a los administradores, una mejora en seguridad al momento de conectarse desde cualquier lugar a la red interna para la administración de la infraestructura.

Capítulo 3. Instalación y Configuración de Firewall Unidad Administrativa (DGAE)

Este capítulo se enfoca más en los detalles de la realización del proyecto principal, objetivo de este escrito.

Primero presento los objetivos y responsables para la realización del proyecto; más adelante, presento la propuesta de red física, lógica y cómo se llevó acabo su implementación para cumplir con los objetivos planteados.

3.1 Objetivo

Configurar un switch de capa 3 acorde a las necesidades de la Unidad Administrativa.

Concentrar el tráfico de red generado por la Unidad Administrativa en un firewall; aplicar políticas de acceso a internet a solicitud de los responsables del personal que labora en el área mencionada.

Negar todo tráfico de red proveniente de internet; solo se permitirá tráfico de red que solicite algún servicio web existente dentro de la red a proteger.

3.2 Responsables

El proyecto se llevó acabo involucrando personal de la Subdirección de Diseño de Proyectos:

a) Lic. Balfred Santaella Hinojosa (subdirector SDP)
b) Ing. Fernando Zaragoza Hernández (coordinador TICS)
c) José Manuel López López (responsable del proyecto)

3.4 Infraestructura de Red

3.4.1 Red y Políticas de Acceso

En la *Tabla 3.1* se encuentra la información de red destinada a la Unidad Administrativa.

Tabla 3.1 - Información segmento de red destinado a Unidad Administrativa.

Datos de Red	
Segmento de Red	132.248.xxx.xx8
Máscara de Red	255.255.255.xxx
Representación 'CIDR'	132.248.xxx.xx8/xx
Gateway Enlace de Fibra Óptica	132.248.xxx.252
Gateway Enlace de Fibra Óptica	132.248.xxx.253
Gateway Principal	132.248.xxx.254
Broadcast	132.248.xxx.255
Rango de Ip´s Utilizables	132.248.xxx.xxx-xxx

La *Tabla 3.2* muestra la relación de direcciones IP y departamentos de la Unidad Administrativa que realicé para un control más óptimo.

La tabla incluye las políticas de acceso hacia internet para cada usuario; ellas fueron definidas por cada director de área para su personal a cargo. Es importante mencionar que estas políticas pueden variar de acuerdo con las necesidades que surjan a mediano plazo; lo mostrado en la tabla es lo que se configuró y entregó en cuanto a políticas se refiere.

Tabla 3.2 - Relación de direcciones IP asignadas a los departamentos de Unidad Administrativa.

Dirección IP	Usuario	Reglas de acceso
132.248.xxx.xx8	Segmento de red	
132.248.xxx.xx9	Director General	**Total**
132.248.xxx.xx0	Coordinación de Gestión	**Total**
132.248.xxx.xx1	AccessPoint	**Total**
132.248.xxx. [xx2 - xx4]	Direcciones IP reservadas	
132.248.xxx.xx5	Jefe de la Unidad Administrativa	**Total**
132.248.xxx. [xx6 - xx2]	Usuarios	**Restringido**
132.248.xxx.xx3	Jefe del Departamento de Bienes y Suministros	**Total**
132.248.xxx. [xx4 - xx6]	Usuarios	**Restringido**
132.248.xxx.xx7	Jefe del Departamento de Personal	**Total**
132.248.xxx.xx8	Jefe del Departamento de Contabilidad y Presupuesto	**Total**
132.248.xxx.xx9	Servidor SIRF	**Restringido**
132.248.xxx. [xx0 - xx4]	Direcciones IP reservadas	
132.248.xxx.xx5	Dirección de Gestión Estratégica y Primer Ingreso	**Total**
132.248.xxx.xx6	Jefe del Departamento de Coordinación y Seguimiento	**Total**
132.248.xxx. [xx7 - xx1]	Usuarios	**Total**
132.248.xxx. [xx2 - xx4]	Direcciones IP reservadas	
132.248.xxx.xx5	Departamento de Planes y Programas de Estudio	**Total**
132.248.xxx. [xx6 - xx7]	Usuarios	**Total**

132.248.xxx. [xx8 - xx3]	Usuarios	**Restringido**
132.248.xxx. [xx4 - xx9]	Direcciones IP Libres	
132.248.xxx.xx0	Dirección Ip de administración del switch	
132.248.xxx.xx1	Firewall - hermes	
132.248.xxx.xx2	Gateway - enlace de fibra óptica	
132.248.xxx.xx3	Gateway - enlace de fibra óptica	
132.248.xxx.xx4	Gateway - segmento de red	
132.248.xxx.xx5	Broadcast	

3.4.2 Esquemas de Red Físico y Lógico

Las *Figuras 3.1, 3.2* y *3.3* son los esquemas que presenté para el nuevo diseño de la red de datos; estos fueron aprobados para su implementación después de un análisis por los involucrados.

- Esquema lógico

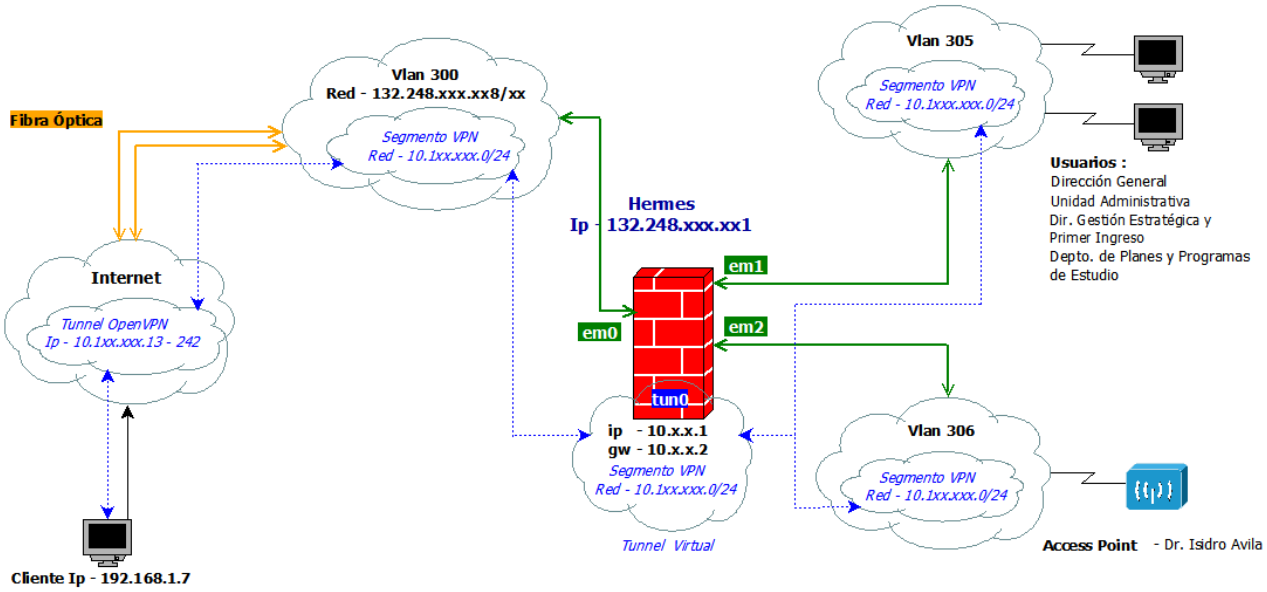

Figura 3.1 - Esquema de red lógico para el nuevo diseño de la red de datos

- Esquema físico

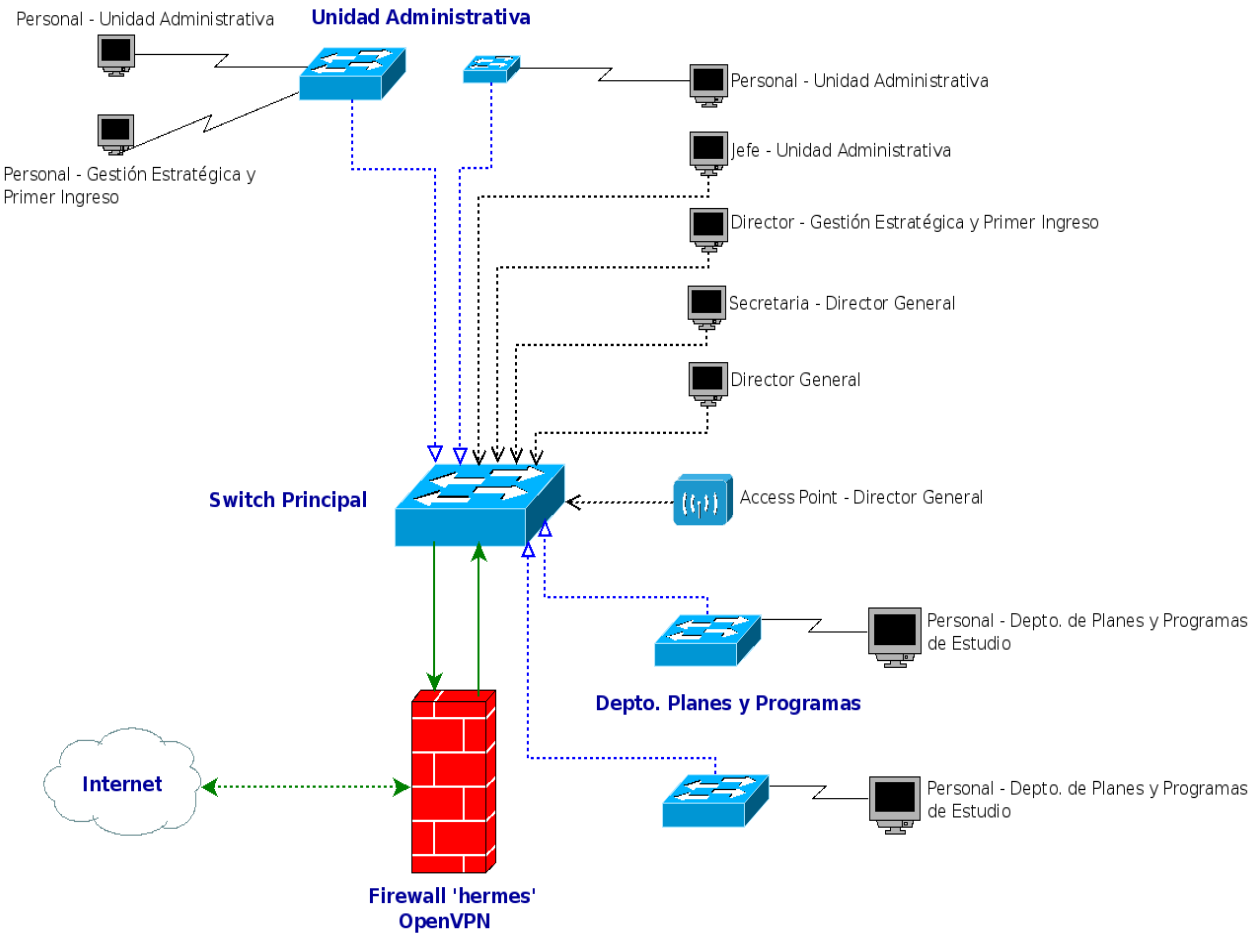

Figura 3.2 - Esquema de red físico para el nuevo diseño de la red de datos

- Distribución RACK IDF

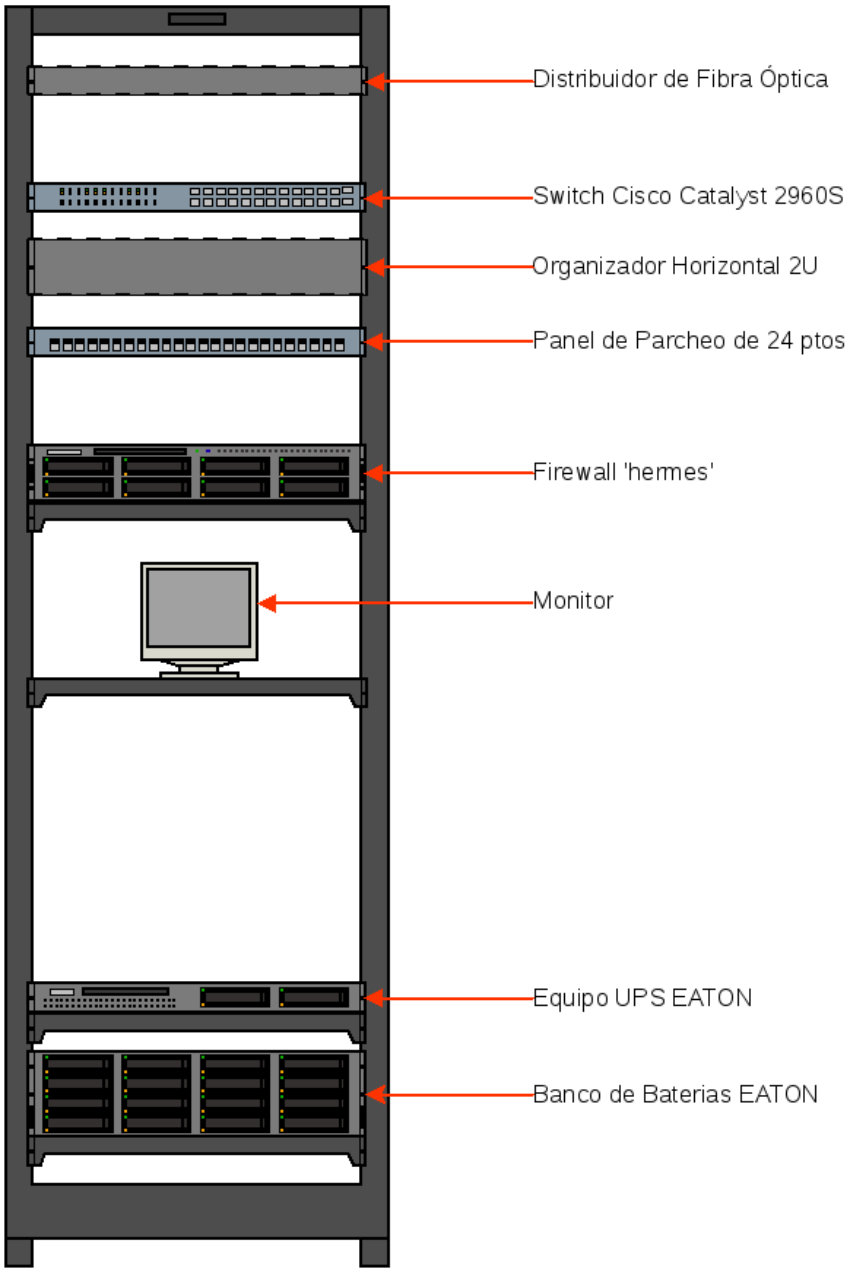

Figura 3.3 - Distribución de equipos en Rack IDF para el nuevo diseño de la red de datos

3.4.3 Características de los Equipos de Red

Para la adquisición de nuevo equipo, entregué una lista (ver abajo) con las características mínimas en hardware para su compra; estas especificaciones fueron basadas a las dimensiones del proyecto.

- Cisco Catalyst 2960S-24TS-L
  - Capa 2 y 3
  - Backplane (capacidad mínima de 100 Gbps)
  - Memoria Flash externa o interna
  - Interfaces 24 puertos 10/100/1000 Base T
  - Interfaces 4 puertos 1000 Base X tipo SFP (soporta Base T, SX y LX)
- Equipo de cómputo para firewall
  - 16 GB RAM
  - Procesador AMD Opteron 6376
  - Controladora RAID
  - 1 TB de almacenamiento
  - Interfaces 4 puertos 10/100/1000 Mb/s

La parte de adquisición y puesta física del switch fue hecha por un proveedor; en esta parte se solicitó contar con todo lo necesario para su instalación y etiquetado.

De igual manera fue la adquisición del equipo para firewall, a través de otro proveedor. Ambas adquisiciones se obtuvieron con un contrato de mantenimiento que permitiría una temprana mitigación de fallos en hardware.

3.2 Instalación y Configuración del Firewall

3.2.1 Sistema Operativo

Para la instalación del Sistema Operativo definí una lista de las configuraciones que realizaría para la implementación del firewall:

- Información general del firewall
    - OpenBSD 5.6 amd 64
    - Hostname *hermes*
    - Dirección IP 132.248.xxx.xx1
- Construir el almacenamiento desde la interface consola Web BIOS proporcionada por la controladora raid del equipo de cómputo para el firewall.
    - RAID level 1 (mirroring)
        - Grupo 0 (1TB); 2 discos duros de 1 TB cada uno
        - Grupo 1 (2TB); 2 discos duros de 2 TB cada uno
- Esquema de particiones
    - 1 HDD (1TB - sd0)
        - / - 15 GB
        - /tmp - 7 GB
        - /usr - 70 GB
        - /var - 70 GB
        - /home - 780 GB
        - Swap - 82 GB
    - 1 HDD (2TB - sd1)
        - /sistema - 1 TB
        - /opt - 1 TB

- La instalación del Sistema Operativo la llevé a cabo desde una imagen net-install de disco y no directamente de red.
    - El sistema operativo reconoció sus 4 Interfaces de red como:
        - em0 - #0 física
        - em1 - #1 física
        - em2 - #2 física
        - em3 - #3 física
    - Durante la Instalación se definieron los siguientes parámetros
        - Hostname
        - Dominio
        - Dirección IP de las interfaces
        - Usuarios
        - Particiones

Ahora bien, una vez instalado el Sistema Operativo realicé la instalación de los paquetes básicos a utilizar, para ello:

- Exporté en una variable de ambiente la dirección de donde obtendría los paquetes

```
# export PKG_PATH=ftp://ftp.openbsd.org/pub/OpenBSD/5.6/packages/amd64
```

- Obtuve información de cada paquete a instalar:

```
# pkg_info bash
Information for
ftp://ftp.openbsd.org/pub/OpenBSD/5.6/packages/amd64/bash-4.3.18.tgz
Comment:
GNU Bourne Again Shell
….................
….................
aintainer: Christian Weisgerber <naddy@openbsd.org>
WWW: http://cnswww.cns.cwru.edu/~chet/bash/bashtop.html
#
```

- Llevé a cabo la instalación y personalización del software bash para el usuario root:

```
# pkg_add bash
quirks-2.9 signed on 2014-07-31T22:37:55Z
quirks-2.9: ok
bash-4.3.18:libiconv-1.14p1: ok
bash-4.3.18:gettext-0.19.1p0: ok
bash-4.3.18: ok
# pwd
/root
# vi .bashrc
PS1="\[\033[1;37m\][\[\033[1;37m\]\u\[\033[1;37m\]@\[\033[1;31m\]\h\[\03
3[1;33m\]\[\033[1;37m\]-\[\033[1;34m\]\w\[\033[1;37m\]]#\[\033[1;36m\] "
# bash
[root@hermes-~]#
```

- Por último, instalé y personalicé el software vim además de otros paquetes restantes:

```
[root@hermes-~]# pkg_add vim
[root@hermes-~]# vim .vimrc
syntax on
set autoindent
[root@hermes-~]#
[root@hermes-~]# pkg_add ruby-2.1.2.tgz
[root@hermes-~]# pkg_add nmap
```

3.2.2 Configuración de Red

Para la asignación de direcciones IP creé los archivos de cada interface de red em0, em1 y em2:

- Interface em0 (tarjeta externa)

```
[root@hermes-/etc]# vim hostname.em0
# gaara
# Tarjeta externa -- em0 ( Fisica #0 )
inet 132.248.xxx.xx1 255.255.255.xxx NONE
[root@hermes-/etc]#
```

  - La dirección IP del firewall al exterior 132.248.xxx.xx1
  - Para la configuración planteada en el proyecto, la interface de red externa em0 no solo escucha la dirección IP asignada, sino que también es capaz de escuchar las demás direcciones IP públicas que se encuentren activas detrás de las interfaces de red internas em1 y em2.

- Interface em1 (tarjeta interna 1)

```
[root@hermes-/etc]# vim hostname.em1
# gaara
# Tarjeta interna -- em1 ( Fisica #1 )
inet 132.248.xxx.xx1 255.255.255.xxx NONE
[root@hermes-/etc]#
```

  - La interface em1 es el puente para el tráfico de red de todos los usuarios de DGAE en Rectoría.

- Interface em2 (tarjeta interna 2)

```
[root@hermes-/etc]# vim hostname.em2
# gaara
# Tarjeta interna -- em2 ( Fisica #2 )
inet 132.248.xxx.xx1 255.255.255.xxx NONE
[root@hermes-/etc]#
```

  - La interface em2 es el puente para el tráfico de red de los usuarios conectados al Access Point (punto de acceso inalámbrico).

Una vez creados los archivos de las interfaces de red, creé el puente que los conecta lógicamente.

```
[root@hermes-/etc]# vim hostname.bridge0
# gaara
add em0
add em1
add em2
up
[root@hermes-/etc]#
```

Definí los archivos correspondientes al gateway, dns y hosts.

```
[root@hermes-/etc]# vim mygate
132.248.xxx.xx3
[root@hermes-/etc]#


[root@hermes-/etc]# vi resolv.conf
# gaara
# DNS
lookup file bind
nameserver 132.248.10.2
nameserver 132.248.204.1
[root@hermes-/etc]#
```

```
[root@hermes-/etc]# cp hosts hosts_18-Feb-2015
[root@hermes-/etc]# vim hosts
###################################
# gaara
# ( 18 - Febrero - 2015 )
###################################

#Hermes
::1 localhost
127.0.0.1 localhost
132.248.xxx.xx1 hermes.dgae.unam.mx hermes loghost

#Ares
132.247.160.17 ares.dgae.unam.mx ares
132.247.160.17 Ares.dgae.unam.mx Ares
[……]
[root@hermes-/etc]#
```

Para hacer uso de NAT en el firewall, habilité en el kernel la bandera de *ip forwarding* (routing) para IPv4. Esto lo hice de la siguiente manera:

```
[root@hermes-/etc]# cp sysctl.conf sysctl.conf_25-Abril-2015
[root@hermes-/etc]# vim sysctl.conf
###################################
# gaara
# ( 25 - Abril - 2015 )
###################################
machdep.lidsuspend=1 # Try to suspend on lid close
net.inet.ip.forwarding=1 # 1=Permit forwarding (routing) of IPv4
packets
[root@hermes-/etc]#
```

- Para que los cambios se vieran reflejados reinicié el equipo; era posible ejecutar el comando *sh /etc/netstart* para que los cambios los realizara en el momento, pero lo recomendable fue reiniciar el equipo.

3.2.3 Registro de Paquetes Filtrados

Fue indispensable contar con una bitácora de todo lo que estaba filtrando nuestro firewall para futuras consultas; esto lo logré haciendo uso del demonio *syslog*.

