

UNIVERSIDAD PERUANA UNIÓN
FACULTAD DE INGENIERÍA Y ARQUITECTURA
E.A.P INGENIERÍA DE SISTEMAS



Una Institución Adventista

PROYECTO DE INVESTIGACIÓN

Plan de mejora del ancho de banda de internet y seguridad aplicados a la red de datos basado en la metodología del diseño descendente de redes top down en la Universidad Nacional de San Martín

Tesis Presentada para optar el título de Ingeniero de Sistemas

Autor

Bach. Erick José Carrasco Guerrero

Asesor

Ing. Flor Elizabeth Cerdán León

Fecha

Morales, Noviembre 2013

FICHA CATALOGRÁFICA

CARRASCO GUERRERO, Erick. Plan de mejora del ancho de banda de internet y seguridad aplicados a la red de datos basado en la metodología del diseño descendente de redes top down en la Universidad Nacional de San Martín. Tesis (TESIS PARA OPTAR EL TÍTULO DE INGENIERO DE SISTEMAS Morales, San Martin: Universidad Peruana Unión, Facultad de Ingeniería y Arquitectura.2013. p.:21cm x29.7 cm.

Asesor: Flor Elizabeth Cerdán León
Ing.

Dedicatoria

A Dios, por ser fuente de sabiduría.

A mis padres, por su apoyo incondicional que me motivan en el desarrollo de la investigación.

Agradecimientos

A Dios, por darme la fuerza e inteligencia y por sus múltiples bendiciones.

A José Carrasco y Sabina Guerrero quienes no miden esfuerzos, tiempo y dinero y en quienes siempre puedo confiar.

A mi asesor Ing Flor Cerdán Leon, por sus importantes recomendaciones para el desarrollo del presente trabajo de investigación.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS.....	viii
ÍNDICE DE ANEXOS.....	ix
LISTA DE ACRÓNIMOS	x
RESUMEN	xi
ABSTRACT.....	xiii
CAPÍTULO I. INTRODUCCIÓN	1
CAPÍTULO II. MARCO TEÓRICO	10
2.1 Introducción.....	10
2.2 Sobre la Universidad Nacional de San Martín.....	10
2.3 Sobre la oficina de administración de red óptica.....	12
2.4 Sobre la infraestructura tecnológica y la red de datos.	13
2.5 Linux.....	15
2.6 Squid.....	20
2.7 IPTables.....	34
2.8 Webmin.....	43
2.9 Estándares.....	44
2.9.1 Linux standard base.....	44
2.9.2 Estándares de protocolos de Internet RFC.....	46
2.9.3 Principales protocolos de internet	49
2.9.4 Estándares de internet IETF	51
2.10 Metodología de diseño de red top down.....	53
2.10.1 Historia	53
2.10.2 Fase I: Identificar objetivos y necesidades del cliente.....	54
2.10.3 Fase II: Diseño de una red lógica.....	61
2.10.4 Fase III: Diseño de la red física	66
2.10.5 Fase IV: Testeo, optimización y documentación de la red.....	66
2.11 Casos de éxito.....	68
CAPÍTULO III. METODOLOGÍA DE LA INVESTIGACIÓN	70
3.1 Introducción.....	70
3.2 Lugar de aplicación	70
3.3 Tipo de investigación	70
3.4 Metodología de la investigación	71

3.4.1	Descripción del Problema.....	71
3.4.2	Análisis de los objetivos del negocio.....	71
3.4.3	Análisis de las limitaciones técnicas	72
3.4.4	Requerimientos identificados y documentados	72
3.4.5	Diseño de Políticas de Seguridad para la red.....	72
3.4.6	Creación de controles para el ingreso y salida de la red	72
3.4.7	Selección de la tecnología más adecuada.....	72
3.4.8	Implementación de la solución técnica	72
3.4.9	Implantación de la solución técnica.....	72
3.4.10	Pruebas.....	72
3.4.11	Optimización.....	72
3.4.12	Documentación	73
CAPÍTULO IV. DESARROLLO DE LA SOLUCIÓN		74
4.1	Introducción	74
4.2	Análisis de requerimientos.	74
4.2.1	Acceso a internet.....	74
4.2.2	Red inalámbrica.....	74
4.2.3	Compartición de recursos.....	75
4.2.4	Control del servicio de acceso a internet.	75
4.2.5	Servidor de configuración dinámica de host.....	76
4.2.6	Servidor FTP.....	76
4.2.7	Seguridad.....	77
4.2.8	Disponibilidad	77
4.2.9	Escalabilidad	77
4.2.10	Performance	78
4.2.11	Seguridad.....	79
4.3	Diseño lógico.....	80
4.4	Diseño físico.....	82
4.5	Testeo, optimización y documentación de la red	90
4.5.1	Servidor Proxy.....	90
4.5.2	Corta fuego.....	101
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES.....		103
REFERENCIAS		104

ANEXOS.....	105
-------------	-----

ÍNDICE DE TABLAS

Tabla 1. Reporte de uso de acceso a Internet	1
Tabla 2. Cantidad de Alumnos Matriculados por Facultad y Escuela - Ciclo 2013-I ..	2
Tabla 3. Personal Docente que labora en la UNSM-T.....	3
Tabla 4. Tráfico generado por el uso del servicio de internet.	4
Tabla 5. Lista de servidores disponibles en la red de datos.	5
Tabla 6.- Tabla carga de trabajo squid.....	30
Tabla 7.-Estándares de protocolos oficiales de internet.....	50
Tabla 8.-Distribución De Equipos De Cómputo UNSM-T	83

ÍNDICE DE FIGURAS

Figura 1 Organigrama de oficina de informática y comunicaciones	13
Figura 2.- Centos Terminal	44
Figura 3.- Diseño de la metodología de la investigación	71
Figura 4.-Diagrama de la red de datos de la UNSM-T	81
Figura 5.- Diagrama de Canalización de la UNSM-T	84
Figura 6.- Nodo Central	84
Figura 7. Nodo central - switch core	85
Figura 8.- Nodo Secundario - Instituto Materno Infantil	85
Figura 9.- Nodo Secundario switch de borde - Instituto Materno Infantil.....	86
Figura 10.-Nodo Secundario - Ciencias Contables	86
Figura 11.- Nodo Secundario switch de borde - ingeniería Civil.....	87
Figura 12.-Nodo Secundario detalle de codificación de Ingeniería Civil	87
Figura 13.- Nodo Secundario switch de borde - Ciencias Agrarias.....	88
Figura 14.-Nodo Secundario detalle de codificación Ciencias Agrarias	88
Figura 15.- Nodo Secundario switch de borde - Agroindustrial	89
Figura 16.- Nodo Secundario switch de borde -Agroindustrial- Laboratorios	89
Figura 17.-Gestor administrador del webmin	90
Figura 18.-Opciones de administración del squid.....	90
Figura 19.-_Configuración de puertos de entrada y salida.....	91
Figura 20.-Ingreso direcciones del cache	91
Figura 21.- Cantidad de memoria y porcentaje del mismo	92
Figura 22.- Almacenar datos en el servidor	92
Figura 23.- Configurar la dirección del cache.....	93
Figura 24.-Opción: Helper Programs.....	93
Figura 25.-Opción: Access Control	94
Figura 26.-IP de las máquinas de los alumnos conectado	94
Figura 27.-IP de los Administrativos dentro del servidor	95
Figura 28.-Direcciones denegadas por el servidor proxy	95
Figura 29.-Opción: Administrative Options	96
Figura 30.-Opción: authentication programs	96
Figura 31.- Squid Report Generator	97
Figura 32.-Squid Report Generator	97
Figura 33.-Lista con todos los reportes generados	98
Figura 34.-Direcciones IP que se encuentran conectadas a la red.....	98
Figura 35.-Intento de ingresar a páginas denegadas.....	99
Figura 36.- Reportes generados	99
Figura 37.- Lista de reportes efectuados hasta las fecha	100
Figura 38.- Direcciones a las cuales se decidió ingresar.	100
Figura 39.- Reporte general que se conectaron a internet.....	101
Figura 40.- Paquetes Entrantes	101
Figura 41.- Paquetes Reenviados.....	102
Figura 42.- Paquetes salientes.....	102

ÍNDICE DE ANEXOS

Anexo 1.- Diagrama General de la Red del campus UNSM-T.....	106
Anexo 2.- Comparación entre el modelo OSI y el TCP/IP	107
Anexo 3.- Constancia de reconocimiento y respaldo de la UNSM-T	108
Anexo 4.-Evaluación Costo Beneficio	109

LISTA DE ACRÓNIMOS

ASN	Autonomous System Number
BSD	Berkeley Software Distribution
CPU	Central Processing Uni
DLL	Dynamic Link Library
FTP	File transfer protocol
FPU	Floating Point Unit
FAT32	File Allocation Table 32
GNU	General Public License
HPFS	High Performance File System
IAB	Internet Architecture Board
IP	Internet Protocol
ISO	International Organization for Standardization
LSB	Linux Standard Base
Minix	Unix Clone
NAT	Network Address Translation
NTFS	NT File System
POSIX	Portable Operating System Interface para UNIX
RFC	Request For Comments
SCO	Unix system intellectual property
SVR	Advanced Compatibility Package
STD	Security Tool Distro
TCP	Transmission Control Protocol

RESUMEN

El presente proyecto trata sobre la implementación del plan de mejora del ancho de banda de internet y seguridad aplicados a la red de datos basado en la metodología del diseño descendente de redes top down en la Universidad Nacional de San Martín – Tarapoto.

El proyecto inicia con el reconocimiento de la importancia, el impacto, el rendimiento de internet en su conjunto, siendo mal utilizado en la universidad. El objetivo se centra en optimizar el rendimiento de los servicios relacionados a internet a través de listas de control y filtrado de contenido que nos permite mejorar la seguridad.

Es importante realizar un estudio de la situación actual de la Universidad Nacional de San Martín, referente a la seguridad, esto incluye un análisis de su red local así como también de todas las funcionalidades de Internet que utilizan (navegación por las páginas web, publicación de weblogs y webs, correo electrónico, mensajería instantánea, foros, chats, gestiones y comercio electrónico, entornos para el ocio) que pueden significar algún riesgo.

Se decide utilizar una metodología altamente eficiente, en fases que está sistemáticamente organizadas, desarrollando la propuesta.

Se comienza desde el análisis de requerimiento donde identificamos los servicios, la situación actual, el tráfico interno/externo de la red, la máxima demanda, así como la caracterización del contenido al que la red de la universidad accede.

Seguimos luego con el diseño lógico, donde se divide la red, con la finalidad de no mezclar el contenido académico con el administrativo que son pilares exclusivos de la red de la universidad.

A continuación se revisa el diseño físico y se dan las modificaciones, así como la nueva configuración de la red reubicando dispositivos y cambiando la configuración de los clientes a fin de mejorar el rendimiento y utilizar las elevadas características técnicas de los equipos disponibles.

Posteriormente se trabaja con el producto de este proyecto que se encargará de realizar el filtrado de contenido, bloqueo y permiso de paquetes, cierre y apertura de protocolos, asignación dinámica de direccionamiento, priorización del ancho de banda.

Siendo la parte final, donde se utiliza ampliamente los reportes estadísticos del uso de servicios, a fin de mejorar el contenido, priorizando el ancho de banda, con la finalidad de ir optimizando el rendimiento de la red.

ABSTRACT

This project deals with the implementation of the improvement plan of internet bandwidth and security applied to data network based on the methodology of top - down network design in the National University of San Martín - Tarapoto.

The project begins with the recognition of the importance, impact, internet performance as a whole, being misused at the university. The focus is on optimizing the performance of services related to internet through checklists and content filtering that allows us to improve safety.

It is important to conduct a study of the current status of the National University of San Martín, regarding safety; this includes an analysis of its local network as well as all the features of Internet they use (browsing web pages, publishing weblogs and websites, email, instant messaging, forums, chats, efforts and e-commerce, leisure environments) that can entail some risk.

It was decided to use a highly efficient methodology, in phases which are systematically organized, developing the proposal.

It starts from the analysis of requirement where services are identified, the current situation, the internal/external network traffic, high demand, and the characterization of the content to which the network of the university access.

Then we continue with the logical design, where the network is divided, in order not to mix the academic content with the administrative that are unique pillars of the university network.

Then the physical design is reviewed and the changes are given, as well as the new configuration of the network relocating devices and changing the settings of the clients to improve the performance and use the high technical characteristics of the available equipment.

Then we work with the product of this project that will be responsible for performing content filtering, blocking and permit of packages, opening and closing protocols, dynamic assignment of routing, and bandwidth prioritization.

As the final part, where it is widely used statistical reports of the use of services, to improve the content, prioritizing bandwidth, in order to go optimizing network performance.

CAPÍTULO I. INTRODUCCIÓN

La Universidad Nacional de San Martín – Tarapoto (UNSM-T), es una institución descentralizada, autónoma con personería jurídica de derecho público interno, creada por D.L. N°22803 del 18 de diciembre de 1979 y ratificada por Ley 23261, el 18 de julio de 1981, se rige por la Constitución Política del Estado, la Legislación Universitaria vigente, el Estatuto y el reglamento General, tiene como misión formar profesionales académicos, desarrollo permanente de la investigación científica y tecnológica, la proyección y extensión social.

Por su ubicación, tiene una afluencia considerable de estudiantes de toda la región y provincias cercanas. En el ciclo 2013-I, su población estudiantil fue de 4,785 alumnos de pregrado, la demanda de acceso a internet se incrementó y la infraestructura de comunicaciones existente no es suficiente para responder a las necesidades de una comunidad universitaria dependiente de este servicio.

De acuerdo a los reportes y cálculos estadísticos que se muestran a continuación, se tiene:

Tabla 1. Reporte de uso de acceso a Internet

Concepto	Valor
Usuarios conectados	700
Ancho de banda disponible	6000 Kbps
Descarga promedio diaria	25.5 Mbps
Tasa de transferencia por usuario	8.57 Kbps
Tiempo promedio conectado por usuario	2.33 horas
Tasa de descarga por usuario	130.78 Kbps

Fuente: SARG Sistema Reporteador de Accesos del Squid (2013).

La población estudiantil está distribuida en 7 sedes descentralizadas (Tarapoto, Morales, Lamas, Moyobamba, Rioja, Juanjui y Tocache), 8 Facultades y 19 especialidades (Escuelas Profesionales) distribuidos de la siguiente manera:

Tabla 2. Cantidad de Alumnos Matriculados por Facultad y Escuela - Ciclo 2013-I

Facultad	Escuela Profesional	Matriculados
Ciencias agrarias	Agronomía - Tarapoto	322
	Agronomía - Tocache	101
	Medicina Veterinaria y Zootecnia – Tarapoto	113
Ingeniería agroindustrial	Ingeniería Agroindustrial – Tarapoto	311
	Ingeniería Agroindustrial - Juanjui	140
Ingeniería civil y arquitectura	Ingeniería civil – Tarapoto	446
	Arquitectura y Urbanismo – Tarapoto	261
Facultad de ciencias de la salud	Obstetricia – Tarapoto	302
	Enfermería – Tarapoto	288
	Medicina Humana – Tarapoto	87
Ecología	Ingeniería Ambiental – Moyobamba	347
	Ingeniería Sanitaria – Moyobamba	157
Educación y humanidades	Educación Inicial – Rioja	47
	Educación Primaria – Rioja	48
	Educación Secundaria con mención en Ciencias Naturales y Ecología – Rioja	29
	Idiomas – Tarapoto	244
Sistemas e informática	Ingeniería de Sistemas e Informática – Tarapoto	371
Ciencias económicas	Contabilidad – Tarapoto	373
	Administración en Turismo – Lamas	258
	Administración – Tarapoto	274
	Economía – Tarapoto	266
Total de alumnos matriculados		4785

Fuente: Reportes Oficina de Control y Registro Académico (2013).

A continuación se presente el cuadro resumen del total de personal docente que labora en la Universidad.

Tabla 3. Personal Docente que labora en la UNSM-T.

Condición	Categoría	Total
Ordinarios	Jefes de Práctica	6
	Auxiliar	104
	Asociado	105
	Principal	66
Contratados	Jefe de Práctica	5
	Auxiliar	58
	Asociado	0
	Principal	0
Total de docentes		344

Fuente: Reportes de la Unidad de Estadística de la UNSM-T (2013).

La UNSM-T, cuenta con una red integral de datos con un backbone de fibra óptica que permite integrar las áreas en una red de datos y es administrada desde el nodo concentrador de administración que está ubicada estratégicamente en el centro del campus universitario (Ver Anexo 1).

Miguel Valles (2013), administrador de la red menciona que el recurso más importante a compartir en la red de datos es internet, sin embargo por la demanda que este servicio tiene la calidad del servicio no es el adecuado, ralentizando los resultados de las búsquedas, así como el desempeño de las oficinas administrativas y académicas que dependen del correo electrónico, las búsquedas, bibliotecas, alertas, actualizaciones y comunicación en línea disponibles en el internet.

A continuación se muestra una tabla que resume el tráfico generado por el uso del servicio de internet:

Tabla 4. Tráfico generado por el uso del servicio de internet.

Fecha	Hora	Usuarios	Descargado	Promedio
23 /ago/2013	18:30:00	709	17.82G	25.13M
16 /ago/2013	18:30:00	709	17.82G	25.13M
09 /ago/2013	18:30:00	709	17.52G	24.72M
02 /ago/2013	18:30:00	684	16.84G	24.62M
26 /jul/2013	18:30:00	684	16.84G	24.62M
19 /jul/2013	18:30:00	684	16.84G	24.62M
12 /jul/2013	18:30:00	750	40.84G	54.46M
05 /jul/2013	18:30:00	750	40.84G	54.46M
28 /jun/2013	18:30:00	750	40.84G	54.46M
21 /jun/2013	18:30:00	682	17.17G	25.18M
14 /jun/2013	18:30:00	593	7.54G	12.72M
07 /jun/2013	18:30:00	710	10.75G	15.15M
31 /may/2013	18:30:00	710	10.75G	15.15M
24 /may/2013	18:30:00	710	10.75G	15.15M
17 /may/2013	18:30:00	710	10.75G	15.15M
10 /may/2013	18:30:00	710	10.75G	15.15M
03 /may/2013	18:30:00	678	8.58G	12.66M
26 /abr/2013	18:30:00	632	6.25G	9.89M

Fuente: SARG Sistema Reporteador de Accesos del Squid (2013).

Actualmente se tienen servidores instalados y configurados que controlan la salida hacia páginas de radio, música, televisión, vídeo, descargas, porno y chistes, sin embargo, el esquema de administración de estos servidores se basa en listas de control de acceso y verificación de urls fáciles de vulnerar utilizando técnicas para vulnerar la seguridad, ocasionando el incremento de tráfico de red, afectando la distribución del ancho de banda disponible.

En la red de la UNSM-T, es común la utilización de servicios que se encuentran fuera de la red y herramientas que ayuden a gestionar la red.

A continuación se lista los servidores disponibles en la red de datos:

Tabla 5. Lista de servidores disponibles en la red de datos.

Uso	Marca y modelo	S.O.
Servidor Proxy, Firewall.	HP Proliant ML370	Linux CentOS 6.3
Servidor de Pruebas	HP Proliant ML370	Linux Debian
Servidor WEB – Sistema SIGA – WEB	HP Proliant ML370 G6	Linux Debian
Servidor de Base de Datos	HP Proliant ML350 G8	CentOS 6.3
Servidor de contingencia (proxy/firewall)	HP Proliant ML350 G8	Linux CentOS 6.3

Fuente: Reportes de la Oficina de Administración de la Red de Fibra Óptica (2013).

La UNSM-T, consciente de la necesidad de enfrentar el futuro de forma competitiva se ha trazado dentro de sus planes de desarrollo, el examen permanente de la capacidad de respuesta frente a los requerimientos de sus usuarios, la competencia, su propia cultura y por sobre todo pensando en la proyección a futuro, de universidad acreditada y con excelente desempeño académico.

Realizar este proceso, requiere que el mismo no sea ajeno al desarrollo tecnológico, la aplicación de la informática y de las comunicaciones, como herramienta de apoyo a la consecución de los objetivos institucionales, que genere ventaja competitiva, eficiencia en procesos, seguridad y fiabilidad en la información procesada y compartida, integración de sus funciones y áreas, minimizando costos, entre otras posibilidades.

Al ser la UNSM-T, una institución del estado, el presupuesto para la prestación de servicios es limitada y la optimización, cuidado y uso de sus recursos es fundamental.

Para solucionar los problemas descritos anteriormente se elaboró un plan de mejora del ancho de banda de internet y seguridad aplicados a la red de datos basado en la metodología del diseño descendente de redes top down, en la Universidad Nacional de San Martín, para esto se utilizará una metodología que permita definir adecuadamente cuáles serán las políticas y procedimiento necesarios a aplicar para mejorar el servicio e incrementar la seguridad, identificando cuál es el comportamiento de la red a través de la metodología del diseño descendente de redes a fin de que este proceso sea sistemático y metodológico.

Esta metodología se presenta en fases claramente definidas que son:

Análisis de requerimientos

En esta fase el analista de red entrevista a los usuarios y personal técnico para obtener un mayor entendimiento de los objetivos técnicos y de negocio para el nuevo sistema o actualización. La tarea de representar la red existente, incluyendo la topología física y lógica como también el rendimiento de la red. Los últimos pasos de esta fase es analizar el tráfico de red actual y futuro, como también los comportamientos de protocolo y la calidad de servicio requerido.

Determinar cuáles serán las mejores políticas para el filtrado y censura del servicio. Además se utilizará la base de datos disponible en los servidores identificando las características de la red de datos de la UNSM-T.

Diseño lógico

En esta fase se representa la topología de la nueva red o actualización, direccionamiento de capas de red, protocolos de nombre, intercambio y enrutado. El diseño lógico también incluye el planeamiento de seguridad, la administración de la red y la investigación inicial para que los proveedores de servicio puedan cumplir con el acceso remoto y a la WAN.

Esto se realizará utilizando el análisis de requerimiento realizado, a fin de determinar los niveles de seguridad requeridos, si se tratará de una red de 2 o 3 capas a fin de segmentar el dominio de colisión y dominio de broadcast de los edificios y mejorar la seguridad de la red ante posibles ataques externos.

Para ello nos auxiliaremos de los planos y la documentación existente identificando nodos de distribución que generen elevado tráfico, diseñando así la distribución lógica de paquetes, diseño de la Zona Desmilitarizada -DMZ, ip's, dominios de colisión y broadcast más adecuado para la red.

Diseño físico

Durante la fase del diseño físico se especifica las tecnologías y productos para llevar a cabo los diseños lógicos seleccionados. En esta fase también debe completada la investigación de proveedores de servicio que se inició en la fase anterior.

En esta fase lo que se hará es documentar la distribución física de los dispositivos, implica además redistribuir el cableado en la oficina de distribución de

la red, creación de las Vlan's y determinación de la ubicación física de los servidores (si se encontrarán en la DMZ, la red perimetral o la red externa).

Pruebas, optimización y documentar el diseño

El paso final consiste en redactar e implementar el plan de prueba y construir un prototipo o piloto, optimizar el diseño de red y documentar el trabajo con el diseño de red propuesto.

Finalmente para garantizar larga duración de la solución propuesta, la documentación será un aspecto importante en la red de la UNSM-T, para ello se utilizarán diferentes herramientas de ofimática.

Una ventaja importante del diseño descendente es que es rápido: inicia con la tecnología que se tiene disponible y la tecnología para construir una red.

Cuando se inicia un proyecto de diseño descendente, se identifican los objetivos de negocio, sus objetivos para la red y si sus principales preocupaciones son la seguridad, la velocidad, el rendimiento o alguna combinación de ellos. Con diseño descendente, nos fijamos en la tecnología sólo después de resolver los problemas que necesita resolver. Esto lleva más tiempo que un simple diseño, pero tiene la ventaja de que lo está adaptando a las necesidades del usuario. Esto debe resultar en un usuario más satisfecho.

Un buen diseño descendente no se conforma con aceptar una meta general, se pone en detalle lo que el cliente requiere. Si la prioridad del cliente es contar con una red rápida y sin cuellos de botella, el siguiente paso es hablar con los usuarios. Lo que te dicen puede determinar si su prioridad debe ser la creación de listas de control de acceso y filtrado de paquetes más eficientes.

Para este proyecto se definieron los siguientes objetivos: Primero, Definir las listas de control de acceso que determinen los límites de uso de los servicios relacionados a internet utilizados en la red de datos de la Universidad Nacional de San Martín. Implantar políticas y procedimientos mediante el filtrado de paquetes a fin de mejorar la seguridad de la red perimetral de la Universidad Nacional de San Martín – Tarapoto. Aplicar la metodología del diseño descendente de redes top down para optimizar el rendimiento de la red.

Luego de identificar las causas que están generando el problema del bajo rendimiento de los servicios relacionado a internet en la UNSM-T e identificada la

metodología que aplique los criterios más adecuados para garantizar el uso adecuado y administración de las tecnologías de información sobre la plataforma base existente, aplicada la solución tecnológica y realizadas las pruebas necesarias para garantizar su éxito, el escenario que se pretende dejar implementado será de una administración, con mayores niveles de seguridad, permitiendo a los encargados de la oficina de administración de la red, disponer de mayor tiempo para dar valor agregado a sus funciones y mejorar en términos cualitativos las características de la red de datos de la universidad.