- Creé la estructura de directorios donde se guardarían tanto la bitácora actual como las bitácoras de respaldo.

```
[root@hermes-/sistema]# pwd
/sistema
[root@hermes-/sistema]# mkdir pf
[root@hermes-/sistema]# cd pf
[root@hermes-/sistema/pf]# mkdir log
[root@hermes-/sistema/pf]# cd log/
[root@hermes-/sistema/pf/log]# pwd
/sistema/pf/log
[root@hermes-/sistema/pf/log]#
```

- Creé un script que permitiría registrar en un archivo, independiente al que activa el sistema, los paquetes filtrados por Packet Filter.

```
[root@hermes-/etc]# pwd
/etc
[root@hermes-/etc]# vim pflogrotate
#!/bin/bash

####################################

# gaara
# ( 25 - Abril - 2015 )
# ( www.openbsd.org )
####################################

PFLOG=/var/log/pflog
FILE=/var/log/pflog5min.$(date "+%Y%m%d%H%M")
pkill -ALRM -u root -U root -t - -x pflogd
if [ -r $PFLOG ] && [ $(stat -f %z $PFLOG) -gt 24 ]; then
mv $PFLOG $FILE
pkill -HUP -u root -U root -t - -x pflogd
tcpdump -n -e -s 160 -ttt -r $FILE | logger -t pf -p local0.info
```

```
rm $FILE
fi

[root@hermes-/etc]#
```

- Agregué al *crontab* del usuario root una línea que indicaría el tiempo (cada 5 minutos) de ejecución del script, creado anteriormente, para actualizar el archivo de bitácora de los paquetes filtrados.

```
[root@hermes-/etc]# crontab -u root -e
….............................
….............................
####################################
# gaara
# ( 25 - Abril - 2015 )
# ( www.openbsd.org )
####################################
# Rotar archivo pflog cada 5 minutos
0-59/5 * * * * /bin/sh /etc/pflogrotate
[root@hermes-/etc]#
```

- Las siguientes líneas (en recuadro) las agregué al archivo */etc/syslog.conf* para que el sistema tuviera conocimiento del nuevo archivo donde mandaría los registros.

```
[root@hermes-/etc]# cp syslog.conf syslog.conf_25-Abril-2015
[root@hermes-/etc]# vim syslog.conf
.............................
.............................
####################################
# gaara
# ( 25 - Abril - 2015 )
# ( www.openbsd.org )
####################################
#Mando los registros del cortafuegos en formato ASCII
local0.info /sistema/pf/log/pflog.txt
.............................
.............................
[root@hermes-/etc]#
```

- Por último:
    - Creé el archivo de bitácora.
    - Reinicié el demonio *syslogd*.
    - Desactivé la bitácora que el sistema activó en automático; activé el registro de paquetes para el nuevo archivo de bitácora modificando el archivo de sistema */etc/newsyslog.conf*.

```
[root@hermes-/sistema/pf/log]# pwd
/sistema/pf/log
[root@hermes-/sistema/pf/log]# touch pflog.txt
[root@hermes-/sistema/pf/log]# chmod 600 pflog.txt
[root@hermes-/sistema/pf/log]# file /sistema/pf/log/pflog.txt
/sistema/pf/log/pflog.txt: empty
[root@hermes-/sistema/pf/log]#


[root@hermes-/etc]# kill -HUP $(cat /var/run/syslog.pid)


[root@hermes-/etc]# cp newsyslog.conf newsyslog.conf_25-Abril-2015
[root@hermes-/etc]# vim newsyslog.conf
….............................
….............................
####################################
# gaara
# ( 25 - Abril - 2015 )
# ( www.openbsd.org )
####################################
# Rotar el archivo pflog.txt 24 veces; de esta manera tendremos el
registro de 24 meses
# $M1D0 Indica rotacion el primer dia de cada mes a media noche
/sistema/pf/log/pflog.txt 600 24 5242880 $M1D0 Z
[root@hermes-/etc]#
```

    - Es recomendable reiniciar el firewall para descartar cualquier fallo en configuración; en este caso tras configurar los archivos de bitácora reinicié el equipo.

3.3 Net-SNMP (Instalación y Configuración)

3.3.1 Protocolo SNMP

Como había mencionado antes, SNMP (Simple Network Management Protocol) es un protocolo usado para monitorear, en su mayoría, equipos de infraestructura de red; este protocolo cuenta con 3 versiones tanto para IPv4 e IPv6.

Hice uso de este protocolo para poder monitorear los recursos del firewall:

- Uso de memoria ram
- Uso de procesador
- Espacio en disco duro
- Uso de ancho de banda

Con el software Cacti grafiqué los recursos mencionados e hice uso de las Traps que maneja el protocolo para su envío de alarmas ante cualquier anomalía en el comportamiento del Sistema Operativo.

3.3.2 Instalación y Configuración

Instalé el paquete desde repositorios

```
[root@hermes-~]# pkg_add net-snmp
quirks-2.9 signed on 2014-07-31T22:37:55Z
net-snmp-5.7.2.1p2: ok
The following new rcscripts were installed: /etc/rc.d/netsnmpd
/etc/rc.d/netsnmptrapd
[root@hermes-~]#
```

En un archivo nuevo escribí solo las líneas que me interesaban existieran en el archivo de configuración de SNMP.

```
[root@hermes-/sistema]# pwd
/sistema
[root@hermes-/sistema]# mkdir net-snmp
[root@hermes-/sistema]# cd net-snmp/
[root@hermes-/sistema/net-snmp]# cp /etc/snmp/snmpd.conf .


[root@hermes-/sistema/net-snmp]# vim snmpd.conf
#=====================================
# gaara - Configuracion SNMP
#=====================================
# Fecha de Creacion 23/Febrero/2015

#----------------------------
# System Information Setup
#----------------------------
#sysdescr Sun SNMP Agent, Hermes
syscontact jlopez@galois.dgae.unam.mx
syslocation Direccion General de Administracion Escolar - Rectoria

#----------------------------
# SNMPv3 Authentication
#----------------------------
createUser agenteDGAE SHA ******** DES ********

#----------------------------
# Access Control Setup
#----------------------------
rocommunity comunidad 132.247.***.**7
rocommunity comunidad 127.0.0.1
rouser agenteDGAE

#----------------------------
# Trap Destinations
#----------------------------
trapcommunity comunidad
trap2sink 132.247.***.**7
informsink 132.247.***.**7
#authtrapenable 1

#----------------------------
# Monitor Various Aspects of
# the Running Host
```

```
#---------------------------

# Central processing unit ( CPU )
proc CPU

# Hard Disk Drive ( HDD ) -> disk PATH [ minspace ]
disk / 102400kB
disk /tmp 102400kB
disk /usr 102400kB
disk /var 102400kB
disk /opt 102400kB
disk /sistema 102400kB
disk /home 102400kB

# Random Access Memory ( RAM )
Load

#---------------------------
# DisMan Event MIB
#---------------------------
iquerySecName agenteDGAE


#============================
# MIB objects to monitor
#============================
# By default, the expression will be evaluated every 600s (10 minutes)
# -r frequency ( seconds )

# UCD-SNMP-MIB::dskErrorFlag ( .1.3.6.1.4.1.2021.9.1.100 )
# UCD-SNMP-MIB::dskPath ( .1.3.6.1.4.1.2021.9.1.2 )
# UCD-SNMP-MIB::dskDevice ( .1.3.6.1.4.1.2021.9.1.3 )

monitor -o .1.3.6.1.4.1.2021.9.1.2 -o .1.3.6.1.4.1.2021.9.1.3 "Espacio
Minimo Disco-Hermes" .1.3.6.1.4.1.2021.9.1.100 != 0

# UCD-SNMP-MIB::ssCpuIdle ( .1.3.6.1.4.1.2021.11.11 ) -> ( deprecated )
# UCD-SNMP-MIB::ssCpuUser ( .1.3.6.1.4.1.2021.11.9 ) -> ( deprecated )
# UCD-SNMP-MIB::ssCpuSystem ( .1.3.6.1.4.1.2021.11.10 ) -> ( deprecated
)

monitor -r 300 -o .1.3.6.1.4.1.2021.11.9 -o .1.3.6.1.4.1.2021.11.10
"Aumento Carga CPU-Hermes" .1.3.6.1.4.1.2021.11.11 < 80

[root@hermes-/sistema/net-snmp]#
```

Levanté el servicio manualmente y al no arrojar error alguno agregué el equipo a Cacti.

```
[root@hermes-/sistema/net-snmp]# /usr/local/sbin/snmpd -v
NET-SNMP version: 5.7.2.1
Web: http://www.net-snmp.org/
Email: net-snmp-coders@lists.sourceforge.net
[root@hermes-/sistema/net-snmp]# /usr/local/sbin/snmpd -c /sistema/net-
snmp/snmpd.conf
[root@hermes-/sistema/net-snmp]#

[root@hermes-/sistema/net-snmp]# ps -aux | grep net-snmp
root 25024 0.0 0.0 4168 4676 ?? S 4:51AM 0:00.92 /usr/local/sbin/snmpd
-c /sistema/net-snmp/snmpd.conf
root 11449 0.0 0.0 364 288 p0 R+ 4:51AM 0:00.00 grep net-snmp
[root@hermes-/sistema/net-snmp]#
```

Para iniciar el servicio automáticamente ante cualquier reinicio del equipo bastó con crear un archivo de inicio que OpenBSD reconoce en automático; ello es posible con las siguientes líneas:

```
[root@hermes-/etc]# touch /etc/rc.local
[root@hermes-/etc]# vim rc.local
#!/bin/bash

####################################
# gaara
# ( 28 - Abril - 2015 )
####################################

echo -n ' Iniciando Net-SNMP '

if [ -x /usr/local/sbin/snmpd ]; then
echo -n ' net-snmp ' ; /usr/local/sbin/snmpd -c /sistema/net-
snmp/snmpd.conf
fi

[root@hermes-/etc]#
```

- De esta forma aseguramos que el servicio siempre estará disponible.

3.4 Secure SHell ‘SSH’ (Configuración)

3.4.1 Secure SHell

Secure SHell (*ssh*) es un protocolo para conexiones remotas seguras. Este protocolo se encarga de cifrar todo el tráfico de una conexión cliente-servidor.

Sin embargo, existen parámetros para asegurar la configuración de *ssh*; parámetros cómo: número de intentos permitidos para introducir las credenciales, tiempo de espera para cierre de sesión por inactividad, login de usuarios específicos, puerto de escucha para aceptar conexiones al protocolo *ssh*, etc.

3.4.2 Configuración

Definí un banner de saludo para usuarios que intenten realizar una conexión *ssh* con el firewall:

```
[root@hermes-/etc]# vim banner_ssh

Estas Autorizado ? ...

[root@hermes-/etc]#
```

Cuando las credenciales introducidas son las correctas, muestro un banner de saludo:

```
[root@hermes-/etc]# cp motd motd_28-Abril-2015
[root@hermes-/etc]# vim motd
OpenBSD 5.6 (GENERIC.MP) #333: Fri Aug 8 00:20:21 MDT 2014
```

```
Bienvenido a Hermes ...!
[root@hermes-/etc]#
```

El archivo de configuración que guardé como principal para el protocolo *ssh* tiene como objetivo principal las siguientes directivas:

- Dirección IP y puerto de escucha
- Versión del protocolo a utilizar
- PID del servicio
- Modo *verbose*
- Tiempo de espera para las credenciales
- Máximo número de intentos
- Negar conexión directa al usuario root
- Permitir conexiones a usuarios de direcciones IP específicas

```
[root@hermes-/etc/ssh]# cp sshd_config sshd_config_28-Abril-2015

[root@hermes-/etc/ssh]# vim sshd_config
#====================================
# gaara
# ( 28 - Abril - 2015 )
# ( Firewall hermes )
#====================================

########################
# Global
########################
# Indicamos el Puerto e IP de escucha
Port 22
ListenAddress 132.248.xxx.xx1

# Indicamos la Version de SSH
Protocol 2

# Solo dejaremos que utilize el par de claves RSA
HostKey /etc/ssh/ssh_host_rsa_key

# Habilitamos esta directiva para prevenir "Escalacion de Privilegios"
UsePrivilegeSeparation yes

# Indicamos el archivo donde alojara el PID del demonio SSH
```

```
PidFile /var/run/sshd.pid


#########################
# Logging
#########################
# Indicamos el valor para Loguear Mensajes de SSH
SyslogFacility AUTH

# Directiva para Debuggear
LogLevel INFO

#########################
# Authentication
#########################
# Tiempo para desconectar del servidor si no hay exito en introducir
las credenciales
LoginGraceTime 30


# Indicamos el numero de intentos para introducir las credenciales (
evita ataques de fuerza bruta )
MaxAuthTries 3

# No permitimos la posibilidad de logearse como usuario 'root'
directamente atravez de SSH
PermitRootLogin no

# Hacemos que sshd verifique modos de archivo y propietarios de los
archivos y del home del usuario antes de iniciar sesion
StrictModes yes

# Indicamos solo autenticacion RSA
RSAAuthentication yes

# Permitimos la autenticacion por llave publica
PubkeyAuthentication yes

# Especificamos el archivo que contiene las llaves publicas para
autenticacion ( %h --> home del usuario a loguearse )
#AuthorizedKeysFile %h/.ssh/authorized_keys # --> No necesario, debemos
solicitar introducir las credenciales

# Ignoramos los archivos ~/.rhosts and ~/.shosts de los usuarios
IgnoreRhosts yes

# No permitimos la autenticacion rhosts
HostbasedAuthentication no

# Desabilitamos cadenas de passwords vacias
```

```
PermitEmptyPasswords no

# Habilitamos la autenticacion por Password
PasswordAuthentication yes


# Desabilitamos "X11 forwarding"
X11Forwarding no

# Especificamos el Primer Numero del Display disponible para "X11
forwarding" de SSH

#X11DisplayOffset 15 # --> No necesaria, ya que esta desabilitado "X11
forwarding"

# Mostramos un banner despues de que el usuario se logueo con exito
PrintMotd yes

# Mostramos la ultima vez en que el usuario se logueo en el servidor
PrintLastLog yes

# Enviamos mensajes al otro lado de la conexion para no tener
conexiones colgadas indefinidamente en el servidor
TCPKeepAlive yes

# Mostramos un Banner antes de que el usuario introduzca sus
credenciales
Banner /etc/banner_ssh


# Permitimos el uso de sftp
#Subsystem sftp /usr/libexec/sftp-server # --> No necesario

# Indicamos la Lista de Usuarios e IP's permitidos a loguearse en el
Servidor
AllowUsers user@132.248.xxx.*

[root@hermes-/etc/ssh]#
```

Lo último fue reiniciar el demonio *ssh* para que tomara los cambios realizados:

```
[root@hermes-/etc]# cat /var/run/sshd.pid
29823
[root@hermes-/etc]# kill -HUP 29823
[root@hermes-/etc]# cat /var/run/sshd.pid
16077
[root@hermes-/etc]#
```

3.5 Virtual Private Network (VPN)

3.5.1 OpenVPN

Una red privada virtual (Virtual Private Network, VPN) consiste en una tecnología que nos permite estar conectados directamente a la red LAN de nuestra empresa de forma segura a través de internet. Esta tecnología nos permite extender virtualmente una LAN a un usuario que físicamente no está conectado a esta LAN, pero sí conectado virtualmente; hace que el equipo del usuario sea un equipo más en LAN.

Por su parte, OpenVPN es un software que permite implementar la tecnología VPN. Este software se encuentra bajo licencia GPL lo cual lo hace ideal para su adaptación al proyecto.

3.5.2 Instalación

Llevé a cabo la instalación de los paquetes mediante los repositorios de OpenBSD:

```
[root@hermes-/etc]# pkg_info openvpn
[root@hermes-/etc]# pkg_add openvpn-2.3.2.tgz
quirks-2.9 signed on 2014-07-31T22:37:55Z
openvpn-2.3.2:lzo2-2.08: ok
openvpn-2.3.2: ok
Look in /usr/local/share/doc/pkg-readmes for extra documentation.
[root@hermes-/etc]#


[root@hermes-/etc]# pkg_info easy-rsa
[root@hermes-/etc]# pkg_add easy-rsa-2.2.0p0.tgz
quirks-2.9 signed on 2014-07-31T22:37:55Z
easy-rsa-2.2.0p0: ok
```

```
[root@hermes-/etc]#


[root@hermes-/etc]# pkg_add unzip
[root@hermes-/etc]# pkg_add zip
[root@hermes-/etc]# pkg_add lzo-1.08p3.tgz
```

El paquete de utilerías *easy-rsa* contiene todos los archivos necesarios para la configuración de la VPN en el firewall.

3.5.3 PKI (Public Key Infrastructure)

La infraestructura de clave pública (Public Key Infrastructure, PKI) es un conjunto de procedimientos para la creación de comunicaciones seguras haciendo uso de certificados digitales.

Para la configuración de PKI fue necesario que creara:

- Certificado CA y una llave privada *raíz*.
- Certificado y una llave privada para el servidor OpenVPN.
- Certificado y llave privada por separado para cada cliente que se conecta a la VPN.

Realicé las configuraciones necesarias para los certificados y llaves, estas fueron:

- Crear el directorio principal de trabajo:

```
[root@hermes-/etc]# mkdir openvpn
```

- Crear los directorios de trabajo para la utilería *easy-rsa*

```
[root@hermes-/etc]# cd /usr/local/share/easy-rsa/
[root@hermes-/usr/local/share/easy-rsa]# pwd
```

```
/usr/local/share/easy-rsa
[root@hermes-/usr/local/share/easy-rsa]#

[root@hermes-/usr/local/share/easy-rsa]# mkdir keys_llaves
[root@hermes-/usr/local/share/easy-rsa]# cp openssl-1.0.0.cnf
openssl.cnf
```

- Inicializar el archivo que contendrá los datos de la empresa, de esta forma evitamos que al crearse más certificados tengamos que ingresar nuevamente esta información.

```
[root@hermes-/usr/local/share/easy-rsa]# cp vars vars_29-Abril-2015
[root@hermes-/usr/local/share/easy-rsa]# vim vars
###################################
# gaara
# ( 29 - Abril - 2015 )
# easy-rsa parameter settings
###################################

# Directorio donde se aloja 'easy-rsa'
export EASY_RSA="`pwd`"

# Executables
export OPENSSL="openssl"
export PKCS11TOOL="pkcs11-tool"
export GREP="grep"

# Archivo "openssl.cnf" incluido con 'easy-rsa'
export KEY_CONFIG=`$EASY_RSA/whichopensslcnf $EASY_RSA`

# Directorio donde se crearan las 'Llaves'
export KEY_DIR="$EASY_RSA/keys_llaves"

# Issue rm -rf warning
echo NOTE: If you run ./clean-all, I will be doing a rm -rf on $KEY_DIR
echo NOTA: Si ejecutas ./clean-all, Estare haciendo un rm -rf on
$KEY_DIR

# PKCS11 fixes
export PKCS11_MODULE_PATH="dummy"
export PKCS11_PIN="dummy"

# Tamano de la Llave
export KEY_SIZE=1024
```

```
# Dias para que el Certificado raiz CA expire ( 5 anios )
export CA_EXPIRE=1825

# Dias para que los Certificados expiren ( 2 anios )
export KEY_EXPIRE=730

# Informacion de la Institucion que estara en el Certificado
export KEY_COUNTRY="MX"
export KEY_PROVINCE="Mexico"
export KEY_CITY="Distrito Federal"
export KEY_ORG="DGAE - SysAdmin ( J. Manuel Lopez L. )"
export KEY_EMAIL=seguridad@galois.dgae.unam.mx

[root@hermes-/usr/local/share/easy-rsa]#
```

3.5.4 Autoridad Certificadora y Llave 'raíz'

En esta parte inicialicé la PKI construyendo los parámetros con el algoritmo Diffie-Hellman:

```
[root@hermes-/usr/local/share/easy-rsa]# . ./vars
NOTE: If you run ./clean-all, I will be doing a rm -rf on
/usr/local/share/easy-rsa/keys_llaves
NOTA: Si ejecutas ./clean-all, Estare haciendo un rm -rf on
/usr/local/share/easy-rsa/keys_llaves
[root@hermes-/usr/local/share/easy-rsa]#
[root@hermes-/usr/local/share/easy-rsa]# ./clean-all
[root@hermes-/usr/local/share/easy-rsa]# ./build-dh
Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
[…..]
[root@hermes-/usr/local/share/easy-rsa]#
```

Pasé a crear el certificado CA raíz al igual que la Llave:

```
[root@hermes-/usr/local/share/easy-rsa]# ./pkitool --initca
Using CA Common Name: DGAE - SysAdmin ( J. Manuel Lopez L. ) CA
Generating a 1024 bit RSA private key
.++++++
```

```
.................++++++
writing new private key to 'ca.key'
-----
[root@hermes-/usr/local/share/easy-rsa]#
```

3.5.5 Certificados y Llaves

Lo primero que llevé a cabo fue la creación del certificado y llave propios del servidor para aceptar la conexiones VPN:

```
[root@hermes-/usr/local/share/easy-rsa]# ./pkitool --server hermes-
vpn.dgae.unam.mx
Generating a 1024 bit RSA private key
.......++++++
........++++++
writing new private key to 'hermes-vpn.dgae.unam.mx.key'
-----
Using configuration from /usr/local/share/easy-rsa/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName :PRINTABLE:'MX'
stateOrProvinceName :PRINTABLE:'Mexico'
localityName :PRINTABLE:'Distrito Federal'
organizationName :PRINTABLE:'DGAE - SysAdmin ( J. Manuel Lopez L. )'
commonName :PRINTABLE:'hermes-vpn.dgae.unam.mx'
emailAddress :IA5STRING:'seguridad@galois.dgae.unam.mx'
Certificate is to be certified until Apr 28 22:53:43 2017 GMT (730
days)
Write out database with 1 new entries
Data Base Updated
[root@hermes-/usr/local/share/easy-rsa]#
```

Lo siguiente fue la creación de los certificados y llaves para cada usuario a conectarse al servidor VPN:

```
[Usuario 1]
```

```
[root@hermes-/usr/local/share/easy-rsa]# ./pkitool --pass gaara-
hermes.dgae.unam.mx
Generating a 1024 bit RSA private key
.........++++++
.....................++++++
writing new private key to 'gaara-hermes.dgae.unam.mx.key'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
Using configuration from /usr/local/share/easy-rsa/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName :PRINTABLE:'MX'
stateOrProvinceName :PRINTABLE:'Mexico'
localityName :PRINTABLE:'Distrito Federal'
organizationName :PRINTABLE:'DGAE - SysAdmin ( J. Manuel Lopez L. )'
commonName :PRINTABLE:'gaara-hermes.dgae.unam.mx'
emailAddress :IA5STRING:'seguridad@galois.dgae.unam.mx'
Certificate is to be certified until Apr 28 23:16:45 2017 GMT (730
days)
Write out database with 1 new entries
Data Base Updated
[root@hermes-/usr/local/share/easy-rsa]#


[Usuario 2]
[root@hermes-/usr/local/share/easy-rsa]# ./pkitool --pass fzara-
hermes.dgae.unam.mx
[.......]
[root@hermes-/usr/local/share/easy-rsa]#
```

Los archivos generados para los clientes son los que tienen por extensión:

- gaara-hermes.dgae.unam.mx.crt
- gaara-hermes.dgae.unam.mx.csr
- gaara-hermes.dgae.unam.mx.key

3.5.6 Configuración Servidor OpenVPN

La instalación de OpenVPN creó un directorio */etc/openvpn* que es donde se guardan los archivos de configuración, así como también el usuario y grupo _openvpn; debajo de esta ruta creé el directorio *private* para alojar la llave privada del servidor. A este directorio agregué los permisos adecuados para evitar consultas no autorizadas:

```
[root@hermes-/etc]# mkdir -p openvpn/private
[root@hermes-/etc]# chmod 700 openvpn/private
```

Los archivos necesarios para la configuración tanto del servidor como del cliente son:

- *ca.crt,* (certificado 'CA'), existe en el directorio */etc/openvpn/* para ambos.
- *ca.key,* solo existe en el servidor bajo la ruta */etc/openvpn/private/*.
- *dh1024.pem*, solo existe en el servidor bajo la ruta */etc/openvpn/.*
- **.crt*, **.key* (certificados y llaves privadas), existen en los equipos clientes cada uno con sus respectivos archivos.
    - Las llaves privadas deben alojarse bajo la ruta */etc/openvpn/private/*.
    - Los certificados deben alojarse bajo la ruta */etc/openvpn/*.