Las ventajas de la solución planteada para la UNSM-T, se valoran porque la metodología a utilizar se adecúa a la plataforma de hardware existente y al máximo aprovechamiento de las características técnicas de la misma, pues se cuentan con los equipos necesarios y adecuados, sólo que al realizar el levantamiento de información de las necesidades de distribución, segmentación, priorización y seguridad de la red el diseño lógico será más adecuado para el escenario actual.

La presente investigación también se justifica, porque gracias a la metodología del diseño descendente de redes top down, se podrá documentar el trabajo realizado para que futuras mejoras a implantarse en la solución puedan fácilmente acoplarse a lo diseñado/planificado, evitando realizar el trabajo desde cero, y aplicando las mejoras sobre una solución robusta, escalable, modular y segura.

La solución permitirá mejorar el tiempo de respuesta de la red, con lo que la percepción de la calidad del servicio proporcionado a la comunidad universitaria se mejora, permitiéndoles optimizar el uso de sus horas disponibles a fin de utilizarlas en actividades relacionadas.

Se justifica además porque permitirá dimensionar el tráfico generado, analizando el uso de la red mediante aplicaciones diseñadas para este propósito, además del análisis de los reportes generados por la configuración actual, optimizando el filtrado de paquetes y el análisis de las peticiones mediante URL, para determinar cuál serán las mejores políticas de filtrado y listas de control de acceso que garanticen un adecuado uso del recurso internet.

Este documento se estructuró de la siguiente manera: En el Capítulo I se muestra la identificación del problema, el planteamiento de la investigación, los objetivos definidos y la justificación de la investigación. En el capítulo II se presenta el marco teórico que se utilizó como fundamento para desarrollar la investigación entre ellos

destacan la metodología top down, casos de éxito, En el capítulo III la metodología de investigación de esta sección se presenta el lugar de aplicación, el tipo de investigación y la metodología de investigación. El capítulo IV se presenta la solución el producto que se desarrolló utilizando la metodología top down, webmin para reportar, squip. En el capítulo V se muestra la validación y el análisis de resultados. Finalmente se menciona las conclusiones y recomendaciones del proyecto.

CAPÍTULO II. MARCO TEÓRICO

2.1 Introducción

El presente capítulo presenta la fundamentación teórica de la investigación; considerando la metodología descendente de redes top down; siendo de vital importancia mencionar al modelo de procesos, tipo de procesos e indicadores de procesos

2.2 Sobre la Universidad Nacional de San Martín

La Universidad Nacional de San Martín - Tarapoto (UNSM-T) fue creada por D.L. N° 22803 el 18/12/79 en la ciudad de Tarapoto, como consecuencia de la lucha del pueblo Sanmartinense por obtener una institución educativa con nivel universitario, ratificándose con Ley N° 23262 el 18/07/81. En diciembre del mismo año se instala la Primera Comisión de Gobierno, presidida por el Ing. Raúl Ríos Reátegui.

El 17 de Mayo de 1982 la UNSM-T, inicia formalmente sus actividades académicas con las Carreras Profesionales de:

- Agronomía, Ingeniería
- Agroindustrial
- Ingeniería Civil
- Obstetricia.

La UNSM-T es un centro superior de estudios, autónoma y de carácter estatal, nuestro compromiso es formar profesionales académicos competentes con responsabilidad social, participando plenamente en la transformación de la sociedad para su desarrollo integral, mediante la generación de innovación de conocimientos, cultura y valores, en un proceso permanente de actualización y acreditación.

Luego de varios años de trabajo la Primera Comisión de Gobierno entrega el cargo a la Comisión Organizadora presidida por el Ing. Augusto Montes Gutiérrez, nombrada por la ANR, el cual a su vez da paso a la última Comisión Organizadora presidida por el Dr. Roberto Calderón Gonzáles, la misma que culmina el periodo de implementación académica administrativa y da las condiciones para la adecuación a la Ley Universitaria. De esa manera, luego de las elecciones de la Asamblea Universitaria y la promulgación del Estatuto, se dio la elección para la conformación de la Asamblea Universitaria, este organismo en 1993 eligió como primer Rector de la UNSM-T al Dr. Jorge González Ramírez. En 1995 se crean las Facultades de:

- Educación y Humanidades con sede en la ciudad de Rioja
- Ecología en Moyobamba
- Ingeniería de Sistemas e Informática
- Ciencias Económicas en Tarapoto
- Carrera Profesional de Turismo en Lamas

Logrando de esta manera la descentralización de las carreras profesionales e incrementando las posibilidades de profesionalización a los jóvenes en las ciudades mencionadas.

Ante la renuncia del Dr. Jorge González Ramírez, fue elegido como Rector al Lic. Marco Armando Gálvez Díaz en febrero de 1997, estando en el cargo hasta Setiembre del 2001, dando paso a la Comisión de Orden y Gestión el 14/02/2002 nombrada por la ANR y presidida por el Lic. Arturo Ruíz Chapilliquén. Esta Comisión entrega el cargo a las nuevas autoridades, elegidos democráticamente mediante un proceso eleccionario el 29 de agosto del año 2003, en el que fue elegido como actual Rector el Ing.M.Sc. Alfredo Quinteros García, Vicerrector Académico el Ing.M.Sc. Abner Milán Barzola Cárdenas y Vicerrector Administrativo el Econ.M.Sc. Réniger Sousa Fernández, quienes están comprometidos con la región y el país de forjar líderes profesionales con una visión integradora.

En agosto del 2008 fue elegido como rector el Ing. M. Sc. Alfredo Quinteros García para continuar en el cargo hasta agosto del 2013. Asimismo fue elegido como Vicerrector Académico el Ing. M.Sc. Julio Armando Ríos Ramírez y Vicerrector Administrativo Ing. M. Sc. Jorge Sánchez Ríos, quienes asumen el compromiso de seguir forjando el desarrollo de la UNSM-T hasta agosto del 2013.

Misión

Somos una institución universitaria formadora de profesionales competitivos para la sociedad, generando innovación de conocimientos y fortaleciendo cultura y valores en proceso de acreditación tips de belleza y moda

Visión

La Universidad Nacional de San Martín - Tarapoto, es una institución amazónica acreditada, líder en la formación profesional al servicio de la sociedad.

2.3 Sobre la oficina de administración de red óptica

Misión

Brindar soporte tecnológico de información y comunicación, al proceso de innovación de conocimientos y al fortalecimiento de cultura y valores para el proceso de acreditación.

Visión

La oficina de Informática y comunicaciones es una unidad orgánica pionera en innovación tecnológica que contribuye a la modernización, calidad y productividad con proyección a la comunidad.

Funciones:

- Formular y ejecutar el plan de desarrollo informático de la UNSM-T.
- Desarrollar y administrar el funcionamiento de la red integral de la UNSM-T.
- Mantener y administrar las comunicaciones internas y externas de la UNSM-T.
- Mantener y administrar servicios de correos y servicios de internet.
- Mantener y actualizar la página web de nuestra casa superior de estudios.
- Mantener y administrar las bases de datos.
- Desarrollar medidas de seguridad para salvaguardar la información de la base de datos, correo, página web de la Institución.
- Controlar licencias adquiridas por nuestra institución con las instaladas en las estaciones de la red de la UNSM-T.
- Normar y supervisar el uso de software y hardware.
- Asesorar, evaluar y brindar el soporte técnico de software y Hardware a las dependencias de la UNSM-T
- Controlar por medio de un inventario periódico del hardware que poseen las computadoras de nuestra casa de estudios.

Datos de contacto : Correo Electronico: informatica@unsm.edu.pe

Personal : Director de OlyC Ing. Carlos Enrique López Rodríguez.

: Secretaria Grethel Victoria Pasquel Quevedo

Unidades de la oficina de informática y comunicaciones y personal a cargo.

Unidad : Infraestructura tecnológica

Área : Soporte tecnológico

Auxiliar Einstein Jean Lozano Flores

Auxiliar Jose Luis Chavarri Perez

Tec. Admin. ST Tercero Víctor Navarro Gonzáles

Área : Administración de la fibra óptica

Encargatura F.O Ing. Miguel Ángel Valles Coral

Unidad : Administración de sistemas de información

Jefe de Unidad Ing. Víctor Manuel Vallejos Monja

 Ing. Fiorella Mercedes Vincés Mori

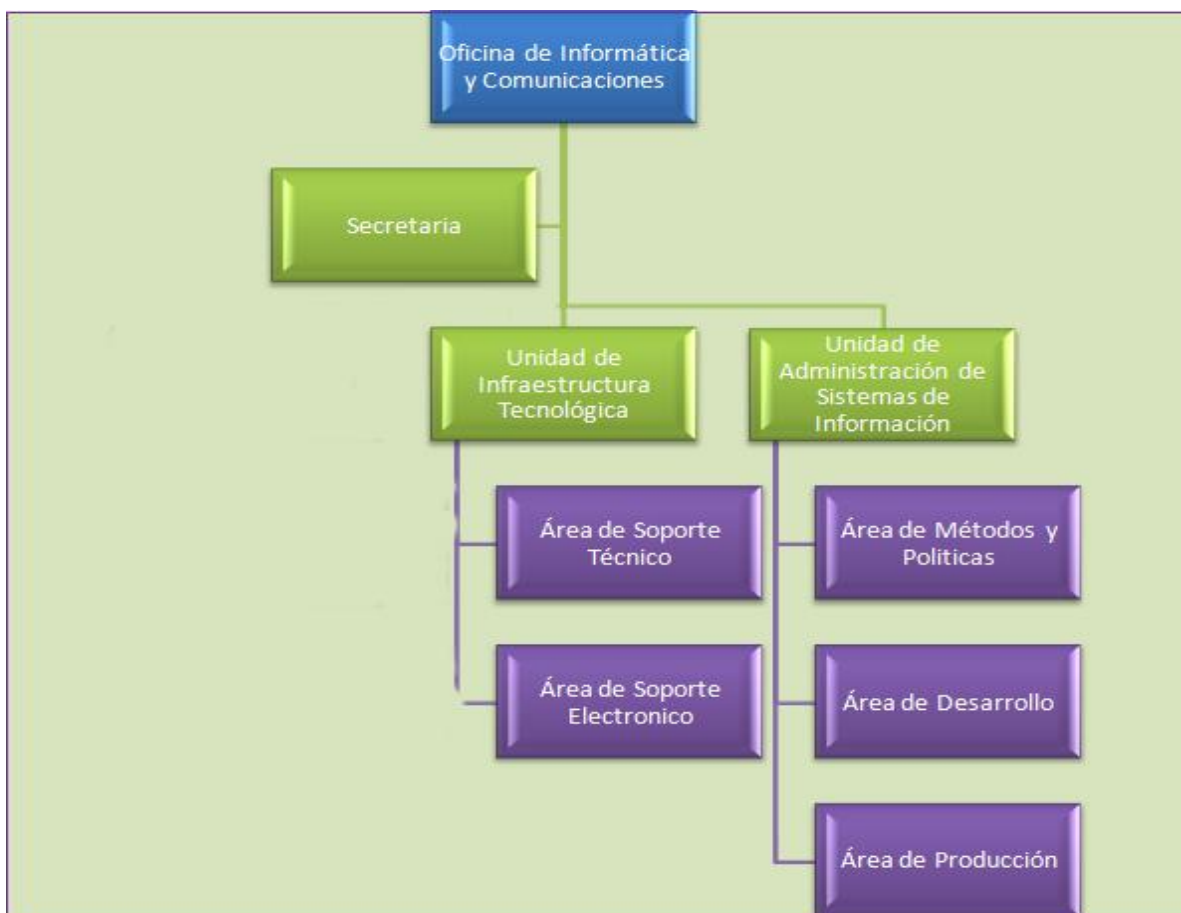


Figura 1 Organigrama de oficina de informática y comunicaciones

2.4 Sobre la infraestructura tecnológica y la red de datos.

La Universidad Nacional de San Martín (UNSM-T), consciente de la necesidad de enfrentar el futuro de forma competitiva se ha trazado dentro de sus planes de desarrollo, el examen permanente de la capacidad de respuesta frente a los requerimientos de sus usuarios, la competencia, su propia cultura y por sobre todo pensando en la proyección a futuro de universidad acreditada y con excelente desempeño académico.

Realizar este proceso, requiere que el mismo no sea ajeno al desarrollo tecnológico y la aplicación de la informática y de las comunicaciones, como herramienta de apoyo a la consecución de los objetivos institucionales, que genere ventaja competitiva, eficiencia en procesos, seguridad y fiabilidad en la información procesada y compartida, integración de sus funciones y áreas, minimización de costos, entre otras posibilidades.

La UNSM-T apostando a sus fortalezas y características académicas especiales en nuestro medio no escapa a este requisito de desarrollo.

Se reconoce además que es la institución de formación Universitaria líder en nuestra región, identificada en su mayor experiencia, mayor infraestructura, mayor población estudiantil, docente y administrativa, así como su estratégica ubicación geográfica.

El desarrollo informático de una institución con estas características debe ser planificado y la infraestructura tecnológica de redes y los sistemas de información que lo soporte será de importancia radical para garantizar que las mismas continúen como ventaja competitiva y comparativa.

Estos dos componentes son complementarios entre sí; no podrá desarrollarse aplicaciones informáticas eficientes que generen información útil si no existe la infraestructura tecnológica adecuada que los soporte; ni podrá implantarse dicha tecnología sin que paralelamente se haya planificado los sistemas de información y los servicios que se implementarán en ella.

Se describe las características y especificaciones técnicas y operativas de funcionamiento de la infraestructura tecnológica de la Universidad Nacional de San Martín – Tarapoto, es decir, la red de transmisión de información principal, que integra a todas sus sedes en Tarapoto, Morales y Lamas, con capacidades de transmisión a alta velocidad de un gran volumen de datos e información, en formatos distintos como texto, datos, voz, video; así como una comunicación eficiente, segura y controlada con Internet.

Cabe recalcar que parte de esta red está conforme a los exigentes estándares de calidad y seguridad, que internacionalmente los sistemas de comunicaciones requieren (estándares ANSI/TIA/EIA), con la finalidad de seguir siendo una institución comprometida con el avance tecnológico y con la formación de profesionales de amplia capacidad competitiva a nivel nacional e internacional, además de preparar y

orientar a la institución hacia la búsqueda de una certificación y acreditación de calidad internacional.

2.5 Linux

Linux es un sistema operativo diseñado por cientos de programadores de todo el planeta, aunque el principal responsable del proyecto es Linus Torvalds. Su objetivo inicial es impulsar el software de libre, distribuir junto con su código fuente para que pueda ser modificado por cualquier persona, dando rienda suelta a la creatividad. El hecho de que el sistema operativo incluya su propio código fuente expande enormemente las posibilidades de este sistema. Este método también es aplicado en numerosas ocasiones a los programas que corren en el sistema, lo que hace que podamos encontrar muchos programas útiles totalmente gratuitos y con su código fuente.

Las funciones principales de este sistema operativo son:

1. Sistema multitarea: En Linux es posible ejecutar varios programas a la vez sin necesidad de tener que parar la ejecución de cada aplicación.
2. Sistema multiusuario: Varios usuarios pueden acceder a las aplicaciones y recursos del sistema Linux al mismo tiempo. Y, por supuesto, cada uno de ellos puede ejecutar varios programas a la vez (multitarea).
3. Shells programables: Un shell conecta las órdenes de un usuario con el Kernel de Linux (el núcleo del sistema), y al ser programables se puede modificar para adaptarlo a tus necesidades. Por ejemplo, es muy útil para realizar procesos en segundo plano.
4. Independencia de dispositivos: Linux admite cualquier tipo de dispositivo (módems, impresoras) gracias a que cada vez instalado uno nuevo, se añade al Kernel el enlace o controlador necesario con el dispositivo, haciendo que el Kernel y el enlace se fusionen. Linux posee una gran adaptabilidad y no se encuentra limitado como otros sistemas operativos.
5. Comunicaciones: Linux es el sistema más flexible para poder conectarse a cualquier ordenador del mundo. Internet se creó y desarrollo dentro del mundo de

Unix, y por lo tanto Linux tiene las mayores capacidades para navegar, ya que Unix y Linux son sistemas prácticamente idénticos.

Linux no sacrifica en ningún momento la creatividad, tal y como lo hacen algunas compañías informáticas. Linux es una ventana abierta por la que es posible huir hacia un mundo donde la verdadera informática puede ser disfrutada sin límites ni monopolios.

Linux es distribuido mediante una serie de distribuciones como RedHat, Slackware, Debian, Ubuntu, Centos, las cuales se diferencian por su método de instalación y por los paquetes (software) que viene incluido. Todo el software de Linux está regido por la licencia de GNU, con la cual cualquier persona puede modificar un programa y venderlo según el desee, con la condición que la persona que compra ese producto puede realizar la misma acción o simplemente hacer copias para todos aquellos que lo quieran sin tener que pagar más. Esta licencia es la garantía que afirma la absoluta libertad de este sistema operativo.

Historia de linux

Linux, es un sistema operativo. Es una implementación de libre distribución UNIX para computadoras personales (PC), servidores y estaciones de trabajo.

Linux es la denominación de un sistema operativo tipo-Unix y el nombre de un núcleo.

Es uno de los modelos más prominentes del software libre y del desarrollo del código abierto, cuyo código fuente está disponible públicamente, para que cualquier persona pueda libremente usarlo, estudiarlo, redistribuirlo y, con los conocimientos informáticos adecuados, modificarlo.

Linux es usado como sistema operativo en una amplia variedad de plataformas de hardware y computadores, incluyendo los computadores de escritorio (PCs x86 y x86-64, Macintosh y PocketPC), servidores, supercomputadores, mainframes, y dispositivos empotrados así como teléfonos celulares.

En 1983 Richard Stallman fundó el proyecto GNU, con el fin de crear sistemas operativos parecidos a UNIX y compatibles con POSIX. Dos años más tarde creó la "Fundación del Software Libre" y escribió la GNU General Public License para posibilitar el software libre en el sistema de copyright.

El software GNU se extendía muy de prisa y dentro de poco una multitud de programas fueron escritos, de manera que ya a principios de 1990 había bastantes software GNU como para hacer un sistema operativo propio, pero faltaba el Kernel.

A principios de los años 1990, no había un sistema operativo libre completo a pesar de que el proyecto GNU era desarrollado constantemente, no disponía sin embargo de ningún buen Kernel basado en UNIX, por el contrario era un número de proyectos de software libres que podían ser traducidos en las variantes UNIX mediante el compilador de GNU.

Su creador Linus Benedict Torvalds nació en Helsinki, Finlandia, en el año de 1969. Su abuelo, matemático y estadista le compró un Comodore en 1980 y fue quien "enganchó" a Linus al mundo de los computadores.

En 1988 Linus Torvalds entra a la Universidad. Ese mismo año fue cuando el sistema operativo didáctico, basado en UNIX y creado por Andy Tannenbaum, empezó a cobrar importancia. Dicho sistema era el Minix.

Linux entró a formar parte de la comunidad de usuarios Minix. Andy Tannenbaum cometió un error en su sistema operativo. Era demasiado limitado, tanto técnicamente como políticamente, en ningún momento tuvo en cuenta la posibilidad de incluir Minix al proyecto GNU. La creación de Andy Tannenbaum estaba pensando para ser distribuida. Su primer error fue ceder todos sus derechos a Prentice Hall, que empezó a cobrar 150 dólares por licencia.

Así, Linux tomó la decisión de cambiar esta política debido a que el sistema Minix era ideal para los estudiantes de sistemas operativos, y su precio era considerablemente alto.

Año 1991, cuando Linus se acabó de comprar su primer 386, la intención era crear un nuevo Kernel (al que posteriormente llamaría Linux) de UNIX basado en el Kernel de Minix y modificarlo periódicamente de manera que fuera capaz de ejecutar aplicaciones GNU.

La historia de Linux está fuertemente vinculada a la del proyecto GNU. Hacia 1991, cuando la primera versión del núcleo Linux fue liberada, el proyecto GNU había producido varios de los componentes del sistema operativo, incluyendo un intérprete de comandos, una biblioteca C y un compilador, pero aún no contaba con el núcleo

que permitiera complementar el sistema operativo. Entonces, el núcleo creado por Linus Torvalds, llenó el hueco final que el sistema operativo GNU exigía.

Linux nunca anunció la versión 0.01 de Linux (agosto 1991), esta versión no era ejecutable, solamente incluía los principios del núcleo del sistema, estaba escrita en lenguaje ensamblador y asumía que uno tenía acceso a un sistema Minix para su compilación.

El 5 de octubre de 1991, Linus anuncio la primera versión "Oficial" de Linux, - versión 0.02, con esta versión Linus pudo ejecutar Bash (GNU Bourne Again Shell) y gcc (Compilador GNU de C) pero no mucho más funcionaba. En este estado de desarrollo ni se pensaba en los términos soporte, documentación, distribución. Después de la versión 0.03, Linus salto en la numeración hasta la 0.10, más programadores a lo largo y ancho del internet empezaron a trabajar en el proyecto y después de revisiones, Linus incremento el número de versión hasta la 0.95 (marzo 1992). En Diciembre de 1993 el núcleo del sistema estaba en la versión 0.99 y la versión 1.0, llego el 14 de marzo de 1994.

Linux se refiere estrictamente al núcleo Linux, pero es comúnmente utilizado para describir al sistema operativo tipo Unix (que implementa el estándar POSIX), que utiliza primordialmente filosofía y metodologías libres (también conocido como GNU/Linux) y que está formado mediante la combinación del núcleo Linux con las bibliotecas y herramientas del proyecto GNU y de muchos otros proyectos/grupos de software (libre o no libre).

La expresión "Linux" es utilizada para referirse a las distribuciones GNU/Linux, colecciones de software que suelen contener grandes cantidades de paquetes además del núcleo. El software que suelen incluir consta de una enorme variedad de aplicaciones, como: entornos gráficos, suites ofimáticas, servidores web, servidores de correo, servidores FTP. Coloquialmente se aplica el término "Linux" a éstas. Algunas personas opinan que es incorrecto denominarlas distribuciones Linux, y proponen llamarlas sistema GNU/Linux. Otras personas opinan que los programas incluidos proceden de fuentes tan variadas que proponen simplificarlo denominándolo simplemente a "Linux".

Características de linux

Linux implementa la mayor parte de las características que se encuentran en otras implementaciones de UNIX, más algunas otras que no son habituales:

- Multitarea: varios programas (realmente procesos) ejecutándose al mismo tiempo.
- Multiusuario: varios usuarios en la misma máquina al mismo tiempo (sin licencias para todos!).
- Multiplataforma: corre en muchas CPUs distintas, no sólo Intel.
- Funciona en modo protegido 386.
- Tiene protección de la memoria entre procesos, de manera que uno de ellos no pueda colgar el sistema.
- Carga de ejecutables por demanda: Linux sólo lee de disco aquellas partes de un programa que están siendo usadas actualmente.
- Política de copia en escritura para la compartición de páginas entre ejecutables: esto significa que varios procesos pueden usar la misma zona de memoria para ejecutarse. Cuando alguno intenta escribir en esa memoria, la página (4Kb de memoria) se copia a otro lugar. Esta política de copia en escritura tiene dos beneficios: aumenta la velocidad y reduce el uso de memoria.
- Memoria virtual usando paginación (sin intercambio de procesos completos) a disco: una partición o un archivo en el sistema de archivos, o ambos, con la posibilidad de añadir más áreas de intercambio sobre la marcha (se sigue denominando intercambio, es en realidad un intercambio de páginas). Un total de 16 zonas de intercambio de 128Mb de tamaño máximo pueden ser usadas en un momento dado con un límite teórico de 2Gb para intercambio.
- La memoria se gestiona como un recurso unificado para los programas de usuario y para el caché de disco, de tal forma que toda la memoria libre puede ser usada para caché y éste puede a su vez ser reducido cuando se ejecuten grandes programas.
- Librerías compartidas de carga dinámica (DLL's) y librerías estáticas también, por supuesto.
- Se realizan volcados de estado (core dumps) para posibilitar los análisis post-mortem, permitiendo el uso de depuradores sobre los programas no sólo en ejecución sino también tras abortar éstos por cualquier motivo.
- Casi totalmente compatible con POSIX, System V y BSD a nivel fuente.
- Mediante un módulo de emulación de iBCS2, casi completamente compatible con SCO, SVR3 y SVR4 a nivel binario.
- Todo el código fuente está disponible, incluyendo el núcleo completo y todos los drivers, las herramientas de desarrollo y todos los programas de usuario; además todo ello se puede distribuir libremente. Hay algunos programas comerciales que están siendo ofrecidos para Linux actualmente sin código fuente, pero todo lo que ha sido gratuito sigue siendo gratuito.
- Control de tareas POSIX.
- Pseudo-terminales (pty's).