En el servidor hice las copias de los archivos a las rutas indicadas:

```
[root@hermes-/usr/local/share/easy-rsa/keys_llaves]# pwd
/usr/local/share/easy-rsa/keys_llaves
[root@hermes-/usr/local/share/easy-rsa/keys_llaves]# cp ca.crt
/etc/openvpn/
[root@hermes-/usr/local/share/easy-rsa/keys_llaves]# cp dh1024.pem
/etc/openvpn/
[root@hermes-/usr/local/share/easy-rsa/keys_llaves]# cp hermes-
vpn.dgae.unam.mx.key /etc/openvpn/private/
[root@hermes-/usr/local/share/easy-rsa/keys_llaves]# cp hermes-
vpn.dgae.unam.mx.crt /etc/openvpn/
```

Copié el archivo principal de configuración (de un archivo ejemplo) OpenVPN para su edición:

```
[root@hermes-/usr/local/share/examples/openvpn/sample-config-files]#
pwd
/usr/local/share/examples/openvpn/sample-config-files
[root@hermes-/usr/local/share/examples/openvpn/sample-config-files]# cp
server.conf /etc/openvpn/
[root@hermes-/usr/local/share/examples/openvpn/sample-config-files]# cd
/etc/openvpn/
[root@hermes-/etc/openvpn]# ls -l server.conf
-r--r--r-- 1 root wheel 10290 May 4 17:10 server.conf
[root@hermes-/etc/openvpn]#
```

Creé directorios y archivos para:

- Alojar los archivos de configuración de cada cliente de la VPN.
- Alojar bitácoras de la VPN.
- Guardar las direcciones IP que la VPN asocia a cada cliente cuando este se conecta.
- Registrar una corta salida del estatus de las conexiones VPN; este archivo se sobre escribirá cada minuto.
- Guardar la bitácora de OpenVPN

```
[root@hermes-/etc/openvpn]# mkdir ccd

[root@hermes-/sistema]# mkdir -p openvpn/logs

[root@hermes-/sistema/openvpn/logs]# pwd
/sistema/openvpn/logs
[root@hermes-/sistema/openvpn/logs]# touch ipp.txt

[root@hermes-/sistema/openvpn/logs]# touch openvpn-status.log

[root@hermes-/sistema/openvpn/logs]# touch openvpn.log
```

La configuración del archivo principal de OpenVPN que realicé es:

```
[root@hermes-/etc/openvpn]# vim server.conf
####################################
# gaara
# ( 4 - Mayo - 2015 )
####################################

################################################
# OpenVPN 2.0 config file for #
# multi-client server. #
# #
# This file is for the server side #
# of a many-clients <-> one-server #
# OpenVPN configuration. #
# #
# OpenVPN also supports #
# single-machine <-> single-machine #
# configurations (See the Examples page #
# on the web site for more info). #
# #
# This config should work on Windows #
# or Linux/BSD systems. Remember on #
# Windows to quote pathnames and use #
# double backslashes, e.g.: #
# "C:\\Program Files\\OpenVPN\\config\\foo.key" #
# #
# Comments are preceded with '#' or ';' #
################################################

# Which local IP address should OpenVPN listen on? (optional)
local 132.248.xxx.xx1

# Which TCP/UDP port should OpenVPN listen on?
port 51194

# TCP or UDP server?
proto udp

# "dev tun" will create a routed IP tunnel,
# "dev tap" will create an ethernet tunnel.
dev tun0

# SSL/TLS root certificate (ca), certificate
# (cert), and private key (key). Each client
# and the server must have their own cert and
# key file. The server and all clients will
```

```
# use the same ca file.
ca /etc/openvpn/ca.crt
cert /etc/openvpn/hermes-vpn.dgae.unam.mx.crt
key /etc/openvpn/private/hermes-vpn.dgae.unam.mx.key # This file should
be kept secret

# Diffie hellman parameters.
dh /etc/openvpn/dh1024.pem

# Configure server mode and supply a VPN subnet for OpenVPN to draw
client addresses from.
# The server will take a.b.c.1 for itself, the rest will be made
available to clients.
# Address range for the "tun(4)" interfaces
server 10.xx9.xxx.0 255.255.255.0

# Maintain a record of client <-> virtual IP address associations in
this file.
ifconfig-pool-persist /sistema/openvpn/logs/ipp.txt

# Push routes to the client to allow it to reach other private subnets
behind the server.
# Add a route to the local network to the client's routing table
# push "route 132.248.xxx.0 255.255.255.0" -> El Ruteo se hara manual

# Add routes to the remote networks to the server's routing table (
Segmento 10.8.8.0 Asignado Manualmente )
# route 192.168.0.0 255.255.255.0
# route 192.168.1.0 255.255.255.0
# route 10.0.2.0 255.255.255.0
route 10.x10.xxx.0 255.255.255.0

# To assign specific IP addresses to specific clients or if a
connecting client has a private
# subnet behind it that should also have VPN access, use the
subdirectory "ccd" for client-specific
# configuration files (see man page for more info).
client-config-dir /etc/openvpn/ccd

# The keepalive directive causes ping-like messages to be sent back and
forth over
# the link so that each side knows when the other side has gone down.
# Ping every 10 seconds, assume that remote peer is down if no ping
received during a 120 second time period.
keepalive 40 240

# Enable compression on the VPN link.
comp-lzo

# The maximum number of concurrently connected clients we want to
```

```
allow.
max-clients 3

# It's a good idea to reduce the OpenVPN daemon's privileges after
initialization.
user _openvpn
group _openvpn

# The persist options will try to avoid accessing certain resources on
restart that may no longer be accessible because
# of the privilege downgrade.
persist-key
persist-tun

# Output a short status file showing current connections, truncated and
rewritten every minute.
status /sistema/openvpn/logs/openvpn-status.log

# Use log or log-append to override this default.
log-append /sistema/openvpn/logs/openvpn.log

# Set the appropriate level of log file verbosity.
# 0 is silent, except for fatal errors
# 4 is reasonable for general usage
# 5 and 6 can help to debug connection problems
# 9 is extremely verbose
verb 9

[root@hermes-/etc/openvpn]#
```

Para cada cliente de la VPN creé un archivo de configuración, este archivo contiene información como la dirección IP que se le asignará una vez la conexión sea exitosa. El segmento de direcciones IP privadas que se maneja es 10.x10.xxx.0/24.

Para la asignación de direcciones IP tomé en cuenta lo siguiente:

- OpenVPN acepta conexiones de sistemas operativos GNU/Linux, Mac OS X y Windows.
- Las direcciones IP se asignan en pares, esto es, la dirección IP del cliente y la dirección IP gateway del enlace.
- Para S.O. Windows, la dirección IP es asignada de una de las 64 subredes creadas a partir del segmento /24 definido en la configuración del servidor:

- Las subredes las obtuvé ejecutando el comando *openvpn.exe --show-valid-subnets* en un equipo Windows con OpenVPN previamente instalado.

```
[ 1, 2] [ 5, 6] [ 9, 10] [ 13, 14] [ 17, 18]
[ 21, 22] [ 25, 26] [ 29, 30] [ 33, 34] [ 37, 38]
[ 41, 42] [ 45, 46] [ 49, 50] [ 53, 54] [ 57, 58]
[ 61, 62] [ 65, 66] [ 69, 70] [ 73, 74] [ 77, 78]
[ 81, 82] [ 85, 86] [ 89, 90] [ 93, 94] [ 97, 98]
[101,102] [105,106] [109,110] [113,114] [117,118]
[121,122] [125,126] [129,130] [133,134] [137,138]
[141,142] [145,146] [149,150] [153,154] [157,158]
[161,162] [165,166] [169,170] [173,174] [177,178]
[181,182] [185,186] [189,190] [193,194] [197,198]
[201,202] [205,206] [209,210] [213,214] [217,218]
[221,222] [225,226] [229,230] [233,234] [237,238]
[241,242] [245,246] [249,250] [253,254]
```

- Estas subredes pueden usarse para construir los Túneles VPN en Windows.
- La asignación de direcciones IP será del mismo rango para clientes con sistema operativo GNU/Linux y Mac OS X.
- Solo estos pares de direcciones IP no serán asignados a los clientes:

```
[ 1, 2] [ 5, 6] [ 9, 10]
[245,246] [249,250] [253,254]
```

- De acuerdo a nuestro archivo de configuración *server.conf* el segmento de red sobre el que se asignan direcciones IP es:
  - 10.x10.xxx.0/24
  - Contamos con 58 subredes de direcciones IP para asignar.
- Si el número de clientes llegase a exceder las subredes, solo será necesario declarar un nuevo segmento de red en el archivo de configuración *server.conf*.
  - Como ejemplo puede ser: 10.x20.xxx.0/24
  - Este segmento de red nos permite tener 58 subredes mas.

Ahora bien, los archivos de configuración de cada cliente los nombré de acuerdo con el nombre de los archivos creados *Common Name* con OpenVPN.

```
[root@hermes-/etc/openvpn/ccd]# vim gaara-hermes.dgae.unam.mx
ifconfig-push 10.x10.xxx.13 10.x10.xxx.14
[root@hermes-/etc/openvpn/ccd]#

[root@hermes-/etc/openvpn/ccd]# vim sjimenez-hermes.dgae.unam.mx
ifconfig-push 10.x10.xxx.17 10.x10.xxx.18
[root@hermes-/etc/openvpn/ccd]#

[root@hermes-/etc/openvpn/ccd]# vim fzara-hermes.dgae.unam.mx
ifconfig-push 10.x10.xxx.21 10.x10.xxx.22
[root@hermes-/etc/openvpn/ccd]#
```

Para no perder el control de subredes utilizadas, hago uso del archivo *Ips_OpenVPN*:

```
[root@hermes-/etc/openvpn]# vim Ips_OpenVPN
# gaara
# 22 - Mayo - 2015
# Segmento 10.x10.xxx.0/24
[ 1, 2] --> No utilizar
[ 5, 6] --> No utilizar
[ 9, 10] --> No utilizar
[ 13, 14] --> Jose Manuel Lopez Lopez ( gaara )
[ 17, 18] --> Sergio Jimenez Tovar
[ 21, 22] --> Fernando Zaragoza Hernandez
[ 25, 26]
[ 29, 30]
[ ….... ]
[237,238]
[241,242]
[245,246] --> No utilizar
[249,250] --> No utilizar
[253,254] --> No utilizar
[root@hermes-/etc/openvpn]#
```

Agregué en el archivo *rc.local* líneas de código necesarias para iniciar OpenVPN junto con el sistema operativo:

```
[root@hermes-/etc]# vim rc.local
….............................
….............................
echo -n ' Iniciando OpenVPN '
if [ -x /usr/local/sbin/openvpn ]; then
echo -n ' OpenVPN ' ; /usr/local/sbin/openvpn --config
/etc/openvpn/server.conf &
fi
[root@hermes-/etc]#
```

Por último, levanté el servicio de VPN manualmente para corroborar que no arrojara error alguno:

```
[root@hermes-/etc]# /usr/local/sbin/openvpn --config
/etc/openvpn/server.conf &
[1] 6429
[root@hermes-/etc]# ps -aux | grep openvpn
_openvpn 6429 0.0 0.0 1284 3664 p0 S 6:24PM 0:32.60
/usr/local/sbin/openvpn --config /etc/openvpn/server.conf
root 16795 0.0 0.0 232 192 p0 R+ 6:24PM 0:00.00 grep openvpn
[root@hermes-/etc]#
```

Ya que había verificado que el servicio no marcaba error reinicié el firewall, corroborando de esta manera que la VPN inicia automáticamente junto con todos los servicios del sistema operativo.

3.5.7 Configuración del cliente OpenVPN (Linux)

Las pruebas de conexión las llevé a cabo con un equipo con GNU/Linux instalado previamente; los datos necesarios fueron:

- Kali Linux (máquina virtual - 10.0.2.15)
- Red interna (10.0.2.0/24, gw 10.0.2.2)
- Red externa (NAT - 132.247.xxx.58)

Instalé OpenVPN en el equipo cliente con la herramienta *apt-get install paquete* la cual instaló la misma versión que la del servidor (ver *Figura 3.4*):

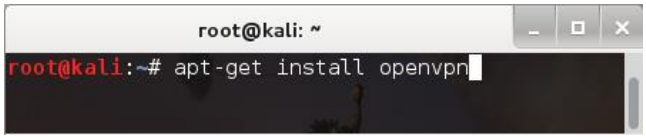

Figura 3.4 - Instalación de OpenVPN en GNU/Linux.

El cliente al instalar OpenVPN creó un directorio donde alojar los archivos de configuración; los archivos de configuración los alojé en el directorio mencionado, estos archivos corresponden a:

- */etc/openvpn/ca.crt*
- */etc/openvpn/gaara-hermes.dgae.unam.mx.crt*
- */etc/openvpn/private/gaara-hermes.dgae.unam.mx.key*

Con los archivos copiados al equipo cliente, edité el archivo de configuración *client.conf* de donde se tomarán los parámetros necesarios para realizar la conexión:

```
[root@hermes-/etc/openvpn]# cp
/usr/local/share/examples/openvpn/sample-config-files/client.conf
/etc/openvpn/
[root@hermes-/etc/openvpn]# vim client.conf
#####################################
```

```
# gaara
# ( 4 - Mayo - 2015 )
####################################

################################################
# #
# Sample client-side OpenVPN 2.0 config file #
# for connecting to multi-client server. #
# #
################################################

# Specify that we are a client and that we will be pulling certain
config file directives from the server.
client

# Use the same setting as you are using on the server.
dev tun0

# UDP server. Use the same setting as on the server.
proto udp

# The hostname/IP and port of the server.
Remote 132.248.xxx.xx1 51194

# Most clients don't need to bind to a specific local port number.
nobind

# Downgrade privileges after initialization (non-Windows only)
user openvpn
group openvpn

# Try to preserve some state across restarts.
persist-key
persist-tun

# SSL/TLS parms.
ca /etc/openvpn/ca.crt
cert /etc/openvpn/gaara-hermes.dgae.unam.mx.crt
key /etc/openvpn/private/gaara-hermes.dgae.unam.mx.key

# Verify server certificate by checking that the certicate has the
nsCertType field set to "server".

# To use this feature, you will need to generate your server
certificates with the nsCertType field set to "server".

# The build-key-server script in the easy-rsa folder will do this.
ns-cert-type server

# Enable compression on the VPN link.
```

```
comp-lzo

# Set log file verbosity.
verb 9
[root@hermes-/etc/openvpn]#
```

Dentro del archivo de configuración es importante que las rutas hacia los archivos de conexión coincidan en donde realmente están alojados (ver *Figura 3.5*); otro directorio que creé fue *scripts* el cual aloja un script (add_routes.sh) escrito en bash para facilitar el enrutamiento de direcciones IP una vez la conexión a la VPN haya sido exitosa.

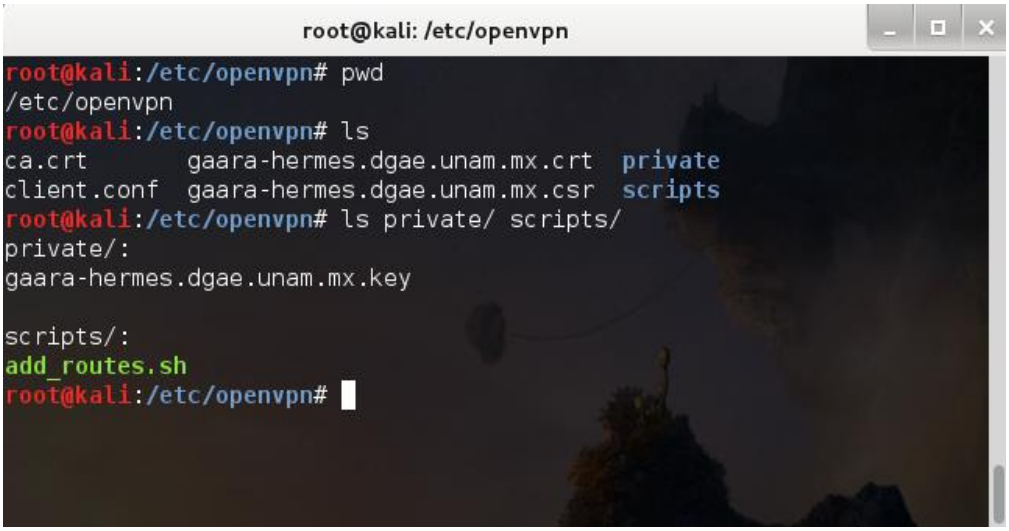

Figura 3.5 - Directorios y archivos para la configuración del cliente OpenVPN.

OpenVPN creó un grupo y usuario *openvpn* (ver *Figura 3.6*), los cuales utilizará para levantar el servicio; estos parámetros pueden cambiar si se desea utilizar otro grupo o usuario del sistema para levantar el servicio.

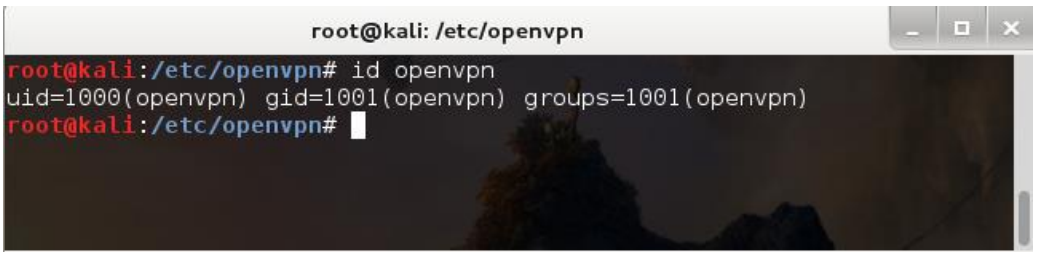

Figura 3.6 - Grupo y usuario con los que inicia el servicio OpenVPN

Del lado del cliente configuré el parámetro *Ip Forwarding* para la correcta creación de los túneles, esto fue:

- Para sistemas operativos base Debian:
    - En el archivo */etc/sysctl.conf* des-comenté la línea:
        - *net.ipv4.ip_forward=1*
    - Para actualizar los cambios ejecuté el comando:
        - *sysctl -p /etc/sysctl.conf*
    - El haber hecho el cambio en el archivo directamente hace que la bandera se mantenga en 1, sin importar si el cliente reinicia o apaga el equipo.

Para la conexión del cliente a la VPN solo fue necesario ejecutar (como root) el comando *openvpn* e indicando el archivo de configuración (ver *Figura 3.7*).

Figura 3.7 - Realizando conexión a la VPN desde un cliente con GNU/Linux.

Una vez realizada la petición al servidor VPN, en pantalla me solicitaba la constraseña del certificado utilizado para la conexión (ver *Figura 3.8*); en este caso el usuario con el que me estaba conectando correspondía a gaara-hermes.dgae.unam.mx.

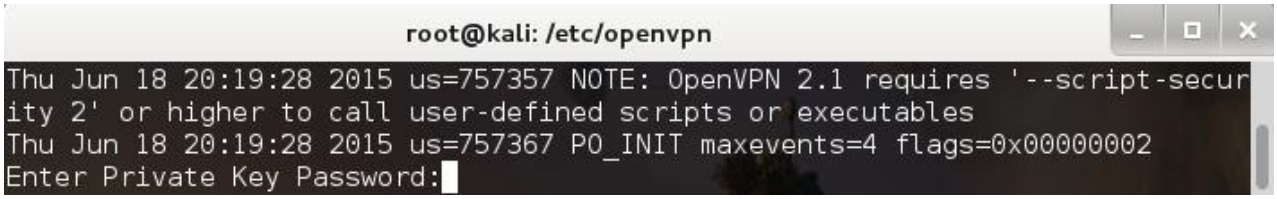

Figura 3.8 - Solicitud de credenciales para el certificado de conexión a la VPN.

En la siguiente imagen *Figura 3.9*, muestro la salida de pantalla que obtuvé al ejecutar el comando *ifconfig*; en ella vemos la interface virtual *tun0* creada por OpenVPN y la dirección IP que el servidor le asignó para así poder comunicarse.

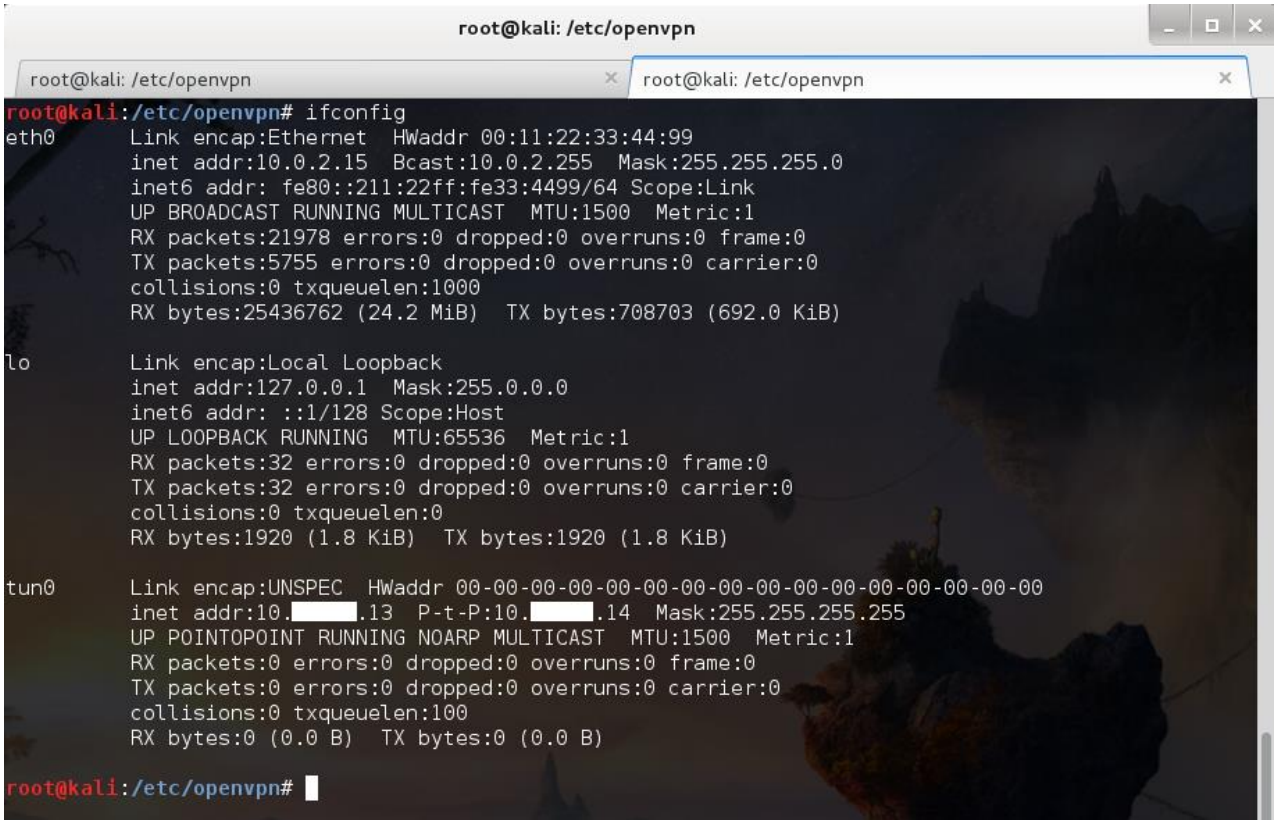

Figura 3.9 - Salida en pantalla del comando ifconfig; conexión exitosa a la VPN.

Ya establecida la conexión ejecuté el script add_routes.sh como usuario root (ver *Figuras 3.10 y 3.11*); lo que hace este script es:

- Agregar direcciones IP públicas de la red interna a nuestra tabla de ruteo.
- Estas direcciones IP al ser públicas, le estaré indicando al equipo cliente que las va a alcanzar vía interface *tun0*; todo lo demás por la interface por defecto, en este caso fue *eth0*.
- Eliminar direcciones IP públicas de la red interna agregadas a la tabla de ruteo.
- El código del script se puede visualizar en el **Anexo A. Script Linux**.

Dentro de las opciones que nos permite elegir el script:

- Agregar direcciones IP a la Tabla de Ruteo
    - Rectoría
        - Linux

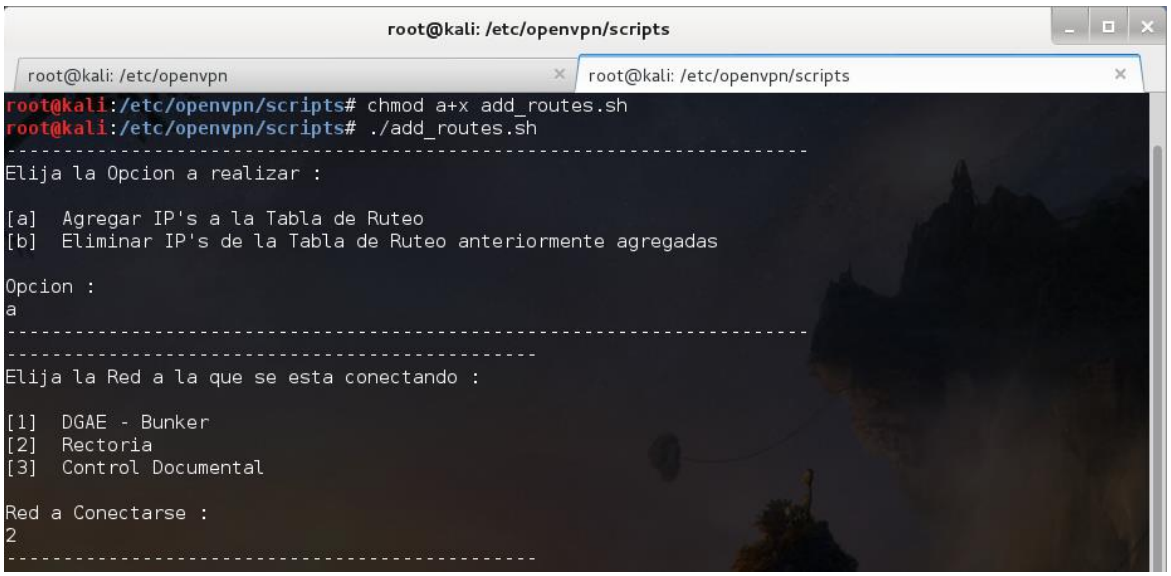

Figura 3.10 - Opciones del menú mostradas por el script add_routes.sh primera parte.

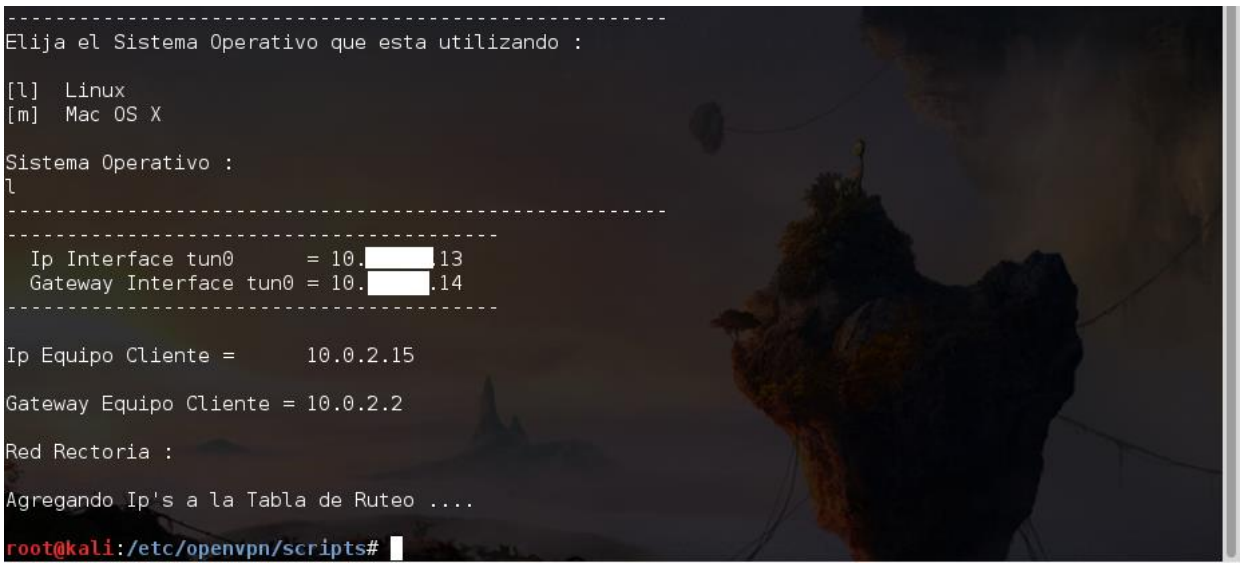

Figura 3.11 - Opciones del menú, selección del sistema operativo del cliente.

Como vemos en las imágenes anteriores, se agregaron las direcciones IP de Rectoría a nuestra tabla de ruteo.

Para detener la conexión con el servidor VPN, en la misma ventana en la que levanté el servicio usé la combinación de teclas *Ctrl - C* para detener el proceso.

- La conexión inicial la pude haber mandado a segundo plano terminando la sentencia de comandos que realiza la conexión con el carácter *&*.
    - Si este hubiese sido el caso bastaba con buscar el *PID* y pasarle el dato al comando *Kill -15 PID*.

Para eliminar las direcciones IP agregadas a nuestra tabla de ruteo ejecuté nuevamente el script add_routes.sh seleccionando las opciones adecuadas (ver *Figura 3.12*).

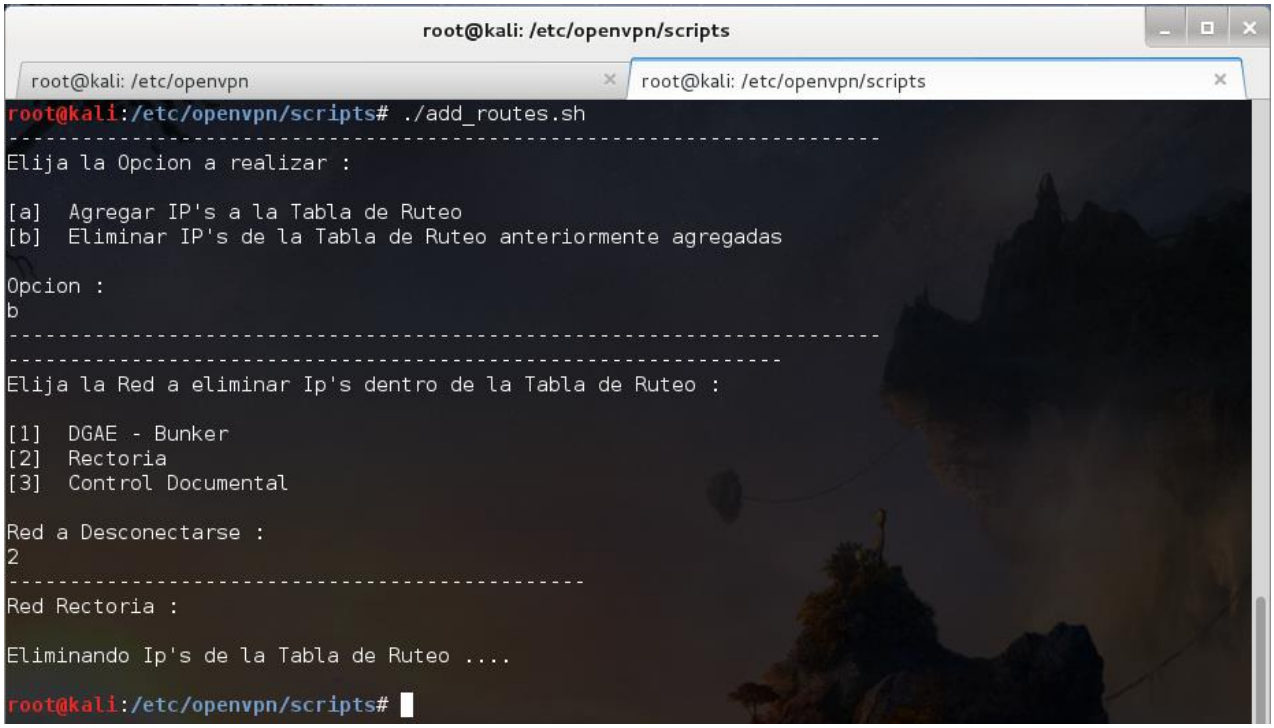

Figura 3.12 - Salida en pantalla de las opciones para eliminar direcciones IP de la tabla de ruteo.

3.5.8 Configuración del cliente OpenVPN (Windows)

La página oficial para la descarga del cliente OpenVPN de Windows es:

https://openvpn.net/index.php/open-source/downloads.html

Para la instalación del cliente, solo ejecuté el archivo descargado siguiendo los pasos recomendados (ver *Figura 3.13*).

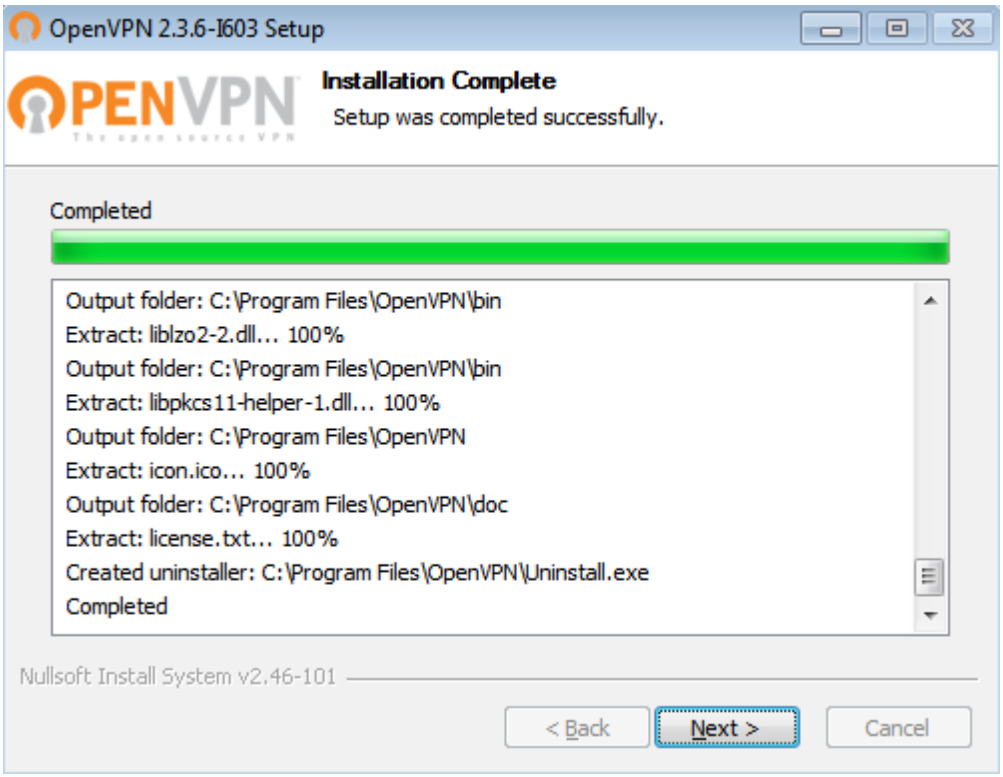

Figura 3.13 - Instalación del cliente OpenVPN en Windows.

La instalación de OpenVPN creó un directorio donde lee los archivos para realizar la conexión con la VPN.

Directorio: *C:\Programs Files\OpenVPN\config*.

Los archivos que necesita OpenVPN debajo del directorio mencionado son (ver *Figura* 3.14):

- client.ovpn - archivo de configuración
- ca.crt
- gaara-hermes.dgae.unam.mx.crt
- gaara-hemres.dgae.unam.mx.key

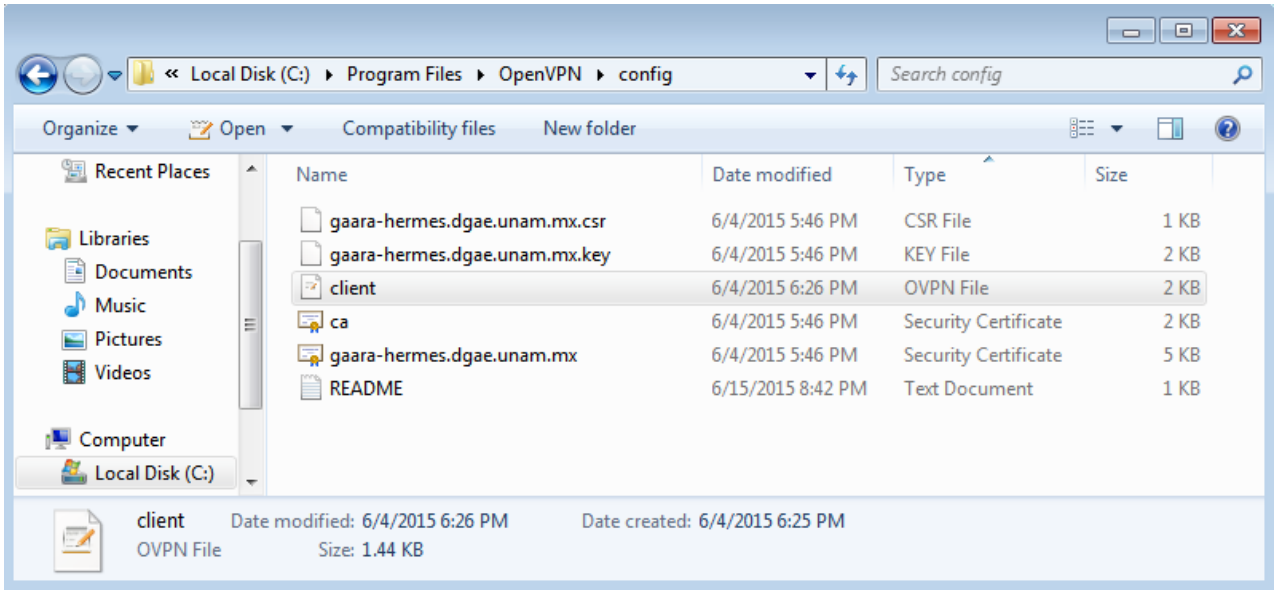

Figura 3.14 - Archivos de configuración del cliente OpenVPN en Windows.

Mi configuración del archivo *client.ovpn* está basada en la misma que la del cliente Linux, sin embargo, difiere en cómo encuentra los archivos para la conexión:

```
#####################################
# gaara
# ( 25 - Mayo - 2015 )
#####################################

#############################################
# #
# Sample client-side OpenVPN 2.0 config file #
# for connecting to multi-client server. #
# #
```

```
#############################################
# Specify that we are a client and that we will be pulling certain
config file directives from the server.
Client

# Use the same setting as you are using on the server.
dev tun0

# UDP server. Use the same setting as on the server.
proto udp

# The hostname/IP and port of the server.
Remote 132.248.xxx.xx1 51194

# Most clients don't need to bind to a specific local port number.
nobind

# Downgrade privileges after initialization (non-Windows only)
user openvpn
group openvpn

# Try to preserve some state across restarts.
persist-key
persist-tun

# SSL/TLS parms.
ca ca.crt
cert gaara-hermes.dgae.unam.mx.crt
key gaara-hermes.dgae.unam.mx.key

# Verify server certificate by checking that the certicate has the
nsCertType field set to "server".

# To use this feature, you will need to generate your server
certificates with the nsCertType field set to "server".

# The build-key-server script in the easy-rsa folder will do this.
ns-cert-type server

# Enable compression on the VPN link.
comp-lzo

# Set log file verbosity.
verb 9
```

Levanté el servicio ejecutando la aplicación *OpenVPN GUI* con permisos de usuario administrador (ver *Figura 3.15*).

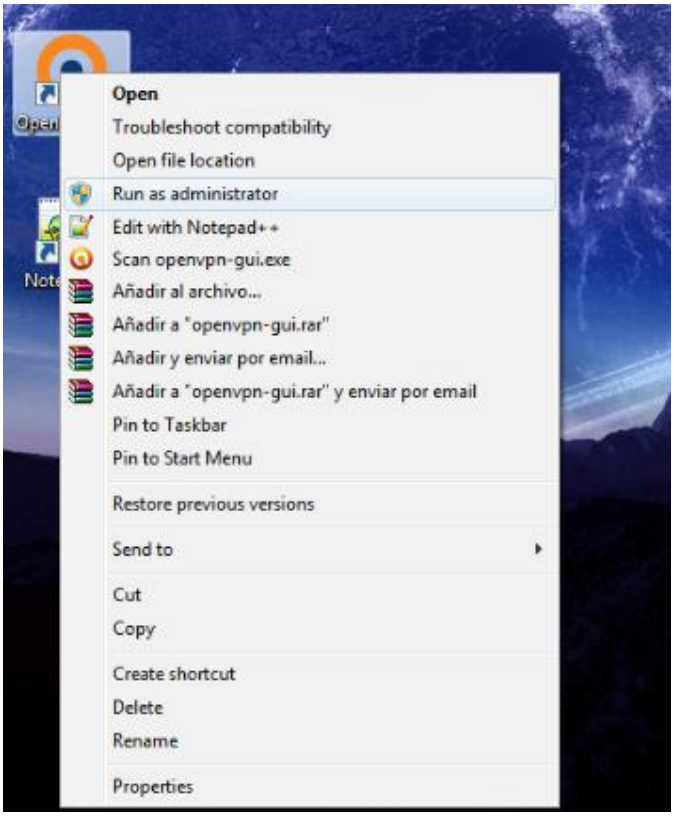

Figura 3.15 - Ejecutando el aplicativo cliente OpenVPN con permisos de administrador.

Después de haber ejecutado el aplicativo, en la parte inferior izquierda del escritorio se mostró el proceso activo; lo siguiente fue validar nuestro archivo de configuración (ver *Figura 3.16*), para ello realicé los siguientes pasos:

- Dar *click* derecho sobre el icono de OpenVPN.
- Seleccioné la pestaña *Edit Config* para abrir el archivo de configuración principal.
- Realicé la edición del archivo con las líneas mostradas anteriormente.
- Guardé los cambios.

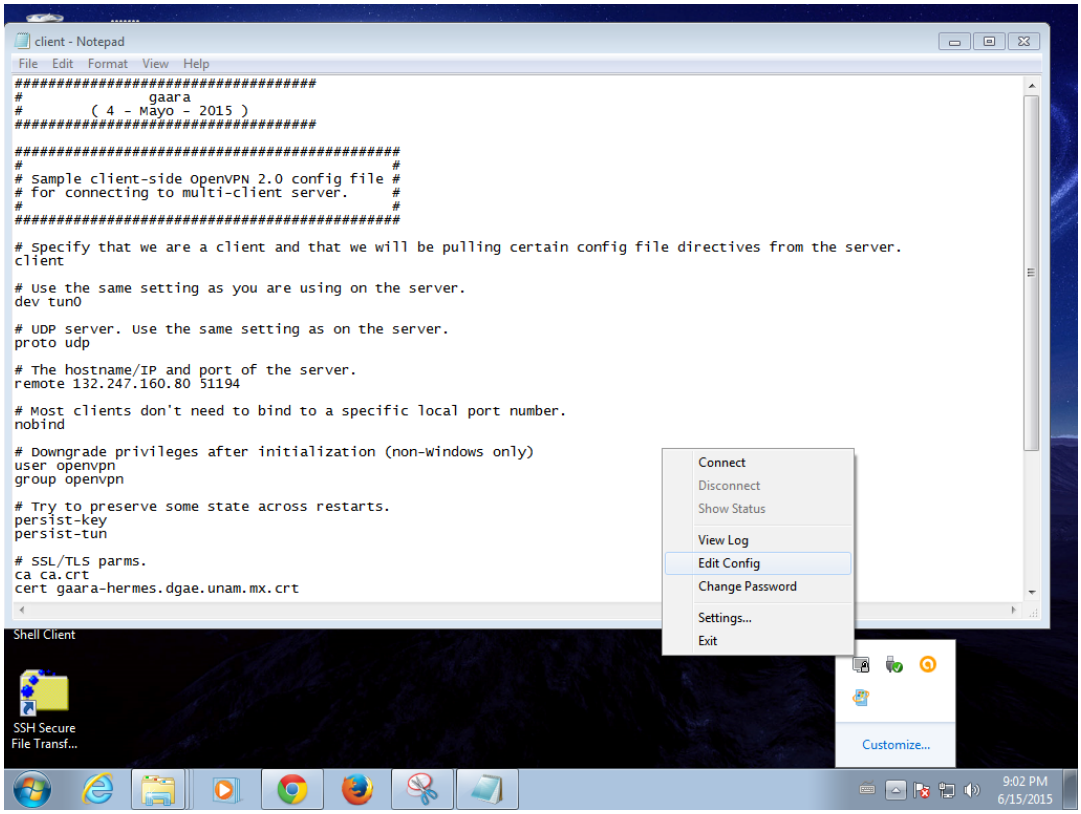

Figura 3.16 - Configuración del archivo principal del cliente OpenVPN.

Para establecer la conexión nuevamente abrí las opciones del icono de *OpenVPN GUI* y seleccioné la pestaña *connect* (ver *Figura 3.17*).

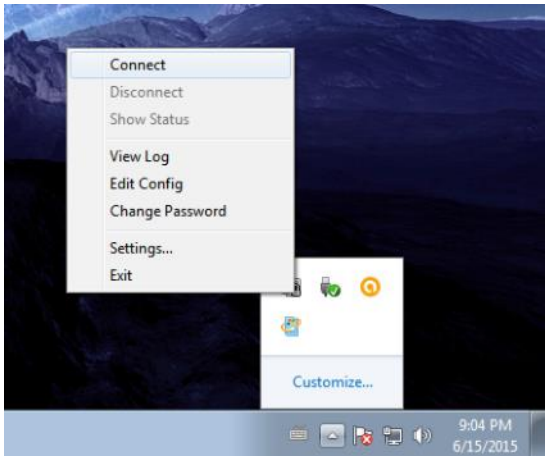

Figura 3.17 - Realizando conexión con el servidor OpenVPN desde el cliente Windows.

Realizada la conexión, me solicitó las credenciales del certificado (ver *Figura 3.18*).

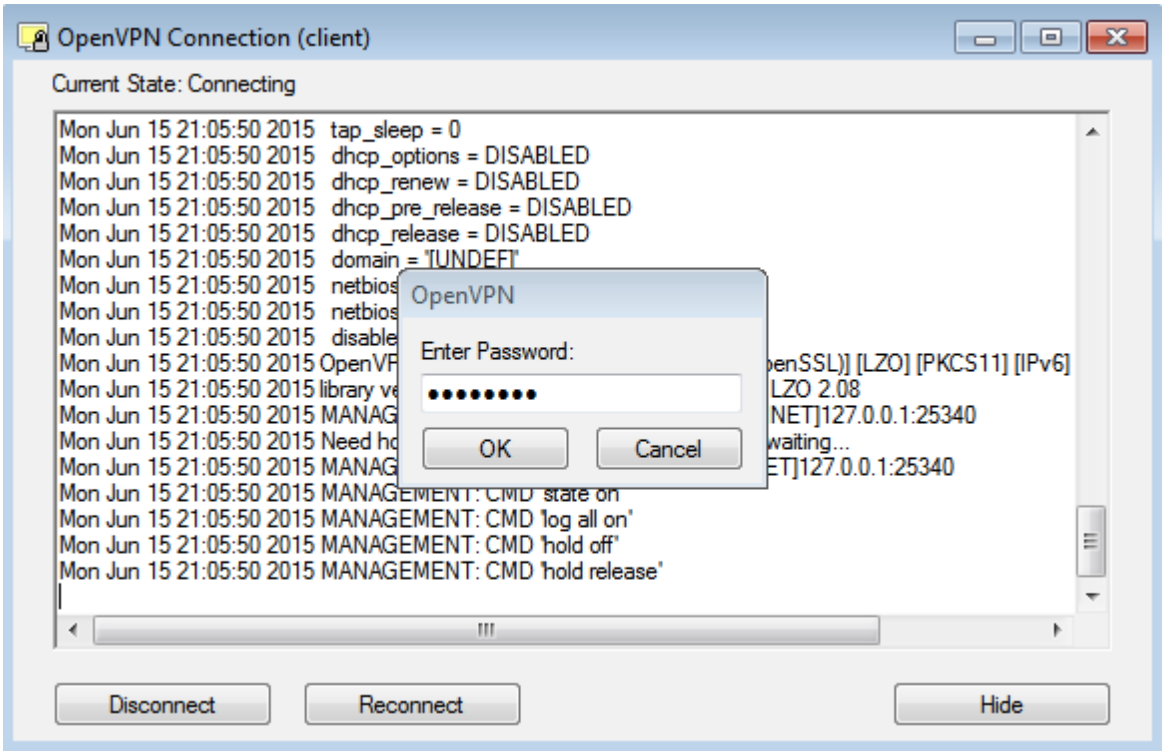

Figura 3.18 - Solicitud de las credenciales del certificado para establecer la conexión con la VPN.

Despues de haber introducido las credenciales, en la parte inferior izquierda del escritorio (ver *Figura 3.19*) se mostró las direcciones IP asignadas por parte de la conexión VPN.