- Emulación de 387 en el núcleo, de tal forma que los programas no tengan que hacer su propia emulación matemática. Cualquier máquina que ejecute Linux parecerá dotada de coprocesador matemático. Por supuesto, si tu ordenador ya tiene una FPU (unidad de coma flotante), será usada en lugar de la emulación, pudiendo incluso compilar tu propio kernel sin la emulación matemática y conseguir un pequeño ahorro de memoria.
- Soporte para muchos teclados nacionales o adaptados y es bastante fácil añadir nuevos dinámicamente.
- Consolas virtuales múltiples: varias sesiones de login a través de la consola entre las que se puede cambiar con las combinaciones adecuadas de teclas (totalmente independiente del hardware de video). Se crean dinámicamente y puedes tener hasta 64.
- Soporte para varios sistemas de archivo comunes, incluyendo minix-1, Xenix y todos los sistemas de archivo típicos de System V, y tiene un avanzado sistema de archivos propio con una capacidad de hasta 4 Tb y nombres de archivos de hasta 255 caracteres de longitud.
- Acceso transparente a particiones MS-DOS WINDOWS (o a particiones OS/2 FAT32, NTFS) mediante un sistema de archivos especial: no necesitas ningún comando especial para usar la partición MS-DOS WINDOWS, parece un sistema de archivos normal de Unix.
- Un sistema de archivos especial llamado UMSDOS que permite que Linux sea instalado en un sistema de archivos DOS.
- Soporte en sólo lectura de HPFS-2 del OS/2 2.1
- Sistema de archivos de CD-ROM que lee todos los formatos estándar de CD-ROM.
- TCP/IP, incluyendo ftp, telnet, NFS, etc.
- Appletalk disponible en el actual núcleo de desarrollo.
- Software cliente y servidor Netware disponible en los núcleos de desarrollo.

2.6 Squid

Servidor proxy

El término en inglés Proxy, tiene un significado muy general y al mismo tiempo ambiguo, aunque invariablemente se considera un sinónimo del concepto de intermediario. Se suele traducir, en el sentido estricto, como delegado o apoderado (el que tiene poder sobre otro).

Un Servidor Intermediario se define como una computadora o dispositivo que ofrece un servicio de red que consiste en permitir a los clientes realizar conexiones de red indirectas hacia otros servicios de red. Durante el proceso ocurre lo siguiente:

Cliente se conecta hacia un Servidor Proxy.

Cliente solicita una conexión, archivo u otro recurso disponible en un servidor distinto.

Servidor Intermediario proporciona el recurso ya sea conectándose hacia el servidor especificado o sirviendo éste desde un caché.

En algunos casos el Servidor Intermediario puede alterar la solicitud del cliente o bien la respuesta del servidor para diversos propósitos.

Los Servidores Proxy generalmente se hacen trabajar simultáneamente como muro cortafuegos operando en el nivel de red, actuando como filtro de paquetes, como en el caso de iptables o bien operando en el Nivel de Aplicación, controlando diversos servicios, como es el caso de TCP Wrapper. Dependiendo del contexto, el muro cortafuegos también se conoce como BPD o Border Protection Device o simplemente filtro de paquetes.

Una aplicación común de los Servidores Proxy es funcionar como caché de contenido de Red (principalmente HTTP), proporcionando en la proximidad de los clientes un caché de páginas y archivos disponibles a través de la Red en servidores HTTP remotos, permitiendo a los clientes de la red local acceder hacia éstos de forma más rápida y confiable.

Cuando se recibe una petición para un recurso de red especificado en un URL (Uniform Resource Locator) el Servidor Intermediario busca el resultado del URL dentro del caché. Si éste es encontrado, el Servidor Intermediario responde al cliente proporcionando inmediatamente el contenido solicitado. Si el contenido solicitado estuviera ausente en el caché, el Servidor Intermediario lo traerá desde servidor remoto, entregándolo al cliente que lo solicitó y guardando una copia en el caché. El contenido en el caché es eliminado luego a través de un algoritmo de expiración de acuerdo a la antigüedad, tamaño e historial de respuestas a solicitudes (hits) (ejemplos: LRU, LFUDA y GDSF).

Los Servidores Proxy para contenido de Red (Web Proxies) también pueden actuar como filtros del contenido servidor, aplicando políticas de censura de acuerdo a criterios arbitrarios.

Servidor Squid.

Squid es un servidor intermediario de alto desempeño que se ha venido desarrollando desde hace varios años y es hoy en día un muy popular y ampliamente

utilizado entre los sistemas operativos como GNU/Linux y derivados de Unix. Es muy confiable, robusto y versátil y se distribuye bajo los términos de la Licencia Pública General GNU (GNU/GPL). Siendo equipamiento lógico libre, está disponible el código fuente para quien así lo requiera.

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

Squid consiste de un programa principal como servidor, un programa para búsqueda en servidores DNS, programas opcionales para reescribir solicitudes y realizar autenticación y algunas herramientas para administración y herramientas para clientes. Al iniciar Squid da origen a un número configurable (de modo predeterminado a través de la opción `dns_children`) de procesos de búsqueda en servidores DNS, cada uno de los cuales realiza una búsqueda única en servidores DNS, reduciendo la cantidad de tiempo de espera para las búsquedas en servidores DNS.

Equipamiento lógico necesario.

Para poder llevar al cabo los procedimientos descritos en este y otros documentos relacionados, se requiere instalar al menos lo siguiente:

- Al menos `squid-2.5.STABLE6`
- Todos los parches de seguridad disponibles para la versión del sistema operativo que esté utilizando.
- Un muro cortafuegos configurado con `system-config-firewall`, `Firestarter` o `Shorewall`.

Squid sólo se instala de manera predeterminada cuando se instala el grupo de paquetes denominado «Servidor Web». El procedimiento de instalación es exactamente el mismo que con cualquier otro equipamiento lógico.

Instalación a través de yum.

Si se utiliza CentOS o Red Hat Enterprise Linux, ejecute:

- `yum -y install squid`

Configuración básica.

Squid utiliza el archivo de configuración localizado en `/etc/squid/squid.conf` y podrá trabajar sobre este utilizando su editor de texto simple preferido. Existen un gran número de opciones, de los cuales recomendamos configurar los siguientes:

- Al menos una Lista de Control de Acceso
- Al menos una Regla de Control de Acceso
- `http_port`
- `cache_dir`
- `error_directory`, sólo si va a personalizar mensajes de error.

El resto de las opciones mencionadas en este documento son opcionales.

Edite el archivo `/etc/squid/squid.conf`:

- `vi /etc/squid/squid.conf`

Controles de acceso.

Para poder controlar el tráfico de los clientes hacia Internet, es necesario establecer Listas de Control de Acceso que definan una red o bien ciertos anfitriones en particular. A cada lista se le asignará una Regla de Control de Acceso que permitirá o denegará el acceso a Squid.

Listas de control de acceso.

De modo predeterminado en CentOS 6 y Red Hat Enterprise Linux 6, Squid habilita el acceso a todas las redes locales, definidas en el RFC1918. Es decir, permite el acceso a `10.0.0.0/8`, `172.16.0.0/12`, `192.168.0.0/16`, `fc00::/7` y `fe80::/10`.

```
# Example rule allowing access from your local networks.
# Adapt to list your (internal) IP networks from where browsing
# should be allowed
acl localnet src 10.0.0.0/8      # RFC1918 possible internal network
acl localnet src 172.16.0.0/12  # RFC1918 possible internal network
acl localnet src 192.168.0.0/16 # RFC1918 possible internal network
acl localnet src fc00::/7      # RFC 4193 local private network range
acl localnet src fe80::/10     # RFC 4291 link-local (directly plugged) machines
```

Deshabilite todo lo anterior, colocando una almoadilla (`#` al inicio de cada línea).

```
# Example rule allowing access from your local networks.
# Adapt to list your (internal) IP networks from where browsing
# should be allowed
# acl localnet src 10.0.0.0/8      # RFC1918 possible internal network
# acl localnet src 172.16.0.0/12  # RFC1918 possible internal network
# acl localnet src 192.168.0.0/16 # RFC1918 possible internal network
# acl localnet src fc00::/7      # RFC 4193 local private network range
```

```
# acl localnet src fe80::/10 # RFC 4291 link-local (directly plugged)
```

Regularmente una lista de control de acceso se establece con la siguiente sintaxis:

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

Si se desea establecer una lista de control de acceso que abarque a toda la red local, basta definir la IP correspondiente a la red y la máscara de la sub-red. Por ejemplo, si se tiene una red donde los anfitriones tienen direcciones del segmento IP 172.16.100.0/28, se puede utilizar lo siguiente:

```
acl localnet src 172.16.100.0/28
```

También puede definirse una Lista de Control de Acceso especificando un archivo localizado en cualquier parte del disco duro y la cual contiene una lista de direcciones IP. Ejemplo:

```
acl permitidos src "/etc/squid/listas/permitidos"
```

El archivo /etc/squid/listas/permitidos tendría un contenido similar al siguiente:

```
172.16.100.1  
172.16.100.2  
172.16.100.3  
172.16.100.15  
172.16.100.16  
172.16.100.20  
172.16.100.40
```

Lo anterior estaría definiendo que la Lista de Control de Acceso denominada permitidos estaría compuesta por las direcciones IP incluidas en el archivo /etc/squid/listas/permitidos.

Reglas de control de acceso.

Estas definen si se permite o deniega acceso hacia Squid. Se aplican a las Listas de Control de Acceso. Deben colocarse en la sección de reglas de control de acceso definidas por el administrador, es decir, a partir de donde se localiza la siguiente leyenda:

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
```

La sintaxis básica de una regla de control de acceso es la siguiente:

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

Para desactivar la configuración predeterminada y poder utilizar una diferente, localice la línea que incluye `http_access allow localnet`

```
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP
networks
# from where browsing should be allowed
http_access allow localnet
http_access allow localhost
```

Deshabilite esta línea colocando una almohadilla (# al inicio de ésta:

```
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP
networks
# from where browsing should be allowed
# http_access allow localnet
http_access allow localhost
```

En el siguiente ejemplo se considera una regla que establece acceso permitido a Squid a la Lista de Control de Acceso denominada permitidos

```
http_access allow permitidos
```

También pueden definirse reglas valiéndose de la expresión !, la cual significa no. Pueden definirse, por ejemplo, dos listas de control de acceso, una denominada lista1 y otra denominada lista2, en la misma regla de control de acceso, en donde se asigna una expresión a una de estas. La siguiente establece que se permite el acceso a Squid a lo que comprenda lista1 excepto aquello que comprenda lista2:

```
http_access allow lista1 !lista2
```

Este tipo de reglas son útiles cuando se tiene un gran grupo de IP dentro de un rango de red al que se debe permitir acceso y otro grupo dentro de la misma red al que se debe denegar el acceso.

Administrar listas y reglas de control de acceso.

Una vez comprendido el funcionamiento de la Listas y las Regla de Control de Acceso, se procede a determinar cuáles utilizar para la configuración.

Caso 1

Considerando como ejemplo que se dispone de una red 172.16.100.0/28, si se desea definir toda la red local, se utilizaría la siguiente línea en la sección de Listas de Control de Acceso:

```
acl localnet src 172.16.100.0/28
```

Habiendo hecho lo anterior, la sección de listas de control de acceso debe quedar más o menos del siguiente modo:

Listas de control de acceso

```
# Recommended minimum configuration:
acl all src 0.0.0.0/0
acl manager proto cache_object
acl localhost src 127.0.0.1/8
acl localnet src 172.16.100.0/28
```

A continuación se procede a aplicar la regla de control de acceso:

```
http_access allow localnet
```

Habiendo hecho lo anterior, la zona de reglas de control de acceso debería quedar de modo similar al siguiente:

Reglas de control de acceso

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
http_access allow localhost
http_access allow localnet
http_access deny all
```

La regla `http_access allow localnet` permite el acceso a Squid a la Lista de Control de Acceso denominada `localnet`, la cual, en el siguiente ejemplo, está conformada por `172.16.100.0/28`. Esto significa que cualquier anfitrión desde `172.16.100.1` hasta `172.16.100.14` podrá acceder a Squid

Caso 2

Si sólo se desea permitir el acceso a Squid a ciertas direcciones IP de la red local, deberemos crear un archivo que contenga dicha lista. Genere el archivo `/etc/squid/listas/localnet`, dentro del cual se incluirán sólo aquellas direcciones IP que desea confirmen la Lista de Control de acceso. Ejemplo:

```
172.16.100.1
172.16.100.2
172.16.100.3
172.16.100.4
172.16.100.5
172.16.100.6
172.16.100.7
```

Denominaremos a esta lista de control de acceso como `localnet`:

```
acl localnet src "/etc/squid/listas/localnet"
```


Habiendo hecho lo anterior, la sección de listas de control de acceso debe quedar más o menos del siguiente modo:

Listas de control de acceso de una red local completa

```
# Recommended minimum configuration:
acl all src 0.0.0.0/0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl localnet src "/etc/squid/listas/localnet"
```

A continuación se procede a aplicar la regla de control de acceso:

```
http_access allow localnet
```

Habiendo hecho lo anterior, la zona de reglas de control de acceso debería quedar de modo similar al siguiente:

Reglas de control de acceso

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
http_access allow localhost
http_access allow localnet
http_access deny all
```

La regla `http_access allow localnet` permite el acceso a Squid a la Lista de Control de Acceso denominada `localnet`, la cual está conformada por las direcciones IP especificadas en el archivo `/etc/squid/listas/localnet`. Esto significa que cualquier anfitrión excluido del archivo `/etc/squid/listas/localnet` se le denegará el acceso a Squid.

Opción `cache_mgr`.

Esta opción es de carácter informativo. De modo predeterminado, si algo ocurre con el caché, como por ejemplo que muera el proceso, se enviará un mensaje de aviso a la cuenta `webmaster` del servidor. Puede especificarse una distinta si acaso se considera conveniente

```
cache_mgr joseperez@midominio.net
```

Opción `http_port`.

Esta opción es utilizada para indicar el puerto a través del cual escuchará peticiones Squid. EL valor predeterminado es 3128, es decir, Squid escuchará peticiones a través del puerto 3128/tcp.