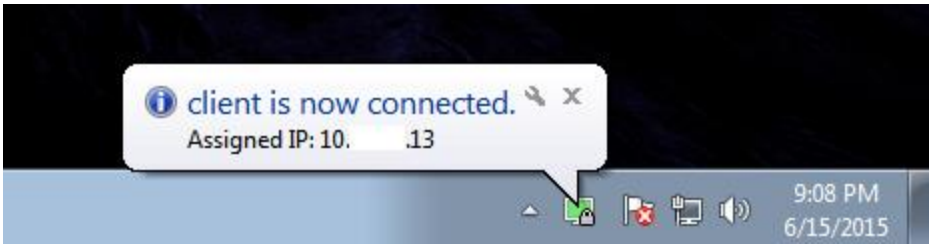

Figura 3.19 - Conexión exitosa con el servidor OpenVPN desde un cliente en Windows.

Ya establecida la conexión ejecuté el script add_routes.bat (ver *Figura 3.20*) con permisos de administrador; realiza lo mismo que el script add_routes.sh, pero con la única diferencia de que está escrito en el lenguaje de procesamiento de lotes de *MS-DOS*.

- Agrega direcciones IP a nuestra tabla de ruteo del equipo cliente.
    - Solicita información para llevar a cabo esta parte.
- Con esto la tabla de ruteo conocerá qué direcciones IP buscar por la interface virtual creada para OpenVPN y qué direcciones IP por la interface por defecto.
- El código del script se puede visualizar en el **Anexo A. Script Windows**.

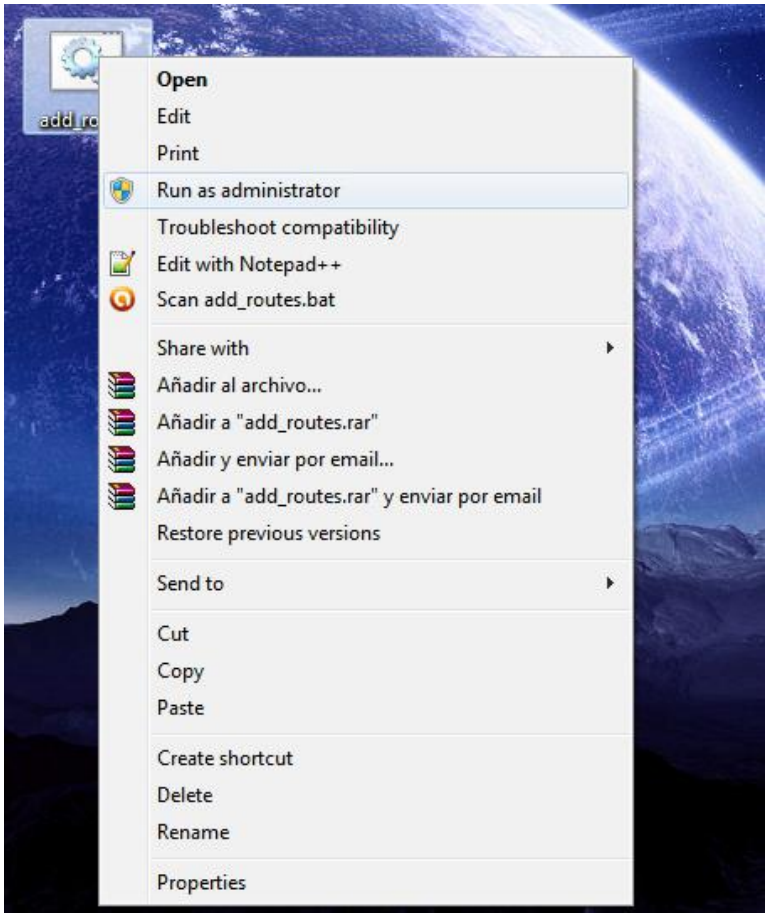

Figura 3.20 - Ejecución del script add_routes.bat para sumar direcciones IP a la tabla de ruteo del cliente.

Las opciones y datos que nos solicita el script consisten en:

- Agregar/Eliminar direcciones IP de la tabla de ruteo.
- Red de la dependencia a la que estamos haciendo la conexión mediante OpenVPN.
- Dirección IP asignada por la conexión OpenVPN.
- Puerta de enlace principal a internet (gateway por defecto); ver *Figura 3.21*.
- Dirección IP asignada por la conexión OpenVPN más una suma de un numero 1 al último octeto de la dirección IP (ver *Figura 3.22*).

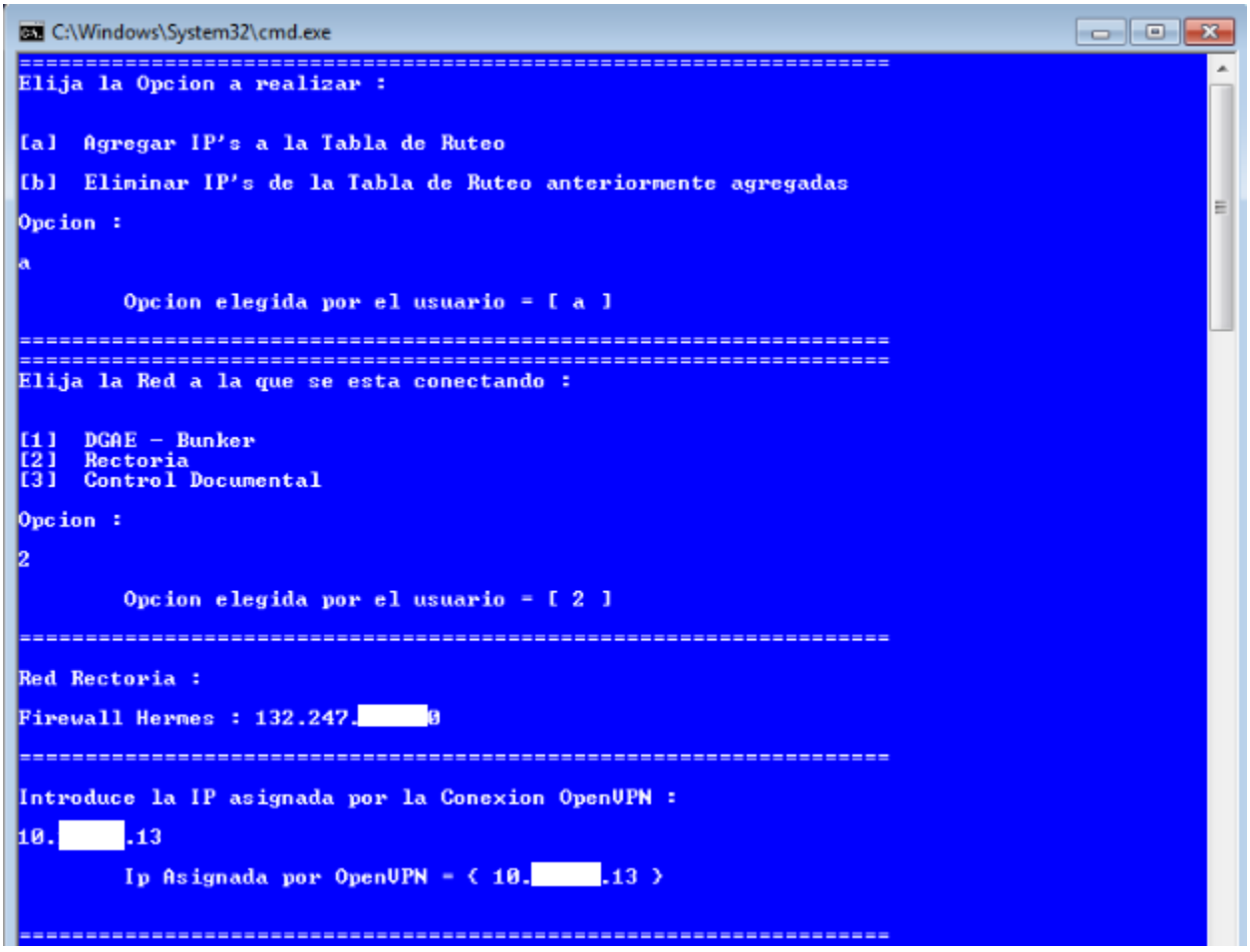

Figura 3.21 - Menú de opciones del script; ingreso de los datos solicitados.

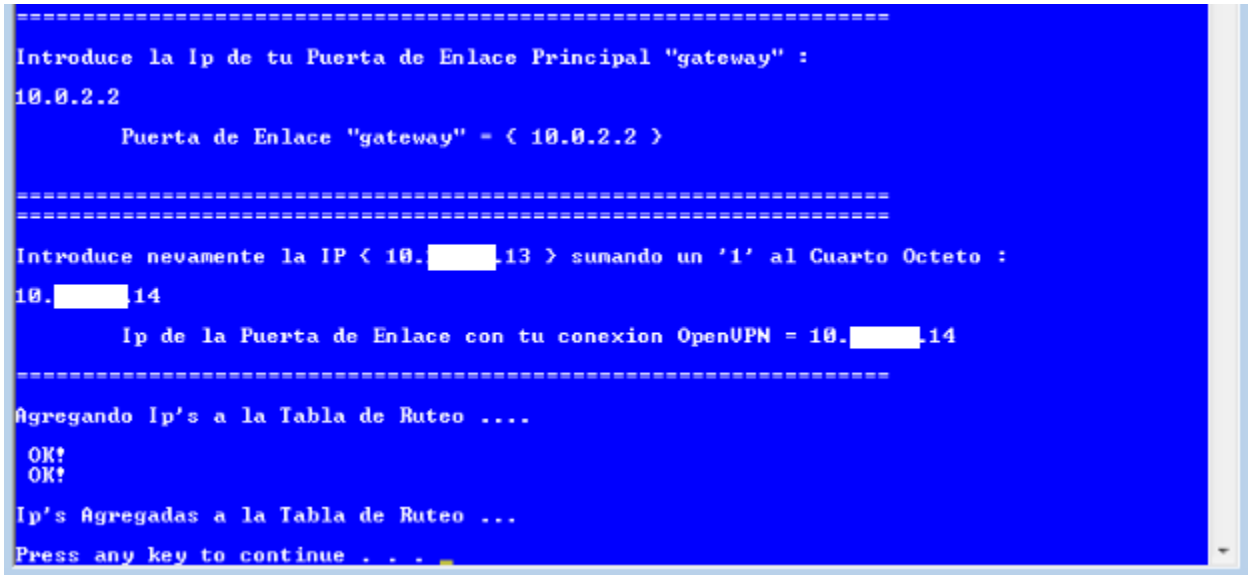

Figura 3.22 - Ingreso de los datos solicitados por el script para establecer la tabla de ruteo.

Y finalmente para terminar con la conexión OpenVPN, de las opciones del icono de *OpenVPN GUI* seleccioné la pestaña *Disconnect* (ver *Figura 3.23*).

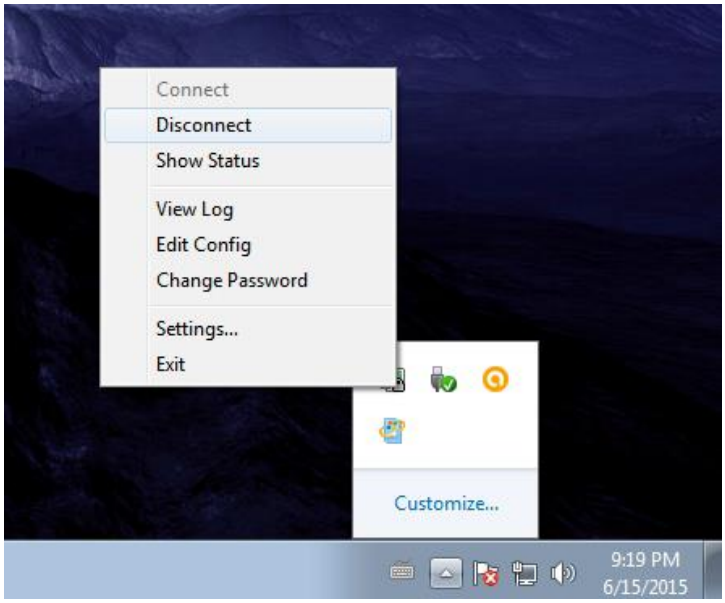

Figura 3.23 - Finalizando la conexión con el servidor OpenVPN.

Nuevamente ejecuté el script add_routes.bat para eliminar las direcciones IP que se agregaron a la tabla de ruteo del cliente (ver *Figura 3.24*).

- Seleccionar opción [b] para eliminar direcciones IP.
- Seleccionar opción [2] que indica eliminar direcciones IP de Rectoría.
- Presionar cualquier tecla para finalizar con la ejecución del script.

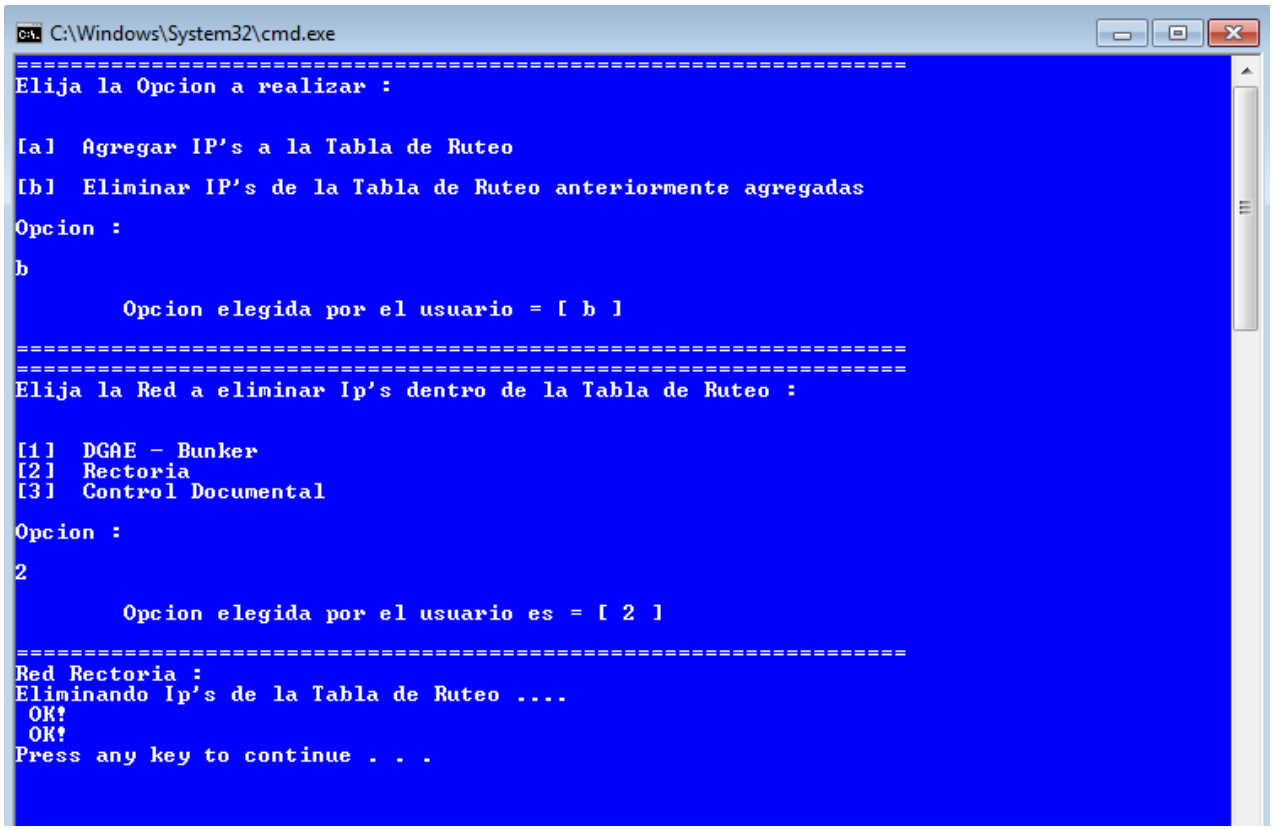

Figura 3.24 - Ejecución del script add_routes.bat con permisos de administrador para eliminar direcciones IP agregadas a la tabla de ruteo.

3.5.9 Revocando Certificados OpenVPN

Es muy importante contar con los mecanismos necesarios para impedir que usuarios no autorizados, que, si en algún momento se hicieron con los certificados y credenciales de un usuario autorizado, puedan realizar conexiones exitosas con la VPN; uno de estos mecanismos consiste en la revocación de certificados en el servidor.

Para el siguiente ejemplo revocaremos un certificado creado anteriormente, este certificado corresponde al usuario cdiaz.

Ajustamos las variables de ambiente en el servidor:

```
[root@hermes-/usr/local/share/easy-rsa]# pwd
/usr/local/share/easy-rsa
[root@hermes-/usr/local/share/easy-rsa]# . ./vars
NOTE: If you run ./clean-all, I will be doing a rm -rf on
/usr/local/share/easy-rsa/keys_llaves
NOTA: Si ejecutas ./clean-all, Estare haciendo un rm -rf on
/usr/local/share/easy-rsa/keys_llaves
[root@hermes-/usr/local/share/easy-rsa]#
```

Nos posicionamos dentro del mismo directorio donde creamos los certificados, ahí reside el comando que le dice a OpenVPN qué certificado revocar, solo es necesario indicar el *Common Name* para llevarlo a cabo:

```
[root@hermes-/usr/local/share/easy-rsa]# ./revoke-full cdiaz-
hermes.dgae.unam.mx
Using configuration from /usr/local/share/easy-rsa/openssl.cnf
Revoking Certificate 05.
Data Base Updated
Using configuration from /usr/local/share/easy-rsa/openssl.cnf
cdiaz-hermes.dgae.unam.mx.crt: C = MX, ST = Mexico, L = Distrito
Federal, O = DGAE - SysAdmin ( J. Manuel
Lopez L. ), CN = cdiaz-hermes.dgae.unam.mx, emailAddress =
seguridad@galois.dgae.unam.mx
error 23 at 0 depth lookup:certificate revoked
```

```
[root@hermes-/usr/local/share/easy-rsa]#
```

- Cuando se muestra (error 23) en la última línea nos está indicando que la validación del certificado a revocar falló, esto indica que el certificado ya no es válido y por lo tanto su revocación se completó exitosamente.

Esto podemos verificarlo en el archivo */usr/local/share/easy-rsa/keys_llaves/index.txt.*

- En la última línea correspondiente al usuario cdiaz aparece la letra *R* la cual indica que el certificado para ese usuario esta revocado.

La ejecución del comando *revoke-full* creó el archivo *keys_llaves/crl.pem*, el cual contiene la cadena del certificado revocado.

- Este archivo debe ser copiado a una ruta donde OpenVPN pueda acceder a él.
- Cada vez que se ejecuta el comando *revoke-full*, el archivo *crl.pem* se sobrescribe con una nueva cadena del certificado a revocar.
- Y nuevamente el archivo *crl.pem* debe copiarse a la ruta donde residen los certificados revocados, pero con la única excepción de que deberá renombrase diferente, por ejemplo: *crl_2.pem*; o en su caso *crl_usuario.pem*.

```
[root@hermes-/etc/openvpn]# mkdir revoked_certificates
[root@hermes-/usr/local/share/easy-rsa]#     cp     keys_llaves/crl.pem
/etc/openvpn/revoked_certificates/
```

Al archivo de configuración del servidor OpenVPN debemos agregar las siguientes líneas que habilitarán la lectura del archivo *crl.pem*.

- Si se revocara otro certificado se deberá agregar otra línea donde se indica la ruta al segundo archivo con la cadena del certificado revocado; como ejemplo puede que este archivo lleve por nombre: *crl_2.pem* o en su caso *crl_usuario.pem*.

```
[root@hermes-/etc/openvpn]# vim server.conf
….....................................
….....................................
# Enable Verification 'CRL' ( This file contains a list about revoked
certificates )
crl-verify /etc/openvpn/revoked_certificates/crl.pem
[root@hermes-/etc/openvpn]#
```

- Es necesario reiniciar el servicio OpenVPN para que tome los nuevos cambios realizados.
- Otra recomendación es eliminar los archivos relacionados al certificado revocado, esto asegura que aun cuando el usuario tenga todo lo necesario para realizar la conexión a la VPN esta no será satisfactoria puesto que en el servidor los archivos de configuración para dicho usuario no existen más.

3.6 Packet Filter

Packet Filter (PF) es un sistema utilizado para filtrar tráfico de red; viene preinstalado en OpenBSD, lo cual es idóneo para configurar reglas de firewall. En la dirección url *https://www.openbsd.org/faq/pf/index.html* se encuentra la documentación necesaria para trabajar con PF.

Para la creación de las reglas realicé un escrito de los requerimientos de los responsables de cada departamento. A estos requerimientos se sumaron reglas para la conexión VPN.

El archivo principal de configuración es:

/etc/pf.conf

Sin embargo, PF nos permite crear diferentes archivos para poder separar las reglas del firewall. Con la revisión de los requerimientos hecha, las reglas para el firewall son:

- Direcciones IP con acceso a internet (sin restricciones).
- Direcciones IP con acceso solo a red UNAM.
- Direcciones IP con acceso solo a un limitado número de páginas web.
- Conexión de direcciones IP externas a OpenVPN (se restringe acceso con certificados).

Estructuré los archivos de la siguiente manera:

- /etc/pf.conf
    - /etc/ips_Internet
    - /etc/ips_Restringidas
    - /etc/ips_AccesoWeb
    - /etc/PaginasWeb_ips

Por seguridad de los departamentos de la Dirección General de Administración Escolar en Rectoría no se mostrarán los archivos de configuración para las reglas que protegen la integridad de la red de datos.

3.7 Configuración del Switch

3.7.1 Características

El equipo adquirido por la DGAE (cumpliendo con las especificaciones mínimas solicitadas para el proyecto) corresponde a un switch de 28 puertos de la marca Cisco. Este dispositivo es administrable y configurable en las capas 2 y 3 referentes al Modelo OSI.

- Modelo: Cisco Catalyst 2960S-24TS-L.
- Puertos: 24 Ethernet 10/100/1000 ports; 4 One Gigabit Ethernet SFP ports.
    - 3 conexiones físicas mediante Gbic (Giga Bit Interface Converter)
        - GigabitEthernet 1/0/25
        - GigabitEthernet 1/0/26
        - GigabitEthernet 1/0/27

Los demás switches de interconexión para la distribución de red de datos a los usuarios son de capa 2.

3.7.2 Distribución de nodos de Red

Se llevó a cabo la instalación de dos enlaces nuevos de fibra óptica para Rectoría como parte de un proyecto de la UNAM; esto dió la posibilidad de concentrar todo el tráfico de red en el switch Cisco; se realizó la instalación de 9 nodos de red distribuidos en las oficinas ocupadas por el personal de la DGAE. El objetivo de la instalación de 9 nodos de red fue utilizar un estandar UTP mas alto en comparación con el que se contaba en la infraestructura de red anterior.

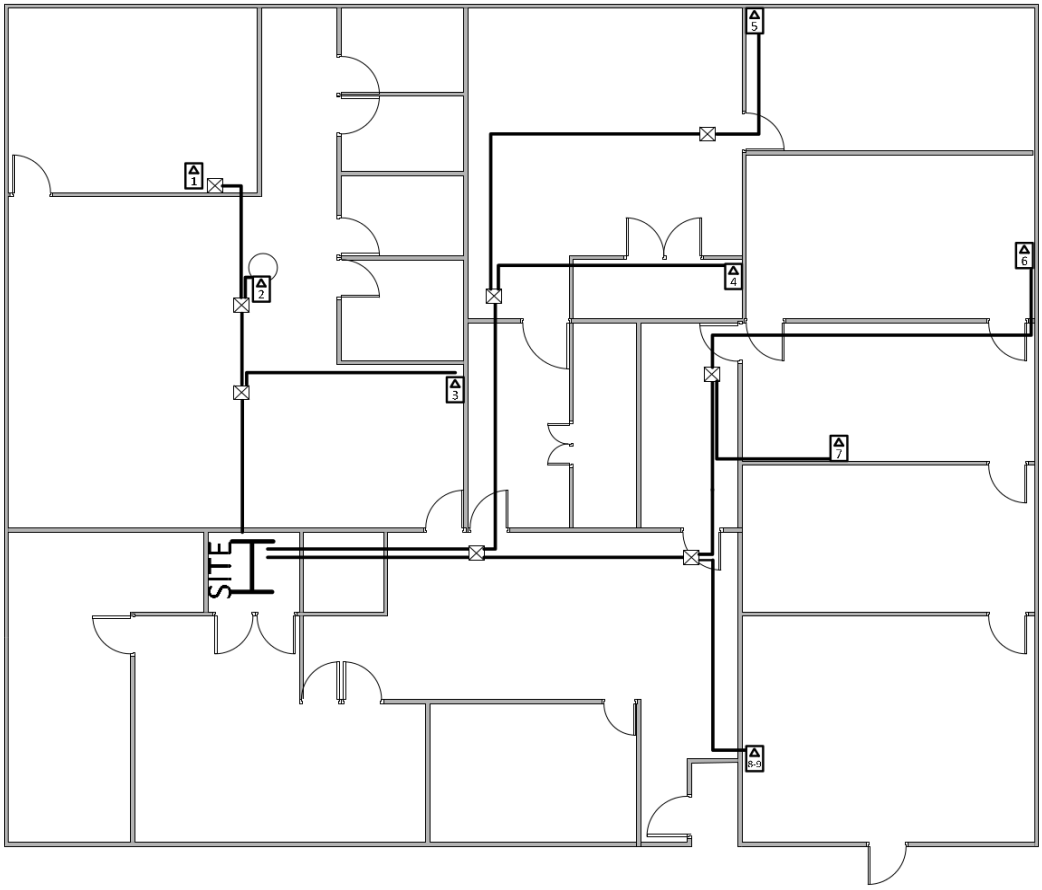

Figura 3.25 - Distribución y ubicación física de nodos de red para DGAE Rectoría.

La instalación fue realizada por la empresa Diversicom; la *Figura 3.25* representa la distribución y ubicación de estos nodos.

En la *Tabla 3.3* se encuentra la distribución que realicé para las conexiones físicas al switch Cisco, así como la asignación a nivel lógico de Virtual Local Area Network (VLAN).

Tabla 3.3 - Conexiones físicas y asignación lógica de puertos en switch Cisco.

Interface Switch	Nodo de Red	VLAN	Descripción
GigabitEthernet1/0/1	D-01	**305**	Jefe - Unidad Administrativa
GigabitEthernet1/0/2	D-02	**305**	Personal - Unidad Administrativa
GigabitEthernet1/0/3	D-03	**305**	Personal - Unidad Administrativa, Gestión Estratégica y Primer Ingreso
GigabitEthernet1/0/4	D-04	**305**	Personal - Depto. de Planes y Programas de Estudio
GigabitEthernet1/0/5	D-05	**305**	Personal - Oficina Depto. de Planes y Programas de Estudio
GigabitEthernet1/0/6	D-06	**305**	Director - Gestión Estratégica y Primer Ingreso
GigabitEthernet1/0/7	D-07	**305**	Secretaria - Director General
GigabitEthernet1/0/8	D-08	**306**	AccessPoint - Director General
GigabitEthernet1/0/9	D-09	**305**	Director General
GigabitEthernet1/0/10		**1**	
GigabitEthernet1/0/11		**1**	
GigabitEthernet1/0/12		**1**	
GigabitEthernet1/0/13		**1**	
GigabitEthernet1/0/14		**1**	
GigabitEthernet1/0/15		**1**	
GigabitEthernet1/0/16		**1**	
GigabitEthernet1/0/17		**1**	
GigabitEthernet1/0/18		**1**	

GigabitEthernet1/0/19		**1**	
GigabitEthernet1/0/20		**1**	
GigabitEthernet1/0/21		**1**	
GigabitEthernet1/0/22		**306**	Firewall ‘hermes’ – tarjeta de red **em2**
GigabitEthernet1/0/23		**305**	Firewall ‘hermes’ – tarjeta de red **em1**
GigabitEthernet1/0/24		**300**	Firewall ‘hermes’ – tarjeta de red **em0**
GigabitEthernet1/0/25		**300**	Enlace de fibra óptica – SFP port
GigabitEthernet1/0/26		**300**	Enlace de fibra óptica – SFP port
GigabitEthernet1/0/27		**1**	
GigabitEthernet1/0/28		**1**	

3.7.3 Configuración Inicial

Llevé a cabo la configuración del switch cisco teniendo en cuenta estos aspectos:

- Dentro del rango de direcciones IP utilizables asigné dos direcciones IP para los enlaces de fibra óptica:
  - 132.248.xxx.xx3
  - 132.248.xxx.xx2
- Contar con 3 Virtual LAN (VLAN) para separar el tráfico de red; estas VLANs son:
  - 1 VLAN capa 2 y 3 (Modelo OSI)
    - Id: Vlan 300
    - Nombre: Red_Rectoria
    - Se asigna la dirección IP 132.248.xxx.xx0 como gateway de la VLAN 300, esta dirección IP será la de administración del switch.
  - 2 VLAN capa 2 (Modelo OSI)

El acceso a la consola de administración del switch la llevé a cabo con las herramientas listadas a continuación:

- 1 laptop con Windows 7
- Software “PuTTY.exe”
- 1 cable USB a mini USB
- Software “Cisco_usbconsole_driver_3_1.zip”

Una vez conectados al puerto consola cambié la sesión a modo configuración:

```
Switch>enable
Switch#
```

Eliminé la configuración existente para luego reiniciar el switch:

```
Switch#erase startup-config
Erasing the nvram filesystem will remove all configuration files!
Continue? [confirm]
[OK]
Erase of nvram: complete
Switch#
Oct 1 21:21:05.658: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of
nvram
Switch#
Switch#reload
Proceed with reload? [confirm]
Oct 1 21:21:30.027: %SYS-5-RELOAD: Reload requested by console. Reload
Reason: Reload command.
[…..]
Switch#
```

Configuré el *hostname* del equipo:

```
Switch# configure terminal
Switch(config)#
Switch(config)# hostname Rectoria_DGAE
Rectoria_DGAE(config)#
```

Definí los banners para *login* e inicio de sesión:

- login
  - imprime una leyenda antes de introducir las credenciales.
- sesión
  - imprime una leyenda después de haber introducido las correctas credenciales para inicio de sesión.

```
Rectoria_DGAE(config)#banner motd %
Enter TEXT message. End with the character '%'.
------------------------------------------------------------
Direccion General de Administracion Escolar - Rectoria
------------------------------------------------------------
_,'| _.-''''-..._____..--';)
/_ \'. __..-' , ,--...--'''
<\ .'--''' ' /'
'-';' ; ; ;
__...--'' ___...--_..' .;.'
(,__....----''' (,..--''
---------
gaara
---------
Bienvenido al Switch - Rectoria ...!
% Rectoria_DGAE(config)#
Rectoria_DGAE(config)#banner login %
Enter TEXT message. End with the character '%'.
Estas Autorizado ! ...
....
.....
% Rectoria_DGAE(config)#
```

Ajusté Zona horaria, Fecha y Hora:

```
Rectoria_DGAE(config)#clock timezone GMT -6
Rectoria_DGAE(config)#clock summer-time GMT recurring 1 Sun Apr 2:00
last Sun Oct 2:00
Rectoria_DGAE(config)#exit
Rectoria_DGAE#
Oct 1 21:57:07.633: %SYS-5-CONFIG_I: Configured from console by console
Rectoria_DGAE#show clock
21:57:10.252 GMT Thu Oct 1 2015
```

```
Rectoria_DGAE# clock set 16:57:00 1 Oct 2015
Rectoria_DGAE# show clock
16:57:19.252 GMT Thu Oct 1 2015
Rectoria_DGAE#
```

3.7.4 Usuarios y Privilegios

Para la gestión de usuarios y privilegios:

- Usualmente se asignan solo contraseñas para la administración del switch cuando:
    - Se inicia sesión por puerto consola.
    - Pasar a modo configuración del switch con el comando *enable*.
- Para la administración del switch habilité específicamente el uso de usuario y contraseña.

Habilité la protección de contraseñas mediante cifrado, creé los usuarios y el inicio de sesión por puerto consola:

```
Rectoria_DGAE(config)#service password-encryption
Rectoria_DGAE(config)#username usuario1 privilege 15 password
usuario123
Rectoria_DGAE(config)#username usuario2 privilege 15 password
usuario321
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#line console 0
Rectoria_DGAE(config-line)#login local
Rectoria_DGAE(config-line)#exit
Rectoria_DGAE(config)#
```

El uso de las Terminales Virtuales para conexiones remotas al switch requiere de los usuarios y contraseñas creados:

```
Rectoria_DGAE(config)#line vty 0 4
Rectoria_DGAE(config-line)#login local
Rectoria_DGAE(config-line)#exit
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#line vty 5 15
Rectoria_DGAE(config-line)#login local
Rectoria_DGAE(config-line)#exit
Rectoria_DGAE(config)#
```

3.7.5 Configuración de VLANs

A partir de los datos de red mencionados en capítulos anteriores creé la VLAN 300 en la capa 2 y 3 referente al Modelo OSI:

```
Rectoria_DGAE(config)#vlan 300
Rectoria_DGAE(config-vlan)#name Red_Rectoria
Rectoria_DGAE(config-vlan)#exit
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#interface vlan 300
Rectoria_DGAE(config-if)#
Oct 1 22:10:05.956: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Vlan300, changed state to down
Rectoria_DGAE(config-if)#ip address 132.248.xxx.xx0 255.255.255.128
Rectoria_DGAE(config-if)#description Red_Local-Rectoria
Rectoria_DGAE(config-if)#no shutdown
Rectoria_DGAE(config-if)#exit
Rectoria_DGAE(config)#
```

Configuración de las VLANs 305 y 306 en la capa 2 referente al Modelo OSI:

```
Rectoria_DGAE(config)#vlan 305
Rectoria_DGAE(config-vlan)#name Rectoria_Usuarios
Rectoria_DGAE(config-vlan)#exit
Rectoria_DGAE(config)#

Rectoria_DGAE(config)#vlan 306
Rectoria_DGAE(config-vlan)#name Rectoria_AccessPoint
Rectoria_DGAE(config-vlan)#exit
Rectoria_DGAE(config)#
```

Deshabilité el ruteo de IP, agregué la ruta por defecto (gateway) que todo paquete de red seguirá para alcanzar los destinos no listados en la tabla de ruteo del switch; la dirección IP es necesariamente alguna de las dos asignadas para los enlaces de fibra óptica.

```
Rectoria_DGAE(config)#no ip routing

Rectoria_DGAE(config)#ip default-gateway 132.248.xxx.xx3
Rectoria_DGAE(config)#
```

3.7.6 Configuración Secure SHell (ssh)

Para las conexiones remotas habilité el protocolo *ssh* el cual nos permitirá administrar el switch de forma segura; la versión a utilizar es la 2 y el tamaño de bits para la llave la manejé de 1024.

Este es el único medio por el que se puede ingresar a la consola de administración del switch, otros protocolos no fueron tomados en cuenta de acuerdo con los requerimientos de los responsables.

Me aseguré que no estuviera habilitada alguna versión de *ssh* en el switch; hecho esto, creé un dominio y generé las llaves RSA (Rivest, Shamir y Adleman) para poder habilitar el protocolo SSH.

```
Rectoria_DGAE#show ip ssh
SSH Disabled - version 1.99
%Please create RSA keys to enable SSH (and of atleast 768 bits for SSH
v2).
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
IOS Keys in SECSH format(ssh-rsa, base64 encoded): NONE
Rectoria_DGAE(config)#
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#ip domain-name rectoria-dgae.unam.mx
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#crypto key generate rsa
The name for the keys will be: Rectoria_DGAE.rectoria-dgae.unam.mx
Choose the size of the key modulus in the range of 360 to 4096 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.
How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 1 seconds)
Rectoria_DGAE(config)#
Oct 1 22:24:28.567: %SSH-5-ENABLED: SSH 1.99 has been enabled
Rectoria_DGAE(config)#
```

Habilité la versión 2 del protocolo *ssh* indicando a las terminales virtuales el uso de *ssh* para las conexiones remotas.

```
Rectoria_DGAE(config)#ip ssh version 2
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#line vty 0 4
Rectoria_DGAE(config-line)#transport input ssh
Rectoria_DGAE(config-line)#exit
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#line vty 5 15
Rectoria_DGAE(config-line)#transport input ssh
Rectoria_DGAE(config-line)#exit
Rectoria_DGAE(config)#
```

Con las nuevas tecnologías que implementa Cisco en sus equipos es posible administrar el switch haciendo uso de los protocolos *http* y/o *https*, sin embargo, estas opciones las deshabilité dejando solamente el protocolo *ssh*.

```
Rectoria_DGAE(config)#no ip http server
Rectoria_DGAE(config)#no ip http secure-server
```

3.7.7 Listas de Acceso

Las Listas de Control de Acceso (ACL - Access Control List) son mecanismos que ayudan a determinar los permisos de acceso al switch filtrando tráfico de red entrante y/o saliente.

Para el acceso a la red filtré conexiones que no pertenecieran a los segmentos de red destinados a la Subdirección de Diseño de Proyectos (SDP).

Para la creación de Listas de Acceso es importante conocer el término *wildcard*; una *wildcard* agrupa a diferentes hosts que pertenecen a diferentes segmentos de red. Para la creación de nuestras Listas de Acceso fue necesario calcular la *wildcard*:

- Segmento de red 132.248.xxx.0/24
    - netmask: 255.255.255.0
    - wildcard: 0.0.0.255
- Dirección Ip 132.247.xxx.x7
    - red: 132.247.xxx.xx/xx
        - netmask: 255 . 255 . 255 . 248
        - representación bits: 11111111.11111111.11111111.11111000
        - wildcard bits: 00000000.00000000.00000000.00000111
        - wildcard: 0 . 0 . 0 . 7

Inicialmente creé la Lista de Acceso para conexiones remotas *ssh* para después aplicar esta ACL a las terminales virtuales.

```
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#ip access-list standard permit_ssh
Rectoria_DGAE(config-std-nacl)#permit 132.248.xxx.0 0.0.0.255
Rectoria_DGAE(config-std-nacl)#exit
Rectoria_DGAE(config)#exit
Rectoria_DGAE#
Rectoria_DGAE#show access-lists
Standard IP access list permit_ssh
10 permit 132.248.xxx.0, wildcard bits 0.0.0.255
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#line vty 0 4
Rectoria_DGAE(config-line)#access-class permit_ssh in
Rectoria_DGAE(config-line)#exit
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#line vty 5 15
Rectoria_DGAE(config-line)#access-class permit_ssh in
Rectoria_DGAE(config-line)#exit
Rectoria_DGAE(config)#end
Rectoria_DGAE#
```

La instalación del agente *snmp* la llevé a cabo junto con la creación de una ACL destinada a filtrar conexiones al puerto 161; en esta parte opté por crear una comunidad de solo lectura.

```
Rectoria_DGAE#
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#ip access-list standard permit_snmp
Rectoria_DGAE(config-std-nacl)#permit 132.247.xxx.xx 0.0.0.7
Rectoria_DGAE(config-std-nacl)#exit
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#snmp-server community comunidad ro permit_snmp
Rectoria_DGAE(config)#end
Rectoria_DGAE#
```

3.7.8 Configuración de Puertos

La última parte fue la asignación (ver *Figura 3.28*) y configuración de los puertos del switch para las conexiones físicas con los demás switches y el firewall; retomando las VLANs existentes:

- VLAN 1 (vlan por defecto)
- VLAN 300 (vlan principal para salida a internet)
- VLAN 305 (vlan de usuarios DGAE - Rectoría)
- VLAN 306 (vlan de AccessPoint - Dirección General)

De acuerdo con el esquema de red propuesto, el firewall hermes cuenta con tres conexiones al switch cisco, estas conexiones forman los puentes lógicos entre VLANs. A estos puertos les apliqué el modo *switchport host*, lo que hace este comando es:

- Modo PortFast
- Deshabilita la negociación Trunking
- Deshabilita EtherChannel

Realicé la asignación de la siguiente manera para el firewall:

- Interface de red em0 (GigabitEthernet1/0/24 - VLAN 300)

```
Rectoria_DGAE(config)#interface gigabitethernet 1/0/24
Rectoria_DGAE(config-if)#switchport access vlan 300
Rectoria_DGAE(config-if)#switchport host
switchport mode will be set to access
spanning-tree portfast will be enabled
channel group will be disabled
Rectoria_DGAE(config-if)#no shutdown
Rectoria_DGAE(config-if)#exit
Rectoria_DGAE(config)#
```

- Interface de red em1 (GigabitEthernet1/0/23 - VLAN 305)

```
Rectoria_DGAE(config)#interface gigabitEthernet 1/0/23
Rectoria_DGAE(config-if)#switchport access vlan 305
Rectoria_DGAE(config-if)#switchport host
switchport mode will be set to access
spanning-tree portfast will be enabled
channel group will be disabled
Rectoria_DGAE(config-if)#no shutdown
Rectoria_DGAE(config-if)#exit
Rectoria_DGAE(config)#
```

- Interface de red em2 (GigabitEthernet1/0/22 - VLAN 306)

```
Rectoria_DGAE(config)#interface gigabitEthernet 1/0/22
Rectoria_DGAE(config-if)#switchport access vlan 306
Rectoria_DGAE(config-if)#switchport host
switchport mode will be set to access
spanning-tree portfast will be enabled
channel group will be disabled
Rectoria_DGAE(config-if)#no shutdown
Rectoria_DGAE(config-if)#exit
Rectoria_DGAE(config)#
```

Para la asignación de los puertos a los demás switches:

- Asignación VLAN 305

```
Rectoria_DGAE#
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#interface range gigabitEthernet 1/0/1-7
Rectoria_DGAE(config-if-range)#switchport access vlan 305
Rectoria_DGAE(config-if-range)#no shutdown
Rectoria_DGAE(config-if-range)#end
Rectoria_DGAE#
```

- Asignación VLAN 306

```
Rectoria_DGAE#
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#interface range gigabitEthernet 1/0/1-7
Rectoria_DGAE(config-if-range)#switchport access vlan 305
Rectoria_DGAE(config-if-range)#no shutdown
Rectoria_DGAE(config-if-range)#end
Rectoria_DGAE#
```

- Asignación VLAN 305

```
Rectoria_DGAE#
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#interface gigabitEthernet 1/0/9
Rectoria_DGAE(config-if)#switchport access vlan 305
Rectoria_DGAE(config-if)#no shutdown
Rectoria_DGAE(config-if)#end
Rectoria_DGAE#
```

- Deshabilitar los puertos sin conexión; automáticamente pertenecerán a la VLAN 1

```
Rectoria_DGAE#
Rectoria_DGAE(config)#interface range gigabitEthernet 1/0/10-21
Rectoria_DGAE(config-if-range)#shutdown
Rectoria_DGAE(config-if-range)#end
Rectoria_DGAE#
```

Finalmente asigné los puertos (SFP) de los enlaces de fibra óptica a la VLAN 300.

- Solo se necesitaron dos puertos, uno para el enlace principal y otro para el enlace secundario, los otros dos puertos fueron deshabilitados.

```
Rectoria_DGAE#
Rectoria_DGAE#configure terminal
Rectoria_DGAE(config)#interface range gigabitethernet 1/0/25-26
Rectoria_DGAE(config-if-range)#switchport access vlan 300
Rectoria_DGAE(config-if-range)#shutdown
```

```
Rectoria_DGAE(config-if-range)#no shutdown
Rectoria_DGAE(config-if-range)#exit
Rectoria_DGAE(config)#
Rectoria_DGAE(config)#interface range gigabitethernet 1/0/27-28
Rectoria_DGAE(config-if-range)#shutdown
Rectoria_DGAE(config-if-range)#exit
Rectoria_DGAE(config)#end
Rectoria_DGAE#
```

Por último y de mucha relevancia fue realizar la copia de la configuración a memoria del switch y los respaldos de dicha configuración. El switch cuenta con dos mecanismos de configuración:

- running-config
    - Configuración leída por el switch en tiempo real; si el switch es reiniciado y no se escribió esta configuración a memoria interna, los últimos cambios serán descartados.
- startup-config
    - Configuración escrita en memoria interna del switch; si el switch es reiniciado este levantará con las configuraciones guardadas en memoria interna.

Todas las configuraciones hechas se mantenían en running-config, lo cual solo bastó con guardarlas a la startup-config'

- De igual forma reinicié el switch para corroborar que la nueva configuración guardada en la startup-config fuese la inicial.

```
Rectoria_DGAE#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Rectoria_DGAE#

Rectoria_DGAE#reload
```

Capítulo 4. Resultados

En este último capítulo, presento los resultados obtenidos por parte del proyecto principal y los cambios que se vieron reflejados con su puesta en producción; más adelante presento las conclusiones obtenidas y algunos puntos de interés a implementar.

El cambio de los enlaces de fibra óptica incrementó el ancho de banda de 100Mb/s a 1000Mb/s, además, el concentrar todo el tráfico en un switch con un Backplane mínimo de 100 Gbps incrementó la velocidad de transferencia de la red de datos para los usuarios.

El haber incluido un switch administrable para el proyecto dio la gran ventaja de poder mejorar el control de la red de datos, además con la implementación de la tecnología VLAN se abrió la posibilidad de separar la red de los usuarios con la red inalámbrica destinada a visitantes; también se mantiene la puerta abierta para poder volver a separar o segmentar la red si fuese el caso necesario.

Con el cambio de las conexiones principales hacia los demás switches se aseguró la transferencia de datos sin pérdidas causadas por el medio físico; cabe mencionar que durante el cambio se detectaron 2 cables seriamente deteriorados lo cual pudo haber provocado un deterioro en las señales que en ellos se transmitían.

La implementación del Firewall permite controlar las conexiones entrantes y salientes de los usuarios, añade una capa de seguridad importante al proteger cualquier petición mal intencionada del exterior hacia la red de los usuarios y de manera inversa, denegar cualquier petición no permitida de los usuarios al exterior.

La VPN instalada se convirtió en una herramienta indispensable para la administración remota de la red; también un medio para la navegación segura (usuarios permitidos) por Internet desde cualquier punto del planeta.

Conclusiones

La implementación de mecanismos de seguridad para las diferentes Subdirecciones de la Dirección General de Administración Escolar fortalece el compromiso que se tiene con la UNAM al resguardar y proteger la información de los alumnos con la que se cuenta.

Es claro que con el avance de nuevas tecnologías nacen nuevas amenazas que están a la espera de poder aprovechar cualquier vulnerabilidad existente en los servicios digitales; haber implementado mecanismos de monitoreo nos mostró que la DGAE no está exenta de intentos de intrusión e intentos de robo de información.

Mejorar las características de los servicios no solo permitió asegurarlos, también permitió asegurar el servicio al público en general y más importante a la comunidad UNAM.

Una de las metas de la Subdirección de Diseño de Proyectos es implementar en tiempo y forma un Sistema de Gestión de la Seguridad de la Información, y con el avance de los proyectos y su implementación estamos dando los primeros pasos para lograrlo.

Actualmente los proyectos presentados en este documento han satisfecho las necesidades que la DGAE ha encomendado a la SDP, sin embargo, aún falta un camino amplio para poder implementar mecanismos de seguridad que cumplan con los requerimientos básicos de la seguridad de la información.

Respecto a lo comentado, haciendo uso de Software Libre es posible implementar mecanismos de seguridad de la información.

El apoyo brindado por la estructura administrativa desde sus más altos funcionarios fue un factor importante para poder implementar estos mecanismos y para poder contar con la compresión de los usuarios finales que son quienes nos apoyan para fortalecer la seguridad de la DGAE.