```
http_port 3128
```

El puerto estándar designado para servidores de caché de Internet (webcache) es el puerto 8080.

```
http_port 8080
```

La opción permite establecer también si se quiere utilizar una dirección IP en particular. Esto añade mayor seguridad al servicio, pues si se tiene dos tarjetas de red, una con una dirección IP pública y otra con una dirección IP privada, se puede establecer que Squid solo permita conexiones desde la dirección IP privada.

```
http_port 192.168.80.1:8080
```

Si se necesita configurar un servidor proxy en modo transparente, solo es necesario añadir la opción `intercept`, misma que desde la versión 3.1 de Squid reemplaza a la opción `transparent`.

```
http_port 192.168.80.1:8080 intercept
```

Opción `cache_dir`.

Esta opción se utiliza para establecer que tamaño se desea que utilice Squid para almacenamiento de caché en el disco duro. De modo predeterminado Squid utilizará el formato `ufs` para crear en el directorio `/var/spool/squid` un caché de 100 MB, dividido en jerarquías de 16 directorios subordinados, hasta 256 niveles cada uno:

```
cache_dir ufs /var/spool/squid 100 16 256
```

Se puede incrementar el tamaño del caché hasta donde lo desee el administrador. Mientras más grande sea el caché, más objetos se almacenarán en éste y por lo tanto se consumirá menos el ancho de banda. La siguiente línea establece un caché de 2 GB:

```
cache_dir ufs /var/spool/squid 2048 16 256
```

El formato de cache `ufs` puede llegar a bloquear el proceso principal de Squid en operaciones de entrada/salida sobre el sistema de archivos cuando hay muchos clientes conectados. Para evitar que esto ocurra, se recomienda utilizar `aufs`, que utiliza el mismo formato de `ufs`, pero funciona de manera asincrónica, consiguiéndose un mejor desempeño.

```
cache_dir aufs /var/spool/squid 2048 16 256
```

Opción `maximum_object_size`.

Esta opción se utiliza para definir el tamaño máximo de los objetos en el caché. Se recomienda establecerla en escenarios con alta carga de trabajo, puesto que permite evitar desperdiciar recursos de sistema almacenando en el caché objetos de gran tamaño que probablemente sólo sean aprovechados por unos pocos usuarios, optimizando el uso del caché con objetos pequeños que de otro modo generarían una gran cantidad de peticiones hacia las redes públicas. En el siguiente ejemplo se establece un límite de 48 MB para los objetos del caché.

```
maximum_object_size 48 MB
```

Opciones `cache_swap_low` y `cache_swap_high`.

Es posible realizar una limpieza automática del caché de Squid cuando éste llegue a cierta capacidad. La opción `cache_swap_low` establece el porcentaje a partir del cual se comenzará a limpiar el cache. La opción `cache_swap_high` establece el porcentaje a partir del cual se comenzará a limpiar de manera agresiva el cache. En el siguiente ejemplo se establece que el cache se comienza a limpiar cuando alcanza el 90% y se comienza a limpiar de manera agresiva cuando alcanza el 95%.

```
cache_swap_low 90  
cache_swap_high 95
```

Lo anterior permite tener un caché saludable que se limpia automáticamente. Se recomienda utilizar estas opciones en escenarios con alta carga de trabajo.

Tabla 6.- Tabla carga de trabajo squid

LRU	Acrónimo de Least Recently Used, que traduce como Menos Recientemente Utilizado. En este algoritmo los objetos que fueron accedidos hace mucho tiempo, son eliminados primero y manteniendo siempre en el caché a los objetos más recientemente solicitados. Ésta política es la utilizada por Squid de modo predeterminado.
LFUDA	Acrónimo de Least Frequently Used with Dynamic Aging, que se traduce como Menos Frecuentemente Utilizado con Envejecimiento Dinámico. En este algoritmo los objetos más solicitados permanecen en el caché sin importar su tamaño optimizando la eficiencia (hit rate) por octetos (Bytes) a expensas de la eficiencia misma, de modo que un objeto grande que se solicite con mayor frecuencia impedirá que se pueda hacer caché de objetos pequeños que se soliciten con menor frecuencia.
GDSF	Acrónimo de GreedyDual Size Frequency, que se traduce como Frecuencia de tamaño <i>GreedyDual</i> (<i>codicioso dual</i>), que es el algoritmo sobre el cual se basa GDSF. Optimiza la eficiencia (hit rate) por objeto manteniendo en el caché los objetos pequeños más frecuentemente solicitados de modo que hay mejores posibilidades de lograr respuesta a una solicitud (hit). Tiene una eficiencia por octetos (Bytes) menor que el algoritmo LFUDA debido a que descarta del caché objetos grandes que sean solicitado con frecuencia.

El algoritmo recomendado y que ha demostrado mejor desempeño en escenarios de alta carga de trabajo es LFUDA.

```
cache_replacement_policy heap LFUDA
```

Opción cache_mem.

La opción cache_mem establece la cantidad ideal de memoria para lo siguiente:

- Objetos en tránsito.
- Objetos frecuentemente utilizados (Hot).
- Objetos negativamente almacenados en el caché.

Los datos de estos objetos se almacenan en bloques de 4 Kb. La opción cache_mem especifica un límite máximo en el tamaño total de bloques acomodados, donde los objetos en tránsito tienen mayor prioridad. Sin embargo los objetos frecuentemente utilizados (Hot) y aquellos negativamente almacenados en el caché, podrán utilizar la memoria sin utilizar hasta que esta sea requerida. De ser necesario, si un objeto en tránsito es mayor a la cantidad de memoria especificada, Squid excederá lo que sea necesario para satisfacer la petición.

De modo predeterminado, desde la versión 3.1 de Squid, se establecen 256 MB, que es más que suficiente para las necesidades de redes de área local con pocos

anfitriones. Puede especificar una cantidad menor para obtener un mejor rendimiento, pues conviene utilizar la memoria disponible para hacer cache en memoria de muchos objetos pequeños que son frecuentemente visitados, que hacer cache de unos pocos objetos grandes que sólo unos pocos usuarios aprovecharán. En el siguiente ejemplo se establecen 48 MB como límite de tamaño para los objetos en tránsito:

```
cache_mem 48 MB
```

Estableciendo el idioma de los mensajes mostrados por squid hacia el usuario.

Squid incluye traducción a distintos idiomas de las distintas páginas de error e informativas que son desplegadas en un momento dado durante su operación. Dichas traducciones se pueden encontrar en `/usr/share/squid/errors/`. Desde la versión 3.0 de Squid, el idioma se detecta automáticamente a partir del navegador utilizado por el usuario. Es innecesario modificar opción alguno, salvo que se haya personalizado los mensajes, en cuyo caso conviene utilizar una ruta distinta a la del idioma utilizado para evitar se sobre-escriban los archivos después de actualizar el sistema.

Iniciando, reiniciando y añadiendo el servicio al arranque del sistema.

Una vez terminada la configuración, para iniciar por primera vez Squid ejecute:

```
service squid start
```

Si necesita volver a cargar la configuración para probar cambios realizados, sin detener el servicio, ejecute:

```
service squid reload
```

Si necesita reiniciar para probar cambios hechos en la configuración, considerando que este proceso puede llegar a demorar algunos minutos, ejecute:

```
service squid restart
```

Para que Squid inicie de manera automática junto con el sistema, ejecute:

```
chkconfig squid on
```

Lo anterior habilitará el servicio squid en todos los niveles de ejecución.

Depuración de errores

Cualquier error al inicio de Squid sólo significa que hubo errores de sintaxis, errores de dedo o bien se están citando incorrectamente las rutas hacia los archivos de las Listas de Control de Acceso.

Puede realizar diagnóstico de problemas indicándole a Squid que vuelva a leer configuración, lo cual devolverá los errores que existan en el archivo `/etc/squid/squid.conf`.

```
service squid reload
```

Cuando se trata de errores graves que impiden iniciar el servicio, puede examinarse el contenido del archivo `/var/log/squid/squid.out` con el mandato `less`, `more` o cualquier otro visor de texto:

```
tail -80 /var/log/squid/squid.out
```

Modificaciones necesarias en el muro cortafuegos.

Si se utiliza un cortafuegos con políticas estrictas, como por ejemplo Shorewall, es necesario abrir el puerto 8080 por TCP (webcache), si se eligió utilizar el puerto 8080 en lugar del 3128.

La regla para el archivo `/etc/shorewall/rules` de Shorewall, que sólo permitirá el acceso hacia Squid desde la zona de red de área local, correspondería a algo similar a lo siguiente:

```
#ACTION SOURCE DEST PROTO DEST SOURCE
# PORT PORT (S) 1
ACCEPT loc fw tcp 8080
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

Para aplicar los cambios en Shorewall, ejecute:

```
service shorewall restart
```

Re-direccionamiento de peticiones a través de la opción REDIRECT en shorewall.

La acción REDIRECT en Shorewall permite redirigir peticiones hacia protocolo HTTP para hacerlas pasar a través de Squid. En el siguiente ejemplo las peticiones hechas desde la zona que corresponde a la red local serán redirigidas hacia el puerto 8080 del cortafuego, en donde está configurado Squid configurado como Servidor Proxy (Proxy) transparente.

```
#ACTION SOURCE DEST PROTO DEST SOURCE
```

```

#
ACCEPT loc fw tcp 8080
REDIRECT loc 8080 tcp 80
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE

```

Exclusión de sitios en Shorewall.

En el caso de sitios que se quiera excluir de ser utilizados con Squid, es decir, sitios problemáticos, se puede configurar en Shorewall que el acceso sea directo, con una configuración similar a la del siguiente ejemplo, donde se excluye de pasar por Squid las peticiones dirigidas a las redes 201.144.108.0/24 (IMSS.gob.mx) y 200.33.74.0/24 (SAT.gob.mx) y se abre el paso directo desde la red local hacia esta red:

```

#ACTION SOURCE DEST PROTO DEST SOURCE
# PORT PORT(S)1
ACCEPT loc fw tcp 8080
REDIRECT loc 8080 tcp 80 -
!201.144.108.0/24,200.33.74.0/24
ACCEPT loc net:201.144.108.0/24 all
ACCEPT loc net:200.33.74.0/24 all
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE

```

Re-direccionamiento de peticiones a través de iptables.

Bajo ciertas circunstancias, se requerirá tener salida transparente hacia Internet para ciertos servicios, pero al mismo tiempo se necesitará re-direccionar peticiones hacia servicio HTTP para pasar a través del el puerto donde escucha peticiones Squid, como proxy en modo transparente, es decir el puerto 8080/tcp, de modo que se impida la salida hacia alguna hacia servidores HTTP en el exterior sin que ésta pase antes por Squid. Ningún proxy conocido puede funcionar en modo transparente para los protocolos HTTPS, FTP, GOPHER ni WAIS, por lo que dichos protocolos tendrán que ser filtrados a través del NAT.

El re-direccionamiento se hace a través de iptables. Considerando para este ejemplo que la red local se accede a través de una interfaz eth1, el siguiente esquema ejemplifica un re-direccionamiento:

```

iptables -A INPUT -m state --state NEW -m tcp -p tcp \
-i eth1 --dport 8080 -j ACCEPT
iptables -t nat -A PREROUTING -i eth1 -p tcp \
--dport 80 -j REDIRECT --to-port 8080
service iptables save

```

2.7 IPTables

Centos tiene una estructura interna de cortafuegos extremadamente poderosa, comúnmente nos referimos a ella como iptables pero más correctamente es iptables/netfilter. Iptables es el módulo para el espacio de usuario, la parte con la cual el usuario, interactúa en la línea de comandos para entrar las reglas del corta fuegos en las tablas predefinidas. Netfilter es el módulo del núcleo, construido dentro del núcleo. Actualmente este es el que se encarga del filtrado.

Existen varias presentaciones GUI para iptables que le permiten a los usuarios adicionar o definir reglas basadas en un punto y con el clic del usuarios en la interface, pero estos a menudo carecen de la flexibilidad de usar la línea de comando y limitan la comprensión de los usuarios de lo que está pasando realmente.

Antes de que podamos enfrentarnos a iptables necesitamos tener al menos una comprensión básica de su forma de trabajo. Iptables usa el concepto de direcciones ip, protocolos (tcp, udp, icmp) y puertos.

Iptables ubica las reglas dentro de cadenas predefinidas (INPUT, OUTPUT y FORWARD) que son comprobadas contra cualquier tráfico de red (paquetes IP) relevantes para esas cadenas y una decisión es tomada sobre qué hacer con cada paquete basado en el significado de esas reglas, por ejemplo aceptar o rechazar el paquete. Estas acciones son referidas como objetivos (targets), de las cuales las dos más usadas son DROP para rechazar un paquete o ACCEPT para permitir el paquete.

Cadenas

Existen tres cadenas predefinidas en la tabla de filtrado para las cuales podemos adicionar reglas para procesar los paquetes IP que pasan a través de las cadenas. Estas cadenas son:

- INPUT - Todos los paquetes dirigidos a la computadora anfitrión.
- OUTPUT - Todos los paquetes originados en la computadora anfitrión.
- FORWARD - Todos los paquetes que no son originados o dirigidos a la computadora anfitrión, pero pasan a través (enrutados) de la computadora anfitrión. Esta cadena es usada si usted está usando su computadora como un enrutado

La mayor parte del tiempo, estaremos tratando con la cadena INPUT para filtrar los paquetes que entran a nuestra computadora.

Las reglas son adicionadas a la lista de cada cadena. Un paquete es comprobado contra cada regla en turno, comenzando por arriba. Si el paquete coincide con esa regla, entonces una acción es realizada, ej. Aceptar (ACCEPT), o rechazar (DROP) el paquete. Una vez que la regla ha coincidido y una acción realizada, entonces el paquete es procesado de acuerdo al resultado de la regla y no es procesada por reglas posteriores en la cadena. Si un paquete pasa todas las reglas en la cadena hasta abajo y llega al final sin haber coincidido con regla alguna, entonces es utilizada la acción por defecto para esa cadena. Esto se refiere a la política por defecto y puede estar fijada en aceptar o en rechazar el paquete.

El concepto de la política predeterminada dentro de las cadenas permite dos posibilidades fundamentales que debemos considerar primero, antes de decidir cómo vamos a organizar el cortafuego.

1. Podemos fijar una política predeterminada para rechazar todos los paquetes y entonces adicionar reglas para permitir (ACCEPT) paquetes específicos que pueden venir de direcciones IP confiables o para algunos puertos en los cuales tenemos servicios corriendo, tales como bittorrent, servidor FTP, servidor Web, servidor de ficheros Samba

2. Podemos fijar una política para permitir todos los paquetes y entonces adicionar reglas que rechacen paquetes específicos que pueden venir de direcciones o rangos IP engorrosas o para algunos puertos en los cuales tenemos servicios privados o ningún servicio corriendo.

Generalmente, la opción 1 de arriba es usada para la cadena INPUT donde controlamos a que queremos permitir acceso en nuestra computadora y la opción 2 sería usada en la cadena OUTPUT donde generalmente confiamos en el tráfico que está saliendo de (originado en) nuestra computadora.

El trabajo con iptables desde la línea de comando requiere los privilegios de root, así que usted necesitará convertirse en root para la mayoría de las cosas que estaremos haciendo.

Iptables debe estar instalado por defecto en todas las instalaciones de Centos 3.x, 4.x y 5.x. Puede comprobar si iptables está instalado en su sistema con:

```
$ rpm -q iptables
iptables-1.3.5-1.2.1
```

Para ver si iptables está corriendo, podemos comprobar que los módulos de iptables están cargados y usar la opción -L para inspeccionar las reglas que que están cargadas actualmente

```
# lsmod | grep ip_tables
ip_tables                29288  1 iptable_filter
x_tables                 29192  6
ip6t_REJECT,ip6_tables,ipt_REJECT,xt_state,xt_tcpudp,ip_tables
# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
RH-Firewall-1-INPUT  all  --  anywhere              anywhere
Chain FORWARD (policy ACCEPT)
target     prot opt source                destination
RH-Firewall-1-INPUT  all  --  anywhere              anywhere
Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
Chain RH-Firewall-1-INPUT (2 references)
target     prot opt source                destination
ACCEPT     all  --  anywhere              anywhere
ACCEPT     icmp --  anywhere              anywhere
icmp any
ACCEPT     esp  --  anywhere              anywhere
ACCEPT     ah   --  anywhere              anywhere
ACCEPT     udp  --  anywhere              224.0.0.251
udp dpt:mdns
ACCEPT     udp  --  anywhere              anywhere
udp dpt:ipp
ACCEPT     tcp  --  anywhere              anywhere
tcp dpt:ipp
ACCEPT     all  --  anywhere              anywhere
state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere              anywhere
state NEW tcp dpt:ssh
REJECT     all  --  anywhere              anywhere
reject-with icmp-host-prohibited
```

Arriba vemos el conjunto de reglas predeterminadas además del acceso al servicio SSH.

Si iptables no está corriendo puede habilitarlo ejecutando:

```
# system-config-securitylevel
```

Usaremos un ejemplo que nos permitirá examinar los comandos de iptables. En este primer ejemplo crearemos un conjunto de reglas muy simples para configurar un cortafuegos del tipo Stateful Packet Inspection (SPI) que permitirá todas las conexiones salientes pero bloqueará todas las conexiones entrantes indeseada.

```
# iptables -F
# iptables -P INPUT DROP
# iptables -P FORWARD DROP
# iptables -P OUTPUT ACCEPT
# iptables -A INPUT -i lo -j ACCEPT
# iptables -A INPUT -m state --state ESTABLISHED,RELATED -j
ACCEPT
# iptables -L -v
```

Lo cual debe darle la siguiente salida:

```
Chain INPUT (policy DROP 0 packets, 0 bytes)
  pkts bytes target      prot opt in      out     source
destination
    0    0 ACCEPT      all  --  lo     any     anywhere
anywhere
    0    0 ACCEPT      all  --  any    any     anywhere
anywhere                state RELATED,ESTABLISHED
Chain FORWARD (policy DROP 0 packets, 0 bytes)
  pkts bytes target      prot opt in      out     source
destination
Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target      prot opt in      out     source
destination
```

Ahora vamos a ver cada uno de los siete comandos de arriba y comprender exactamente lo que acabamos de hacer:

Iptables -F: Lo primero que hemos hecho es usar la opción -F para eliminar las reglas una por una, de forma tal que comencemos con un estado limpio en el cual comenzar a adicionar reglas nuevas.

Iptables -P INPUT DROP: La opción -P fija la política por defecto en la cadena especificada. Así que aquí estamos fijando a DROP como la política por defecto en la cadena INPUT. Esto quiere decir que si un paquete entrante no coincide una de las reglas siguientes será descartada.

Iptables -P FORWARD DROP : De la misma forma, aquí estamos fijando a DROP la política por defecto para la cadena FORWARD porque no estamos usando nuestra computadora como un enrutador así que no deberían estar pasando paquetes a través de nuestra computadora.

Iptables -P OUTPUT ACCEPT: y finalmente fijamos a ACCEPT la política por defecto para la cadena OUTPUT porque queremos permitir todo el tráfico saliente (porque confiamos en nuestros usuarios).

Iptables -A INPUT -i lo -j ACCEPT: Ahora es el momento de comenzar a adicionar algunas reglas. Usamos la opción -A para anexar (o adicionar) una regla a la cadena específica, en este caso la cadena INPUT. Luego usamos la opción -i (interface) para especificar los paquetes que coinciden o están destinados a la interface lo (localhost, 127.0.0.1) y finalmente -j (jump) para saltar al objetivo de acción para el paquete que coincide con la regla, en este caso ACCEPT. Así, esta regla permitirá que todos los paquetes entrantes con destino a la interface localhost sean aceptados. Esto generalmente requiere que las aplicaciones de software sean capaces de comunicarse con el adaptador localhost.

Iptables -A INPUT -m state --state ESTABLISHED, RELATED -j ACCEPT: Esta es la regla que hace la mayor parte del trabajo y nuevamente estamos adicionando (-A) a la cadena INPUT. Aquí estamos usando la opción -m para cargar un módulo (state). El módulo estado está disponible para examinar el estado de un paquete y determinar si este es nuevo (NEW), establecido (ESTABLISHED) o relacionado (RELATED). NEW se refiere a los paquetes entrantes que son conexiones entrantes nuevas que fueron iniciadas por el sistema anfitrión. ESTABLISHED y RELATED se refieren a los paquetes entrantes que son parte de una conexión ya establecida o relacionada a la conexión ya establecida.

Iptables -L -v : Listar (-L) las reglas que acabamos de adicionar para comprobar que han sido cargadas correctamente.

Finalmente, lo último que necesitamos hacer es salvar las reglas para que la próxima vez que reiniciemos la computadora nuestras reglas sean recargadas automáticamente:

```
# /sbin/service iptables save
```

Esto ejecuta el script init de iptables el cual corre /sbin/iptables-save y escribe la configuración actual de iptables a /etc/sysconfig/iptables. Con el reinicio, el script init de iptables vuelve a aplicar las reglas salvadas en /etc/sysconfig/iptables usando el comando /sbin/iptables-restore.

Obviamente escribir estos comandos directamente en el shell puede ser tedioso, así que la forma más fácil de trabajar con iptables es crear un script simple para hacer todo esto por usted. Los comandos de arriba pueden ser entrados en su editor de texto favorito y ser salvado como myfirewall, por ejemplo:

```
#!/bin/bash
#
# iptables example configuration script
#
# Flush all current rules from iptables
#
iptables -F
#
# Set default policies for INPUT, FORWARD and OUTPUT chains
#
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
#
# Set access for localhost
#
iptables -A INPUT -i lo -j ACCEPT
#
# Accept packets belonging to established and related
connections
#
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j
ACCEPT
#
# Save settings
#
/sbin/service iptables save
#
# List rules
#
iptables -L -v
```

Ahora haga el script ejecutable:

```
# chmod +x myfirewall
```

Ahora podemos editar simplemente nuestro script y correrlo desde el shell con el comando siguiente:

```
# ./myfirewall
```

Interfaces

En nuestro ejemplo anterior vimos cómo podemos aceptar todos los paquetes entrantes a una interface particular, en este caso la interface localhost:

```
iptables -A INPUT -i lo -j ACCEPT
```

Supongamos que tenemos dos interfaces separadas, eth0 la cual es nuestra conexión LAN interna y ppp0 dialup modem (o talvés eth1 para una nic) la cual es nuestra conexión externa a internet. Podemos necesitar todos los paquetes entrantes a nuestra LAN interna pero continuar filtrando paquetes entrantes hacia nuestra conexión externa de internet. Podríamos hacer lo siguiente:

```
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -i eth0 -j ACCEPT
```

Pero tenga mucho cuidado - si vamos a permitir todos esos paquetes para nuestra interface externa de internet (por ejemplo ppp0 dialup modem):

```
iptables -A INPUT -i ppp0 -j ACCEPT
```

Efectivamente, con esto tendríamos deshabilitado nuestro cortafuegos!

Direcciones IP

Abrir una interface completa a los paquetes entrantes puede no ser lo suficientemente restrictivo y usted puede necesitar más control como para decir que permitir y que rechazar. Vamos a suponer que tenemos una pequeña red de computadoras que usan la sub red privada 192.168.0.x. Podemos abrir nuestros cortafuegos para los paquetes entrantes desde una sola dirección IP en la cual confiamos (por ejemplo 192.168.0.4):

```
# Accept packets from trusted IP addresses
iptables -A INPUT -s 192.168.0.4 -j ACCEPT # change the IP
address as appropriate
```

Desglosando este comando, primero anexamos (-A) una regla para la cadena INPUT que acepta (ACCEPT) todos los paquetes para la dirección IP de origen (-s) 192.168.0.4. (Observe como podemos utilizar el símbolo # para adicionar comentarios en línea que permitan documentar nuestro script. Cualquier cosa que ponga después del # será ignorado y tratado como un comentario).