Glosario

A

AccessPoint

Punto de Acceso inalámbrico para conexión a internet. 41, 100, 106, 110

Apache

Servidor Web HTTP. 19, 23, 27, 28

B

Backplane

Capacidad de transferencia interna. 45, 115

backup

Copia de seguridad. 52

bash

Bourne again shell 20, 29, 47, 48, 52, 58, 79

Batch

Secuencias de ordenes interpretadas por sistemas ms-dos. 90

Bridge

Puente lógico creado por dos interfaces de red. 36, 37

C

compilación

Acción de generar código objeto para una plataforma específica. 25

crontab

Administra tareas programadas del sistema. 53

D

DNS

Domain Name System 50

dominio

Dirección web de la organización presentada en la Internet. 16, 17, 107

F

Firewall

Dipositivo capaz de filtrar tráfico de red que entra y sale. 5, 6, 9, 10, 16, 19, 20, 32, 33, 36, 38, 39, 41, 45, 46, 49, 51, 52, 54, 55, 59, 60, 64, 76, 97, 101, 110, 115

H

Hardening

Proceso para asegurar un Sistema Informatico. 19, 24, 36, 37

I

Ip

Es un numero de 4 grupos asigando a la interface de red que lo identifica en una red de datos. .. 19, 20, 21, 27, 30, 36, 37, 40, 41, 46, 47, 49, 60, 70, 73, 74, 75, 79, 80, 81, 83, 89, 90, 91, 93, 97, 101, 106, 108

K

kernel

L

M

N

O

P

R

S

V

Referencias

CISCO. (s.f.). *Support & Downloads*. Obtenido de CISCO: https://www.cisco.com/c/en/us/support/index.html

González Duran, S. (2017). *Crear Certificados SSL para Apache*. Obtenido de LINUXTOTAL.com.mx: https://www.linuxtotal.com.mx/index.php?cont=info_seyre_001

Mazzocchio, D. (20 de 12 de 2009). *Building VPNs on OpenBSD*. Obtenido de Kernel-panic: http://www.kernel-panic.it/openbsd/vpn/

McDougall, A. (15 de 09 de 2014). *Protecting traffic with a BSD-based VPN*. Obtenido de BSD Now: https://www.bsdnow.tv/tutorials/openvpn

OpenBSD. (26 de 06 de 2017). *FAQ*. Obtenido de OpenBSD Frequently Asked Questions: https://www.openbsd.org/faq/index.html

OpenBSD. (05 de 2017). *Setting up OpenVPN (free community version) on OpenBSD*. Obtenido de openbsdsupport: http://openbsdsupport.org/openvpn-on-openbsd.html

OpenBSD. (24 de 06 de 2017). *SSHD(8)*. Obtenido de OpenBSD: https://man.openbsd.org/sshd

OpenSSL. (25 de 05 de 2017). *commands*. Obtenido de OpenSSL Cryptography and SSL/TLS Toolkit: https://www.openssl.org/docs/man1.1.0/apps/

OpenVPN. (2013). *HOWTO*. Obtenido de OPENVPN: https://openvpn.net/index.php/open-source/documentation/howto.html

OpenVPN. (2017). *OpenVPN Community Wiki and Tracker*. Obtenido de WikiStart: https://community.openvpn.net/openvpn

Sourceforge. (26 de 02 de 2013). *NET-SNMP*. Obtenido de Net-SNMP: http://net-snmp.sourceforge.net/

Stackoverflow. (s.f.). *Questions*. Obtenido de stackoverflow: https://stackoverflow.com/questions

Anexos

Anexo A

Script Linux

El siguiente script es utilizado para modificar la tabla de ruteo del cliente conectado a la VPN.

```
#!/bin/bash

#############################################################################################
#                                gaara
#
#                        ( 12 - Mayo - 2015 )
#
#
#
# Se agregan IP's a la tabla de ruteo del cliente conectado a las VPN's
#
# Las IP's agregadas perteneceran al segmenteo en donde el Firewall con OpenVPN Reside.                #
# Al ser IP's publicas, haremos que el equipo cliente las visualize por la Interface de          #
#   red creada por OpenVPN en el equipo cliente ( tun0 ).
#
# Equipos con IP publica que proporcionen servicio en Internet se dejara que el equipo cliente       #
#   los busque por la Interface de Red de default ( eth0 - en la mayoria de los casos ).         #
#
#
#############################################################################################

#====================================
#        Informacion Global
#====================================

#-----------
# Variables
#-----------
#Ip OpenVPN Firewall 'Hades' ( DGAE - Bunker )
ip_Hades=132.248.xxx.xx1

#Ip OpenVPN Firewall 'Hermes' ( Rectoria )
ip_Hermes=132.248.xxx.xx1

#Ip y Gateway OpenVPN Firewall ' ' ( Control Documental )


#--------------------------------
#        Accion a Realizar
#--------------------------------
#Solicitamos al usuario la opcion a realizar con el script
echo -e "-------------------------------------------------------------------------"
echo -e "Elija la Opcion a realizar :                                             "
echo -e "                                                                        "
echo -e "[a]  Agregar IP's a la Tabla de Ruteo                                    "
echo -e "[b]  Eliminar IP's de la Tabla de Ruteo anteriormente agregadas    "
```

```
echo -e "                                                                              "
echo -e "Opcion :                                                                      "
         read opcion_r
echo -e "------------------------------------------------------------------------------"



#=============================================
#       Funciones : Sistemas GNU/Linux
#=============================================

#---------------
# Datos de Red
#---------------
#Funcion: Obtenemos Ip y Gateway asignados a la Interface "tun0" por la Conexion OpenVPN establecida
function Obten_tun0 {

        #Ip y Gateway
        ip_vpnTun0=$(ifconfig tun0 | grep "inet addr" | awk -F "addr:"  '{print $2}' | awk -F " " '{print $1}')
        gw_vpnTun0=$(ifconfig tun0 | grep "inet addr" | awk -F "P-t-P:" '{print $2}' | awk -F " " '{print $1}')

        #Mostramos Informacion acerca de la Interface "tun0"
        echo -e "----------------------------------------"
        echo -e "  Ip Interface tun0      = $ip_vpnTun0   "
        echo -e "  Gateway Interface tun0 = $gw_vpnTun0   "
        echo -e "----------------------------------------"
        echo -e "                                        "
}

#Funcion: Obtenemos Ip asiganada a la Interface " eth* / wlan* "
function Obten_ip {

        #Filtramos la ip del archivo "ip_$interface.txt"
        ip_equipo=$(grep "inet addr" ip_$interface.txt | awk -F " " '{print $2}' | awk -F ":" '{print $2}')

        #Imprimimos ip
        echo -e "Ip Equipo Cliente =      $ip_equipo \n"

}

#Funcion: Obtenemos Gateway
function Obten_gateway {

        #Obtenemos los tres primeros Octetos de la Ip del Equipo
        segmentoCliente=$(echo $ip_equipo | awk -F "." '{print $1"."$2"."$3}')

        #Filtramos el gateway del archivo "netstat_red.txt"
        gatewayCliente=$(grep "$interface" netstat_red.txt | awk -F " " '{print $1","$2}' | grep
"[0.0.0.0,default],$segmentoCliente" | awk -F "," '{print $2}')

        #Imprimimos gateway
        echo -e "Gateway Equipo Cliente = $gatewayCliente \n"

}


#-------
# Ip's
#-------
#Funcion: Agregar las Ip's del Segmento 132.248.xxx.0/24 a la Tabla de Ruteo del Cliente ( DGAE - Bunker )
function Ips_Bunker {

        #Enrutamos la Ip del Firewall 'Hades' por la Interface ' eth_ / wlan_ ' del Cliente
        route add -host $ip_Hades gw $gatewayCliente dev $interface

        #Enrutamos las Ip's que dan servicio en Internet

                #Servidores Web ( Firewall 'transparente' )
                route add -host 132.248.2xx.xx  gw $gatewayCliente dev $interface      #secretarios.dgae.unam.mx
                route add -host 132.248.2xx.xx  gw $gatewayCliente dev $interface      #odin.dgae.unam.mx
                route add -host 132.248.2xx.xx  gw $gatewayCliente dev $interface      #circe.dgae.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #maxwell.dgae.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #escolar0.dgae.unam.mx

                #Servidores Web ( Firewall - Balanceador )
                route add -host 132.248.2xx.xx  gw $gatewayCliente dev $interface      #uther.dgae.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #escolar1.dgae.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #lovelace.dgae.unam.mx
```

```
                route add -host 132.248.xxx.xx gw $gatewayCliente dev $interface      #bessel.dgae.unam.mx

                #Servidores Web ( Firewall - DGOSE )
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #bayes.dgae.unam.mx

                #Firewalls ( Segmento de Red )
                #route add -host 132.248.2xx.xx  gw $gatewayCliente dev $interface      #galois.dgae.unam.mx
                route add -host 132.248.2xx.xx  gw $gatewayCliente dev $interface      #peano.dgae.unam.mx
                route add -host 132.248.2xx.xx  gw $gatewayCliente dev $interface      #escolar-fw.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #peano2.dgae.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #hypatia.dgae.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #newton.dgae.unam.mx
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #andromeda.dgae.unam.mx
                #route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #zeus.dgae.unam.mx

                #Correo
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface      #bell.dgae.unam.mx

                #Administrador
                route add -host 132.248.2xx.xx gw $gatewayCliente dev $interface

        #Enrutamos las demas Ip's del Segmento a nuestra Interface 'tun0' de OpenVPN
        route add -net 132.248.2xx.0/24 gw $gw_vpnTun0 dev tun0

}


#Funcion: Agregar las Ip's del Segmento 132.248.1xx.xx8/2x a la Tabla de Ruteo del Cliente
function Ips_Rectoria {

        #Enrutamos la Ip del Firewall 'Hermes' por la Interface ' eth_ / wlan_ ' del Cliente
        route add -host $ip_Hermes gw $gatewayCliente dev $interface

        #Enrutamos las demas Ip's del Segmento a nuestra Interface 'tun0' de OpenVPN
        route add -net 132.248.1xx.xx8/2x gw $gw_vpnTun0 dev tun0

}


#Funcion: Agregar las Ip's del Segmento ' ' a la Tabla de Ruteo del Cliente ( Control Documental )
function Ips_ControlDocumental {

        #Enrutamos la Ip del Firewall '  '
        echo -e "En construccion ... \n "

}


#-----------------
# Ruteo de Ip's
#-----------------
#Ruteamos Ip's de acuerdo a la Red elejida por el usuario
function Agregar_rutas {

        #Comparamos el valor de la variable "num_red"
        case $num_red in

                #Red DGAE - Bunker
                1)
                        echo -e "Red DGAE : 132.248.2xx.0/24 \n "
                        echo -e "Agregando Ip's a la Tabla de Ruteo .... \n"

                        #Invocamos la Funcion 'Ips_Bunker'
                        Ips_Bunker

                        ;;

                #Red Rectoria
                2)
                        echo -e "Red Rectoria : 132.248.1xx.xx8/2x \n "
                        echo -e "Agregando Ip's a la Tabla de Ruteo .... \n"

                        #Invocamos la Funcion 'Ips_Rectoria'
                        Ips_Rectoria

                        ;;

                #Red Control Documental
                3)
```

```
                        echo -e "Red Control Documental : \n "
                        echo -e "Agregando Ip's a la Tabla de Ruteo .... \n"

                        #Invocamos la Funcion 'Ips_ControlDocumental'
                        Ips_ControlDocumental

                        ;;

                #Ninguna Opcion Elegida
                *)
                        echo -e "error : Red no Elejida \n "
                        ;;

        esac

}



#============================================
#       Funciones : Sistemas Mac OS X
#============================================

#---------------
# Datos de Red
#---------------
#Funcion: Obtenemos Ip y Gateway asignados a la Interface "utun0" por la Conexion OpenVPN establecida
function Obten_utun0_Mac {

        #Ip y Gateway
        ip_vpnUtun0=$(ifconfig utun0 | grep "inet" | awk -F " " '{print $2}')
        gw_vpnUtun0=$(ifconfig utun0 | grep "inet" | awk -F " " '{print $4}')

        #Mostramos Informacion acerca de la Interface "utun0"
        echo -e "-------------------------------------------"
        echo -e "  Ip Interface utun0      = $ip_vpnUtun0   "
        echo -e "  Gateway Interface utun0 = $gw_vpnUtun0   "
        echo -e "-------------------------------------------"
        echo -e "                                           "
}

#Funcion: Obtenemos Ip asiganada a la Interface " en* "
function Obten_ip_Mac {

        #Filtramos la ip del archivo "ip_$interface.txt"
        ip_equipo=$(grep "netmask" ip_$interface.txt | awk -F " " '{print $2}')

        #Imprimimos ip
        echo -e "Ip Equipo Cliente =      $ip_equipo \n"

}

#Funcion: Obtenemos Gateway
function Obten_gateway_Mac {

        #Obtenemos los tres primeros Octetos de la Ip del Equipo
        segmentoCliente=$(echo $ip_equipo | awk -F "." '{print $1"."$2"."$3}')

        #Filtramos el gateway del archivo "netstat_red.txt"
        gatewayCliente=$(grep "$interface" netstat_red.txt | awk -F " " '{print $1","$2}' | grep
"[0.0.0.0,default],$segmentoCliente" | awk -F "," '{print $2}')

        #Imprimimos gateway
        echo -e "Gateway Equipo Cliente = $gatewayCliente \n"

}


#-------
# Ip's
#-------
#Funcion: Agregar las Ip's del Segmento 132.248.2xx.0/24 a la Tabla de Ruteo del Cliente ( DGAE - Bunker )
function Ips_Bunker_Mac {

        #Enrutamos la Ip del Firewall 'Hades' por la Interface ' en_ ' del Cliente
        route add -host $ip_Hades $gatewayCliente

        #Enrutamos las Ip's que dan servicio en Internet
```

```
                #Servidores Web ( Firewall 'transparente' )
                route add -host 132.248.2xx.xx  $gatewayCliente           #secretarios.dgae.unam.mx
                route add -host 132.248.2xx.xx  $gatewayCliente           #odin.dgae.unam.mx
                route add -host 132.248.2xx.xx  $gatewayCliente    #circe.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente     #maxwell.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente     #escolar0.dgae.unam.mx

                #Servidores Web ( Firewall - Balanceador )
                route add -host 132.248.2xx.xx  $gatewayCliente     #uther.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente      #escolar1.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente      #lovelace.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente      #bessel.dgae.unam.mx

                #Servidores Web ( Firewall - DGOSE )
                route add -host 132.248.2xx.xx $gatewayCliente      #bayes.dgae.unam.mx

                #Firewalls ( Segmento de Red )
                #route add -host 132.248.2xx.xx  $gatewayCliente   #galois.dgae.unam.mx
                route add -host 132.248.2xx.xx  $gatewayCliente    #peano.dgae.unam.mx
                route add -host 132.248.2xx.xx  $gatewayCliente    #escolar-fw.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente     #peano2.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente     #hypatia.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente     #newton.dgae.unam.mx
                route add -host 132.248.2xx.xx $gatewayCliente     #andromeda.dgae.unam.mx
                #route add -host 132.248.2xx.xx $gatewayCliente           #zeus.dgae.unam.mx

                #Correo
                route add -host 132.248.2xx.xx $gatewayCliente     #bell.dgae.unam.mx

                #Administrador
                route add -host 132.248.2xx.xx $gatewayCliente

        #Enrutamos las demas Ip's del Segmento a nuestra Interface 'utun0' de OpenVPN
        route add -net 132.248.2xx.0/24 $gw_vpnUtun0

}


#Funcion: Agregar las Ip's del Segmento 132.248.1xx.xx8/2x a la Tabla de Ruteo del Cliente
function Ips_Rectoria_Mac {

        #Enrutamos la Ip del Firewall 'Hermes' por la Interface ' en_ ' del Cliente
        route add -host $ip_Hermes $gatewayCliente

        #Enrutamos las demas Ip's del Segmento a nuestra Interface 'utun0' de OpenVPN
        route add -net 132.248.1xx.xx8/2x $gw_vpnUtun0

}


#Funcion: Agregar las Ip's del Segmento ' ' a la Tabla de Ruteo del Cliente ( Control Documental )
function Ips_ControlDocumental_Mac {

        #Enrutamos la Ip del Firewall '  '
        echo -e "En construccion ... \n "

}


#-----------------
# Ruteo de Ip's
#-----------------
#Ruteamos Ip's de acuerdo a la Red elejida por el usuario
function Agregar_rutas_Mac {

        #Comparamos el valor de la variable "num_red"
        case $num_red in

                #Red DGAE - Bunker
                1)
                        echo -e "Red DGAE : 132.248.2xx.0/24 \n "
                        echo -e "Agregando Ip's a la Tabla de Ruteo .... \n"

                        #Invocamos la Funcion 'Ips_Bunker'
                        Ips_Bunker_Mac

                        ;;
```

```
                #Red Rectoria
                2)
                        echo -e "Red Rectoria : 132.248.1xx.xx8/2x \n "
                        echo -e "Agregando Ip's a la Tabla de Ruteo .... \n"

                        #Invocamos la Funcion 'Ips_Rectoria'
                        Ips_Rectoria_Mac

                        ;;

                #Red Control Documental
                3)
                        echo -e "Red Control Documental : \n "
                        echo -e "Agregando Ip's a la Tabla de Ruteo .... \n"

                        #Invocamos la Funcion 'Ips_ControlDocumental'
                        Ips_ControlDocumental_Mac

                        ;;

                #Ninguna Opcion Elegida
                *)
                        echo -e "error : Red no Elejida \n "
                        ;;

        esac

}




#================================
#       Cuerpo Principal
#================================

#Verificamos Opcion elegida por el cliente


#-------------------------------------------
#Comparamos el valor de la variable "opcion_r"
#-------------------------------------------
case $opcion_r in

        #-----------------------------------------
        #Conectarse a una Red UNAM mediante OpenVPN
        #-----------------------------------------
        a)
                #--------------------------------
                #       Red a Conectarse
                #--------------------------------
                #Solicitamos al usuario la Red a la Cual decea conectarse mediante OpenVPN
                echo -e "-------------------------------------------------"
                echo -e "Elija la Red a la que se esta conectando :        "
                echo -e "                                                  "
                echo -e "[1]  DGAE - Bunker        "
                echo -e "[2]  Rectoria             "
                echo -e "[3]  Control Documental "
                echo -e "                          "
                echo -e "Red a Conectarse : "
                          read num_red
                echo -e "-------------------------------------------------"


                #--------------------------------
                #       Sistema Operattivo
                #--------------------------------
                #Solicitamos al usuario el Sistema Operativo del Cual decea conectarse mediante OpenVPN
                echo -e "-----------------------------------------------------------"
                echo -e "Elija el Sistema Operativo que esta utilizando :  "
                echo -e "                                                          "
                echo -e "[l]  Linux                                                "
                echo -e "[m]  Mac OS X                                             "
                echo -e "                                                          "
                echo -e "Sistema Operativo : "
                          read sistema_operativo
                echo -e "-----------------------------------------------------------"
```

```
                #Comparamos el valor de la variable "sistema_operativo"
                case $sistema_operativo in

                        #------------------
                        #Sistemas GNU/Linux
                        #------------------
                        l)
                                #--------------------
                                # Informacion de Red
                                #--------------------
                                #Obtenemos las Interfaces de Red disponibles en el equipo
                                ifconfig | grep "[e,w][t,l][h,a]" | awk -F " " '{print $1}' >
num_interfaces.txt

                                #Obtenemos Informacion de Rutas
                                netstat -rn > netstat_red.txt

                                #--------------------------------------
                                #       Procesamos Informacion
                                #--------------------------------------
                                #Si el numero de interfaces es mayor a 1, solicitamos al cliente nos indique
cual Interface es la correcta
                                num_interface=$(wc -l num_interfaces.txt | awk -F " " '{print $1}')

                                #Interfaces de Red
                                if [ $num_interface -eq 1 ]; then

                                        #Existe solo una interface
                                        interface=$(cat num_interfaces.txt)

                                        #Guardamos datos
                                        ifconfig $interface > ip_$interface.txt

                                        #Invocamos la Funcion 'Obten_tun0'
                                        Obten_tun0

                                        #Invocamos la Funcion 'Obten_ip'
                                        Obten_ip

                                        #Invocamos la Funcion 'Obten_gateway'
                                        Obten_gateway

                                        #Invocamos la Funcion 'Agregar_rutas'
                                        Agregar_rutas

                                else

                                        #Existe mas de una interface
                                        echo -e "                                               "
                                        echo -e "Existe mas de una Interface de Red :      "
                                        echo -e "-------                                      "
                                                cat num_interfaces.txt
                                        echo -e "-------                                      "
                                        echo -e "                                            "

                                        #Solicitamos la Interface sobre la cual trabajaremos
                                        echo -e "Introduce la Interface conectada a Red :  "
                                        read interface
                                        echo -e "                                                        "

                                        #Guardamos Datos
                                        ifconfig $interface > ip_$interface.txt

                                        #Invocamos la Funcion 'Obten_tun0'
                                        Obten_tun0

                                        #Invocamos la Funcion 'Obten_ip'
                                        Obten_ip

                                        #Invocamos la Funcion 'Obten_gateway'
                                        Obten_gateway

                                        #Invocamos la Funcion 'Agregar_rutas'
                                        Agregar_rutas

                                fi

                                ;; #Fin de Eleccion GNU/Linux
```

```
                        #-----------------
                        #Sistemas Mac OS X
                        #-----------------
                        m)
                                #--------------------
                                # Informacion de Red
                                #--------------------
                                #Obtenemos las Interfaces de Red disponibles en el equipo
                                ifconfig | grep "[e][n][0,1][:]" | awk -F " " '{print $1}' | awk -F ":" '{print
$1}' > num_interfaces.txt

                                #Obtenemos Informacion de Rutas
                                netstat -rn > netstat_red.txt

                                #-------------------------------------
                                #       Procesamos Informacion
                                #-------------------------------------
                                #Si el numero de interfaces es mayor a 1, solicitamos al cliente nos indique
cual Interface es la correcta
                                num_interface=$(wc -l num_interfaces.txt | awk -F " " '{print $1}')

                                #Interfaces de Red
                                if [ $num_interface -eq 1 ]; then

                                        #Existe solo una interface
                                        interface=$(cat num_interfaces.txt)

                                        #Guardamos datos
                                        ifconfig $interface > ip_$interface.txt

                                        #Invocamos la Funcion 'Obten_utun0_Mac'
                                        Obten_utun0_Mac

                                        #Invocamos la Funcion 'Obten_ip_Mac'
                                        Obten_ip_Mac

                                        #Invocamos la Funcion 'Obten_gateway_Mac'
                                        Obten_gateway_Mac

                                        #Invocamos la Funcion 'Agregar_rutas_Mac'
                                        Agregar_rutas_Mac

                                else

                                        #Existe mas de una interface
                                        echo -e "                                               "
                                        echo -e "Existe mas de una Interface de Red :      "
                                        echo -e "-------                                      "
                                                cat num_interfaces.txt
                                        echo -e "-------                                      "
                                        echo -e "                                            "

                                        #Solicitamos la Interface sobre la cual trabajaremos
                                        echo -e "Introduce la Interface conectada a Red :  "
                                        read interface
                                        echo -e "                                                        "

                                        #Guardamos Datos
                                        ifconfig $interface > ip_$interface.txt

                                        #Invocamos la Funcion 'Obten_utun0_Mac'
                                        Obten_utun0_Mac

                                        #Invocamos la Funcion 'Obten_ip_Mac'
                                        Obten_ip_Mac

                                        #Invocamos la Funcion 'Obten_gateway_Mac'
                                        Obten_gateway_Mac

                                        #Invocamos la Funcion 'Agregar_rutas_Mac'
                                        Agregar_rutas_Mac

                                fi

                                ;; #Fin de Eleccion Mac OS X

                        #Ninguna Opcion Elegida
                        *)
```

```
                        echo -e "error : Sistema Operativo no Elejido \n "
                        ;;

        esac #Fin de Eleccion de Sistemas Operativos

;; #---> Fin de Opcion 'a'


#-------------------------------------------------------------
#Eliminar IP's de la Tabla de Rutero anteriormente agregadas
#-------------------------------------------------------------
b)
        #---------------------------------
        #       Red a Desconectarse
        #---------------------------------
        #Solicitamos al usuario la Red de la Cual decea eliminar las Ip's agregadas a la Tabla de Ruteo
        echo -e "-------------------------------------------------------------"
        echo -e "Elija la Red a eliminar Ip's dentro de la Tabla de Ruteo :         "
        echo -e "                                                                   "
        echo -e "[1]  DGAE - Bunker               "
        echo -e "[2]  Rectoria                    "
        echo -e "[3]  Control Documental  "
        echo -e "                                 "
        echo -e "Red a Desconectarse :            "
                 read num_red_elim
        echo -e "-------------------------------------------------"


        #--------------------------------------
        #       Procesamos Informacion
        #--------------------------------------
        #Comparamos el valor de la variable "num_red_elim"
        case $num_red_elim in

                #Eliminar Ip's del Segmento de Red "DGAE-Bunker" de la Tabla de Ruteo
                1)
                        echo -e "Red DGAE : 132.248.2xx.0/24 \n "
                        echo -e "Eliminando Ip's de la Tabla de Ruteo .... \n"

                        #Ip del Firewall 'Hades' por la Interface ' eth_ / wlan_ ' del Cliente
                        route delete -host $ip_Hades

                        #Ip's que dan servicio en Internet

                                #Servidores Web ( Firewall 'transparente' )
                                route delete -host 132.248.2xx.xx        #secretarios.dgae.unam.mx
                                route delete -host 132.248.2xx.xx        #odin.dgae.unam.mx
                                route delete -host 132.248.2xx.xx        #circe.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #maxwell.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #escolar0.dgae.unam.mx

                                #Servidores Web ( Firewall - Balanceador )
                                route delete -host 132.248.2xx.xx        #uther.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #escolar1.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #lovelace.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #bessel.dgae.unam.mx

                                #Servidores Web ( Firewall - DGOSE )
                                route delete -host 132.248.2xx.xx       #bayes.dgae.unam.mx

                                #Firewalls ( Segmento de Red )
                                #route delete -host 132.248.2xx.xx        #galois.dgae.unam.mx
                                route delete -host 132.248.2xx.xx        #peano.dgae.unam.mx
                                route delete -host 132.248.2xx.xx        #escolar-fw.unam.mx
                                route delete -host 132.248.2xx.xx       #peano2.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #hypatia.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #newton.dgae.unam.mx
                                route delete -host 132.248.2xx.xx       #andromeda.dgae.unam.mx
                                #route delete -host 132.248.2xx.xx       #zeus.dgae.unam.mx

                                #Correo
                                route delete -host 132.248.2xx.xx       #bell.dgae.unam.mx

                                #Administrador
                                route delete -host 132.248.2xx.xx

                ;;
```

```
                        #Eliminar Ip's del Segmento de Red "Rectoria" de la Tabla de Ruteo
                        2)
                                echo -e "Red Rectoria : 132.248.1xx.xx8/2x \n "
                                echo -e "Eliminando Ip's de la Tabla de Ruteo .... \n"

                                #Ip del Firewall 'Hermes' por la Interface ' eth_ / wlan_ ' del Cliente (
Cambiar IP )
                                route delete -host $ip_Hermes

                        ;;


                        #Eliminar Ip's del Segmento de Red "Control Documental" de la Tabla de Ruteo
                        3)
                                echo -e "Red Control Documental : \n "
                                echo -e "Elminando Ip's de la Tabla de Ruteo .... \n"

                                echo -e "Red en Construccion ... "

                        ;;


                        #Ninguna Opcion Elegida
                        *)
                                echo -e "error : Red no Elejida \n "
                        ;;


                esac # ---> Fin de Segundo Case


        ;; # ---> Finde de Opcion 'b'


        #Ninguna Opcion Elegida
        *)
                echo -e "\nerror : Opcion No Valida !!! \n "
        ;;


esac # ---> Fin de Primer Case
```

Script Windows

El siguiente script es utilizado para modificar la tabla de ruteo del cliente conectado a la VPN.

```
::#############################################################################################
::                                  gaara
::                        ( 25 - Mayo - 2015 )
::
:: Se agregan IP's a la tabla de ruteo del cliente conectado a las VPN's
:: Las IP's agregadas perteneceran al segmenteo en donde el Firewall con OpenVPN Resida.
:: Al ser IP's publicas, haremos que el equipo cliente las visualize por la Interface de
::      red creada por OpenVPN en el equipo cliente ( tun0 ).
:: Equipos con IP publica que proporcionen servicio en Internet se dejara que el equipo cliente
::               los busque por la Interface de Red de default ( eth0 - en la mayoria de los casos ).
::
::#############################################################################################

::====================================
::      Informacion Global
::====================================

::Eliminamos repeticion de la ruta donde nos encontramos
@echo off

::Invocamos nuestra Funcion Principal
call:main



::================================
::        Funciones
::      Agregando Ip's
::================================

::------------------------------
::      Funcion: Red a Conectarse
::------------------------------
:Red_Conectarse

        ::Solicitamos al usuario la opcion a realizar
        echo ================================================================
        echo Elija la Red a la que se esta conectando :


        echo [1]  DGAE - Bunker
        echo [2]  Rectoria
        echo [3]  Control Documental
        echo.
        echo Opcion :

                set /P num_red=

        :: Imprimimos Opcion Seleccionada
        echo.
        echo     Opcion elegida por el usuario = [ %num_red% ]
        echo.
        echo ================================================================

        ::Comparamos el Valor de 'num_red'

                ::Red DGAE - Bunker
                IF %num_red% == 1 (
                        ::Invocamos la Funcion 'Ips_Bunker'
                        call:Ips_Bunker
                )

                ::Red Rectoria
```

```
                IF %num_red% == 2 (
                        ::Invocamos la Funcion 'Ips_Rectoria'
                        call:Ips_Rectoria
                )

                ::Red Control Documental
                IF %num_red% == 3 (
                        ::Invocamos la Funcion 'Ips_ControlDocumental'
                        call:Ips_ControlDocumental
                )

:: Fin de Funcion --> 'Red_conectarse'
goto:EOF


::--------------------------
::      Funcion : Ips_Bunker
::--------------------------
::Agregar las Ip's del Segmento 132.248.2xx.0/24 a la Tabla de Ruteo del Cliente ( DGAE - Bunker )
:Ips_Bunker

        echo.
        echo Red DGAE : 132.248.2xx.0/24

        ::-----------
        :: Variables
        ::-----------
        ::Ip OpenVPN Firewall 'Hades' ( DGAE - Bunker )
        set ip_Hades=132.248.2xx.x1

        ::Imprimimos Valores
        echo.
        echo Firewall Hades : %ip_Hades%
        echo.

        ::--------------------
        :: Informacion de Red
        ::--------------------
        ::Solicitamos al asuario la IP asignada por OpenVPN
        echo ================================================================
        echo.
        echo Introduce la IP asignada por la Conexion OpenVPN :
        echo.
                set /P ip_vpnTun0=

        ::Hacemos un 'ping' a la Ip asignada por OpenVPN
        ping -n 1 %ip_vpnTun0% > ping_vpnTun0.txt

        ::Verificamos si el 'ping' fue exitoso, buscando cadenas donde el 'ping' no fue exitoso
        findstr "out agotado unreachable Please check encontrar Compruebe" ping_vpnTun0.txt > null

        IF %ERRORLEVEL% == 1 (
                ::Imprimimos Ip Asignada por la Conexion OpenVPN
                echo.
                echo    Ip Asignada por OpenVPN = { %ip_vpnTun0% }
                echo.
        )

        IF %ERRORLEVEL% == 0 (
                echo.
                echo    Error : La IP proporcionada no responde Solicitudes "ping"
                echo.
        )
        echo.
        echo ================================================================

        ::Solicitamos al usuario la Puerta de Enlace Principal 'gateway' de su conexion a Red
        echo ================================================================
        echo.
        echo Introduce la Ip de tu Puerta de Enlace Principal "gateway" :
        echo.
                        set /P gatewayCliente=

        ::Hacemos un 'ping' a la Puerta de Enlace Principal "gateway"
        ping -n 1 %gatewayCliente% > ping_gatewayCliente.txt

        ::Verificamos si el 'ping' fue exitoso, buscando cadenas donde el 'ping' no fue exitoso
        findstr "out agotado unreachable Please check encontrar Compruebe" ping_gatewayCliente.txt > null
```

```
        IF %ERRORLEVEL% == 1 (
        ::Imprimimos Ip del gateway
        echo.
        echo     Puerta de Enlace "gateway" = { %gatewayCliente% }
        echo.
        )

        IF %ERRORLEVEL% == 0 (
                echo.
                echo    Error : Puerta de Enlace "gateway" proporcionada no responde Solicitudes "ping"
                echo.
        )
        echo.
        echo ================================================================

        ::Solicitamos al usuario la Ip de la Puerta de Enlace 'gateway' de su conexion OpenVPN
        echo ================================================================
        echo.
        echo Introduce nevamente la IP { %ip_vpnTun0% } sumando un '1' al Cuarto Octeto :
        echo.
                set /P gw_vpnTun0=
        echo.
        echo     Ip de la Puerta de Enlace con tu conexion OpenVPN = %gw_vpnTun0%
        echo.
        echo ================================================================

        echo.
        echo Agregando Ip's a la Tabla de Ruteo ....
        echo.

        ::Invocamos la Funcion 'add_Ips_Bunker', la cual agregara las Ips a la Tabla de Ruteo del cliente
        call:add_Ips_Bunker

::Fin de Funcion --> 'Ips_Bunker'
goto:EOF


::---------------------------
::      Funcion : Ips_Rectoria
::---------------------------
::Agregar las Ip's del Segmento 132.248.1xx.xx8/2x a la Tabla de Ruteo del Cliente
:Ips_Rectoria

        echo.
        echo Red Rectoria :

        ::-----------
        :: Variables
        ::-----------
        ::Ip OpenVPN Firewall 'Hermes'
        set ip_Hermes=132.248.1xx.xx1

        ::Imprimimos Valores
        echo.
        echo Firewall Hermes : %ip_Hermes%
        echo.

        ::--------------------
        :: Informacion de Red
        ::--------------------
        ::Solicitamos al asuario la IP asignada por OpenVPN
        echo ================================================================
        echo.
        echo Introduce la IP asignada por la Conexion OpenVPN :
        echo.
                set /P ip_vpnTun0=

        ::Hacemos un 'ping' a la Ip asignada por OpenVPN
        ping -n 1 %ip_vpnTun0% > ping_vpnTun0.txt

        ::Verificamos si el 'ping' fue exitoso, buscando cadenas donde el 'ping' no fue exitoso
        findstr "out agotado unreachable Please check encontrar Compruebe" ping_vpnTun0.txt > null

        IF %ERRORLEVEL% == 1 (
                ::Imprimimos Ip Asignada por la Conexion OpenVPN
                echo.
                echo     Ip Asignada por OpenVPN = { %ip_vpnTun0% }
                echo.
        )
```

```
        IF %ERRORLEVEL% == 0 (
                echo.
                echo    Error : La IP proporcionada no responde Solicitudes "ping"
                echo.
        )
        echo.
        echo ================================================================

        ::Solicitamos al usuario la Puerta de Enlace Principal 'gateway' de su conexion a Red
        echo ================================================================
        echo.
        echo Introduce la Ip de tu Puerta de Enlace Principal "gateway" :
        echo.
                        set /P gatewayCliente=

        ::Hacemos un 'ping' a la Puerta de Enlace Principal "gateway"
        ping -n 1 %gatewayCliente% > ping_gatewayCliente.txt

        ::Verificamos si el 'ping' fue exitoso, buscando cadenas donde el 'ping' no fue exitoso
        findstr "out agotado unreachable Please check encontrar Compruebe" ping_gatewayCliente.txt > null

        IF %ERRORLEVEL% == 1 (
        ::Imprimimos Ip del gateway
        echo.
        echo    Puerta de Enlace "gateway" = { %gatewayCliente% }
        echo.
        )

        IF %ERRORLEVEL% == 0 (
                echo.
                echo    Error : Puerta de Enlace "gateway" proporcionada no responde Solicitudes "ping"
                echo.
        )
        echo.
        echo ================================================================

        ::Solicitamos al usuario la Ip de la Puerta de Enlace 'gateway' de su conexion OpenVPN
        echo ================================================================
        echo.
        echo Introduce nevamente la IP { %ip_vpnTun0% } sumando un '1' al Cuarto Octeto :
        echo.
                set /P gw_vpnTun0=
        echo.
        echo    Ip de la Puerta de Enlace con tu conexion OpenVPN = %gw_vpnTun0%
        echo.
        echo ================================================================

        echo.
        echo Agregando Ip's a la Tabla de Ruteo ....
        echo.

        ::Invocamos la Funcion 'add_Ips_Rectoria', la cual agregara las Ips a la Tabla de Ruteo del cliente
        call:add_Ips_Rectoria

::Fin de Funcion --> 'Ips_Rectoria
goto:EOF


::-----------------------------------
::      Funcion : Ips_ControlDocumental
::-----------------------------------
::Agregar las Ip's del Segmento ' ' a la Tabla de Ruteo del Cliente ( Control Documental )
:Ips_ControlDocumental

        echo Red Control Documental
        echo En construccion ...

        ::-----------
        :: Variables
        ::-----------
        ::Ip OpenVPN Firewall 'Iris' ( Control Documental -> Cambiar Ip y Hostname )
        set ip_Iris=132.248.xx.xx

        ::Imprimimos Valores
        echo.
        echo Firewall Iris : %ip_Iris%
        echo.
```

```
        ::--------------------
        :: Informacion de Red
        ::--------------------
        ::Solicitamos al asuario la IP asignada por OpenVPN
        echo ================================================================
        echo.
        echo Introduce la IP asignada por la Conexion OpenVPN :
        echo.
                set /P ip_vpnTun0=

        ::Hacemos un 'ping' a la Ip asignada por OpenVPN
        ping -n 1 %ip_vpnTun0% > ping_vpnTun0.txt

        ::Verificamos si el 'ping' fue exitoso, buscando cadenas donde el 'ping' no fue exitoso
        findstr "out agotado unreachable Please check encontrar Compruebe" ping_vpnTun0.txt > null

        IF %ERRORLEVEL% == 1 (
                ::Imprimimos Ip Asignada por la Conexion OpenVPN
                echo.
                echo    Ip Asignada por OpenVPN = { %ip_vpnTun0% }
                echo.
        )

        IF %ERRORLEVEL% == 0 (
                echo.
                echo    Error : La IP proporcionada no responde Solicitudes "ping"
                echo.
        )
        echo.
        echo ================================================================

        ::Solicitamos al usuario la Puerta de Enlace Principal 'gateway' de su conexion a Red
        echo ================================================================
        echo.
        echo Introduce la Ip de tu Puerta de Enlace Principal "gateway" :
        echo.
                        set /P gatewayCliente=

        ::Hacemos un 'ping' a la Puerta de Enlace Principal "gateway"
        ping -n 1 %gatewayCliente% > ping_gatewayCliente.txt

        ::Verificamos si el 'ping' fue exitoso, buscando cadenas donde el 'ping' no fue exitoso
        findstr "out agotado unreachable Please check encontrar Compruebe" ping_gatewayCliente.txt > null

        IF %ERRORLEVEL% == 1 (
        ::Imprimimos Ip del gateway
        echo.
        echo    Puerta de Enlace "gateway" = { %gatewayCliente% }
        echo.
        )

        IF %ERRORLEVEL% == 0 (
                echo.
                echo    Error : Puerta de Enlace "gateway" proporcionada no responde Solicitudes "ping"
                echo.
        )
        echo.
        echo ================================================================

        ::Solicitamos al usuario la Ip de la Puerta de Enlace 'gateway' de su conexion OpenVPN
        echo ================================================================
        echo.
        echo Introduce nevamente la IP { %ip_vpnTun0% } sumando un '1' al Cuarto Octeto :
        echo.
                set /P gw_vpnTun0=
        echo.
        echo    Ip de la Puerta de Enlace con tu conexion OpenVPN = %gw_vpnTun0%
        echo.
        echo ================================================================

        echo.
        echo Agregando Ip's a la Tabla de Ruteo ....
        echo.

        ::Invocamos la Funcion 'add_Ips_ControlDocumental', la cual agregara las Ips a la Tabla de Ruteo del
cliente
        call:add_Ips_ControlDocumental

::Fin de Funcion --> 'Ips_ControlDocumental'
```

```
goto:EOF


::-----------------------------------------------
::               Funcion : add_Ips_Bunker
::        Agregamos Ips de DGAE - Bunker
::-----------------------------------------------
:add_Ips_Bunker

        ::Enrutamos la Ip del Firewall 'Hades' por el Gateway ( Principal )del Cliente
        route add %ip_Hades% %gatewayCliente%

        ::Enrutamos las Ip's que dan servicio en Internet

                ::Servidores Web ( Firewall 'transparente' )
                        rem secretarios.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem odin.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem circe.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem maxwell.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem escolar0.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%

                ::Servidores Web ( Firewall - Balanceador )
                        rem uther.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem escolar1.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem lovelace.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem bessel.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%

                ::Servidores Web ( Firewall - DGOSE )
                        rem bayes.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%

                ::Firewalls ( Segmento de Red )
                        rem galois.dgae.unam.mx
                        ::route add 132.248.2xx.xx        %gatewayCliente%
                        rem peano.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem escolar-fw.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem peano2.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem hypatia.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem newton.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem andromeda.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%
                        rem zeus.dgae.unam.mx
                        ::route add 132.248.2xx.xx        %gatewayCliente%

                ::Correo
                        rem bell.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%

                ::Administrador
                        rem semblante.dgae.unam.mx
                        route add 132.248.2xx.xx %gatewayCliente%

        ::Enrutamos las demas Ip's del Segmento por el Gateway de la Conexion OpenVPN
        route add 132.248.2xx.0/24 %gw_vpnTun0%

        echo.
        echo Ip's Agregadas a la Tabla de Ruteo ...
        echo.

        pause
        exit

::Fin de Funcion --> 'add_Ips_Bunker'
goto:EOF
```

```
::-----------------------------------------
::       Funcion : add_Ips_Rectoria
::        Agregamos Ips de Rectoria
::-----------------------------------------
:add_Ips_Rectoria

        ::Enrutamos la Ip del Firewall 'Hermes' por el Gateway ( Principal )' del Cliente
        route add %ip_Hermes% %gatewayCliente%

        ::Enrutamos las demas Ip's del Segmento por el Gateway de la Conexion OpenVPN
        route add 132.248.1xx.xx8/2x %gw_vpnTun0%

        echo.
        echo Ip's Agregadas a la Tabla de Ruteo ...
        echo.

        pause
        exit

::Fin de Funcion --> 'add_Ips_Rectoria'
goto:EOF


::-----------------------------------------
::       Funcion : add_Ips_ControlDocumental
::       Agregamos Ips de Control Documental
::-----------------------------------------
:add_Ips_ControlDocumental

        ::Enrutamos la Ip del Firewall 'Iris'
        echo.
        echo En construccion ...
        echo.

        pause
        exit

::Fin de Funcion --> 'add_Ips_ControlDocumental'
goto:EOF



::================================
::          Funciones
::       Eliminando Ip's
::================================

::---------------------------------
::       Funcion : Red a Desconectarse
::---------------------------------
:Red_Desconectarse

        ::Solicitamos al usuario la opcion a realizar
        echo ====================================================================
        echo Elija la Red a eliminar Ip's dentro de la Tabla de Ruteo :


        echo [1]  DGAE - Bunker
        echo [2]  Rectoria
        echo [3]  Control Documental
        echo.
        echo Opcion :

                set /P num_red_elim=

        :: Imprimimos Opcion Seleccionada
        echo.
        echo    Opcion elegida por el usuario es = [ %num_red_elim% ]
        echo.
        echo ====================================================================

        ::Comparamos el Valor de 'num_red_elim'
        IF %num_red_elim% == 1 (
                ::Invocamos la Funcion 'Elimina_Ips_Bunker'
                call:Elimina_Ips_Bunker
        )
```

```
        IF %num_red_elim% == 2 (
                ::Invocamos la Funcion 'Elimina_Ips_Rectoria'
                call:Elimina_Ips_Rectoria
        )

        IF %num_red_elim% == 3 (
                ::Invocamos la Funcion 'Elimina_Ips_ControlDocumental'
                call:Elimina_Ips_ControlDocumental
        )

:: Fin de Funcion --> 'Red_Desconectarse'
goto:EOF


::--------------------------------
::      Funcion: Elimina_Ips_Bunker
::--------------------------------
::Eliminar Ip's del Segmento de Red "DGAE-Bunker" de la Tabla de Ruteo
:Elimina_Ips_Bunker

        echo Red DGAE : 132.248.2xx.0/24
        echo Eliminando Ip's de la Tabla de Ruteo ....

        ::-----------
        :: Variables
        ::-----------
        ::Ip OpenVPN Firewall 'Hades' ( DGAE - Bunker )
        set ip_Hades=132.248.2xx.x1

        ::Ip del Firewall 'Hades'
        route delete %ip_Hades%

        ::Ip's que dan servicio en Internet
                ::Servidores Web ( Firewall 'transparente' )
                        rem secretarios.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem odin.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem circe.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem maxwell.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem escolar0.dgae.unam.mx
                        route delete 132.248.2xx.xx

                ::Servidores Web ( Firewall - Balanceador )
                        rem uther.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem escolar1.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem lovelace.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem bessel.dgae.unam.mx
                        route delete 132.248.2xx.xx

                ::Servidores Web ( Firewall - DGOSE )
                        rem bayes.dgae.unam.mx
                        route delete 132.248.2xx.xx

                ::Firewalls ( Segmento de Red )
                        rem galois.dgae.unam.mx
                        ::route delete 132.248.2xx.xx
                        rem peano.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem escolar-fw.unam.mx
                        route delete 132.248.2xx.xx
                        rem peano2.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem hypatia.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem newton.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem andromeda.dgae.unam.mx
                        route delete 132.248.2xx.xx
                        rem zeus.dgae.unam.mx
                        ::route delete 132.248.2xx.xx

                ::Correo
                        rem bell.dgae.unam.mx
```

```
                        route delete 132.248.2xx.xx

                ::Administrador
                        rem semblante.dgae.unam.mx
                        route delete 132.248.2xx.xx

        ::Ip's del Segmento agregadas al Gateway de la Conexion OpenVPN
        route delete 132.248.2xx.0/24

        pause
        exit

::Fin de Funcion --> 'Elimina_Ips_Bunker'
goto:EOF


::---------------------------------
::      Funcion: Elimina_Ips_Rectoria
::---------------------------------
::Eliminar Ip's del Segmento de Red "Rectoria" de la Tabla de Ruteo
:Elimina_Ips_Rectoria

        echo Red Rectoria :
        echo Eliminando Ip's de la Tabla de Ruteo ....

        ::-----------
        :: Variables
        ::-----------
        ::Ip OpenVPN Firewall 'Hermes'
        set ip_Hermes=132.248.1xx.xx1

        ::Ip del Firewall 'Hermes'
        route delete %ip_Hermes%

        ::Ip's del Segmento agregadas al Gateway de la Conexion OpenVPN
        route delete 132.248.1xx.xx8/2x

        pause
        exit

::Fin de Funcion --> 'Elimina_Ips_Rectoria'
goto:EOF


::-----------------------------------------
::      Funcion: Elimina_Ips_ControlDocumental
::-----------------------------------------
::Eliminar Ip's del Segmento de Red "Control Documental" de la Tabla de Ruteo
:Elimina_Ips_ControlDocumental

        echo Red Control Documental :
        echo Elminando Ip's de la Tabla de Ruteo ....
        echo Red en Construccion ...

        ::-----------
        :: Variables
        ::-----------
        ::Ip OpenVPN Firewall 'Iris' ( Control Documental -> Cambiar Ip y Hostname )
        set ip_Iris=132.248.xx.xx

        pause
        exit

::Fin de Funcion --> 'Elimina_Ips_ControlDocumental'
goto:EOF



::=========================
::      Funcion Principal
::=========================
:main
        ::Solicitamos al usuario la opcion a realizar
        echo ================================================================
        echo Elija la Opcion a realizar :


        echo [a]  Agregar IP's a la Tabla de Ruteo
```

```
        echo [b]  Eliminar IP's de la Tabla de Ruteo anteriormente agregadas
        echo.
        echo Opcion :

                set /P opcion_r=

        :: Imprimimos Opcion Seleccionada
        echo.
        echo    Opcion elegida por el usuario = [ %opcion_r% ]
        echo.
        echo ================================================================

        ::Comparamos el Valor de 'opcion_r'
        IF %opcion_r% == a (
                call:Red_Conectarse
        )

        IF %opcion_r% == b (
                call:Red_Desconectarse
        )

:: Fin de Funcion --> 'main'
goto:EOF


::Parametros Globales
exit
```