Obviamente si queremos permitir paquetes entrantes desde un rango de direcciones podemos aplicar una regla para cada dirección IP que confiamos y eso debería funcionar bien. Pero si tenemos muchas de ellas, esto puede hacerse más fácil si adicionamos el rango de direcciones IP en una sola línea. Para hacer esto podemos utilizar una máscara de red o la notación estándar de slash para especificar un rango de direcciones IP. Por ejemplo, si queremos abrir nuestros cortafuegos para

todos los paquetes entrantes desde el rango 192.168.0.x (donde x = de 1 a 254), podemos utilizar cualquiera de los métodos siguientes:

```
# Accept packets from trusted IP addresses
iptables -A INPUT -s 192.168.0.0/24 -j ACCEPT # using
standard slash notation

iptables -A INPUT -s 192.168.0.0/255.255.255.0 -j ACCEPT #
using a subnet mask
```

Finalmente, de la misma forma que filtramos contra una sola dirección IP, podemos también hacer coincidir la dirección MAC de un dispositivo dado. Para hacer esto, necesitamos cargar el módulo (mac) que permite filtrar contra direcciones mac. Anteriormente vimos un ejemplo del uso de módulos para extender la funcionalidad de iptables cuando usamos el módulo estado para hacer coincidir los paquetes ESTABLISHED y RELATED. Aquí usamos el módulo mac para comprobar la dirección mac de un origen de paquetes, además de su dirección IP:

```
# Accept packets from trusted IP addresses
iptables -A INPUT -s 192.168.0.4 -m mac --mac-source
00:50:8D:FD:E6:32 -j ACCEPT
```

Primero usamos -m mac para cargar el módulo mac y luego usamos --mac-source para especificar la dirección mac de la dirección IP origen (192.168.0.4). Usted necesitará encontrar la dirección mac de cada dispositivo ethernet contra el cual esté filtrando. Ejecutando ifconfig (o iwconfig para los dispositivos inalámbricos) como root le mostrará la dirección mac.

Esto puede ser útil en la prevención de la falsificación de direcciones IP originales, pues permitirá a cualquier paquete que sea genuinamente originado de la dirección 192.168.0.4 (con la dirección mac 00:50:8D:FD:E6:32) pero rechazará cualquier paquete que sea falsificado para mostrarse como originario de esa dirección IP.

Puertos y Protocolos

Arriba hemos visto como adicionar reglas a nuestro cortafuegos para filtrar contra paquetes que coinciden con una interface particular o una dirección IP de origen. Esto permite un acceso completo a través de nuestro cortafuegos para algunos orígenes confiables (PCs anfitriones). Ahora veremos cómo podemos filtrar contra protocolos y puertos para refinar cuales paquetes permitimos entrar y cuales rechazamos.

Antes que comencemos, necesitamos saber que protocolo y número de puerto un servicio determinado usa. Por ejemplo, veamos el caso de bittorrent. Bittorrent usa el protocolo tcp en el puerto 6881, así que necesitamos permitir todos los paquetes tcp que tengan como destino el puerto 6881 (en nuestra computadora):

```
# Accept tcp packets on destination port 6881 (bittorrent)
iptables -A INPUT -p tcp --dport 6881 -j ACCEPT
```

Podemos extender lo de arriba para incluir un rango de puertos, por ejemplo, para permitir todos los paquetes tcp en el rango de 6881 a 6890:

```
# Accept tcp packets on destination ports 6881-6890
iptables -A INPUT -p tcp --dport 6881:6890 -j ACCEPT
```

Poniendo todo junto

Ahora que hemos visto las bases, podemos comenzar a combinar estas reglas.

Un servicio popular de UNIX/Linux es el servicio de shell seguro (SSH) que permite hacer login remoto. Por defecto SSH usa el puerto 22 y el protocolo tcp. Así, si queremos permitir logins remotos, necesitamos permitir las conexiones tcp entrantes al puerto 22:

```
# Accept tcp packets on destination port 22 (SSH)
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

Esto abrirá el puerto 22 (SSH) para todas las conexiones tcp lo cual es una potencial brecha de seguridad pues los hackers pueden intentar el cracking por fuerza bruta en cuentas con contraseñas débiles. Sin embargo, si sabemos la dirección IP de la computadora remota en la cual confiamos, esa que será usada para hacer login SSH, podemos limitar el acceso solo esta dirección IP de origen. Por ejemplo, si deseamos abrir solamente el acceso SSH a nuestra LAN privada (192.168.0.x), podemos limitar el acceso solo a este rango de direcciones IP:

```
# Accept tcp packets on destination port 22 (SSH) from private
LAN
iptables -A INPUT -p tcp -s 192.168.0.0/24 --dport 22 -j
ACCEPT
```


2.8 Webmin

El uso del filtrado por IP de origen nos permite abrir seguramente el acceso SSH en el puerto 22 solo a las direcciones IP en las que confiamos. Por ejemplo, podemos usar este método para permitir logins remotos entre las computadoras del trabajo y las del hogar. Para el resto de las direcciones IP, el puerto (y servicio) aparecería cerrado como si el servicio estuviese deshabilitado. De esta forma los hackers que usan los métodos de escaneo de puertos posiblemente nos pasen por un lado.

Webmin es una herramienta de configuración de sistemas accesible vía web para GNU/Linux, OpenSolaris y otros sistemas Unix. Con él se pueden configurar aspectos internos de muchos sistemas operativos, como usuarios, cuotas de espacio, servicios, archivos de configuración, apagado del equipo, etcétera, así como modificar y controlar muchas aplicaciones libres, como el servidor web Apache, PHP, MySQL, DNS, Samba, DHCP, entre otros.

Webmin está escrito en Perl, versión 5, ejecutándose como su propio proceso y servidor web. Por defecto se comunica mediante TCP a través del puerto 10000, y puede ser configurado para usar SSL si OpenSSL está instalado con módulos de Perl adicionales requeridos.

Los pasos para instalar webmin en CentOS 6 Minimal son los siguientes:

1. Añadimos los repositorios de webmin al fichero sources.list:

```
nano /etc/yum.repos.d/webmin.repo
[Webmin]
name=Webmin                Distribution                Neutral
baseurl=http://download.webmin.com/download/yum
mirrorlist=http://download.webmin.com/download/yum/mirrorlist
enabled=1
```

2. Instalamos la clave GPG

```
rpm --import http://www.webmin.com/jcameron-key.asc
```

3. Y por último instalamos webmin:

```
yum install webmin
```

```
centOS Minimal (Instantánea 1) [Corriendo] - Oracle VM VirtualBox
extras | 3.5 kB | 00:00
updates | 3.5 kB | 00:00
updates/primary_db | 2.8 MB | 00:15
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package webmin.noarch 0:1.580-1 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
webmin noarch 1.580-1 Webmin 16 M
=====
Transaction Summary
=====
Install 1 Package(s)
Upgrade 0 Package(s)

Total download size: 16 M
Installed size: 16 M
Is this ok [y/N]: _
```

Figura 2.- Centos Terminal

Ahora para acceder a webmin usamos la siguiente dirección <https://DireccionIP:10000>

Nos saldrá una ventana como esta y tendremos que introducir el usuario root y nuestra contraseña

2.9 Estándares

2.9.1 Linux standard base

Linux Standard Base ha sido estudiada, verificada, y nombrada por la ISO/IEC como estándar ISO. La Base Standard para Linux es una recopilación de pruebas y casos de éxito manejados por el Free Standards Group para lograr la compatibilidad de las distribuciones en la instalación y ejecución de servicios y aplicaciones. Por lo tanto el sistema operativo Linux y sus aplicaciones está siendo reconocido por los organismos internacionales de estandarización.

Este estándar lo realizan las empresas creadoras de distribuciones Linux para que éstas sean totalmente compatibles a nivel de instalación y ejecución. De esta forma, si tenemos una aplicación preparada para ejecutarse sobre una distribución que cumpla con la recomendación LSB, será indiferente el sistema operativo linux sobre la cual corra siempre y cuando esta cumpla con LSB. Existen diferentes

versiones de la especificación LSB, por lo que un determinado producto puede cumplir con las condiciones establecidas en LSB 1.3, 2.0 o 3.0 por ejemplo.

El éxito de un sistema operativo está directamente vinculado con el número de aplicaciones que se ejecutan y la calidad con la que corren sobre él Linux y sus diferentes distribuciones, de cualquier modo, presenta a desarrolladores individuales con un conjunto único de retos: diferentes distribuciones de Linux hacen uso de diferentes versiones de bibliotecas, archivos importantes almacenados en diferentes sitios, y así muchas otras variantes.

La LSB fue creada para resolver estos problemas y reducir los costos globales de apoyo a la plataforma libre Linux. Al minimizar las diferencias entre las distintas distribuciones individuales de Linux, LSB reduce los costos relacionados con las aplicaciones de portar diferentes modos de distribuciones.

La LSB está basada en la Especificación POSIX, la Especificación Única de UNIX (Single UNIX Specification) y en varios otros estándares abiertos, aunque extiende éstos en ciertas áreas.

El objetivo de la LSB es desarrollar y promover un conjunto de estándares que aumentarán la compatibilidad entre las distribuciones de Linux y permitirán que las aplicaciones puedan ser ejecutadas en cualquier sistema Linux. Además, la LSB ayudará a coordinar esfuerzos para poder reclutar productores y proveedores de programas que creen productos nuevos para Linux o adaptaciones de aplicaciones existentes.

Mediante un proceso de certificación es posible obtener la conformidad a la LSB en una aplicación. Esta certificación la dirige el Open Group en conjunto con el Free Standards Group (Grupo de Estándares Libres).

Por ejemplo, la LSB especifica librerías estándar, un conjunto de utilerías que mejoran el estándar POSIX, la estructura del sistema de archivos, los niveles de ejecución, y varias extensiones al modo gráfico X Window.

Hasta ahora, LSB consistía en una especificación única, pero el FSG anuncio la posible creación de dos tipos de estándares LSB, uno para los servidores y otro para los desktop.

Dividiendo el estándar en dos, el FSG consigue que este pueda abarcar una amplia gama de tecnologías, pues no se le da el mismo uso a un servidor que a un

computador de escritorio. Un ejemplo bastante simple, pero didáctico es que puede no ser tan importante el reconocimiento de hardware USB en un servidor al que seguramente nunca le añadiremos un disco duro externo o similar, como en un computador de escritorio, en el cual estaremos continuamente conectando, pen drives USB, cámaras de fotos y de video. De esta forma, dividir los estándares puede ser útil a cada grupo de desarrollo para centrarse solamente en lo que necesitan para cada definición de estándar.

También hay interés en incluir en el estándar LSB el entorno de ejecución de aplicaciones Java (JRE, Java Runtime Environment), aunque esto puede representar una tarea más difícil ya que se requiere de licencia de Sun Microsystems.

2.9.2 Estándares de protocolos de Internet RFC

RFC de TCP/IP

La familia de protocolos de Internet está todavía evolucionando mediante el mecanismo de Petición de Comentario (RFC). Los nuevos protocolos (la mayoría de los protocolos de aplicación) los han diseñado e implementado investigadores y científicos y han sido expuestos a la comunidad de Internet en forma de petición de comentario RFC. El Internet Architecture Board (IAB) supervisa el mecanismo RFC. La mayor fuente de RFC es la Internet Engineering Task Force (IETF). Sin embargo, cualquiera puede proponer una petición de comentario RFC al editor de RFC. Existe una serie de reglas que los autores de RFC deben seguir para que se acepten. Estas reglas se describen en un RFC (RFC 1543) que indica cómo considerar una propuesta para un RFC.

Una vez que se ha publicado un RFC, todas las revisiones y suplementos se publicarán como nuevos RFCs. Un nuevo RFC que revise o reemplace uno existente se dice "actualizado" u "obsoleto". El RFC existente se dice "actualizado por" u "obsoleto por" el nuevo. Por ejemplo el RFC 1521 que describe el protocolo MIME es una "segunda edición", siendo una revisión del RFC 1341 y RFC 1590 es una corrección al RFC 1521. RFC 1521 es por tanto etiquetada como "Obsoleto RFC 1341; Actualizado por RFC 1590". Por consiguiente, no existe confusión alguna de si dos personas se están refiriendo a versiones diferentes de un RFC, dado que no hay nunca versiones diferentes.

Algunos RFCs se describen como documentos de información que otros describen como protocolos de Internet. El Internet Architecture Board (IAB) mantiene

una lista de los RFCs que describen la familia de protocolos. Cada uno de estos tiene asignado un estado y un status.

Un protocolo de Internet puede tener uno de los siguientes estados:

Estándar

El IAB ha establecido esto como un protocolo oficial para Internet. Se separan en dos grupos:

- Protocolo IP y citados, protocolos aplicados enteramente a Internet.
- Protocolos específicos de red, generalmente especificaciones de cómo hacer IP sobre tipos particulares de redes.

Estándar Borrador

El IAB está considerando activamente como un posible protocolo estándar. El

IAB somete los comentarios y resultados de pruebas. Existe una posibilidad que si cambia será hecho un protocolo preliminar antes de que esto se haga un estándar.

Estándar Propuesto

Estos son protocolos propuestos que debe considerar IAB para su estandarización en el futuro. Son deseables implementaciones y comprobaciones de varios grupos. La revisión del protocolo es probable.

Experimental

Un sistema no debería implementar un protocolo experimental a no ser que esté participando en el experimento y ha coordinado su uso del protocolo con el desarrollador del protocolo.

Informativo

Los protocolos desarrollados por otras organizaciones, o vendedores, o que están por otras razones fuera del alcance de IAB deben publicarse como RFCs por conveniencia de la comunidad de Internet como protocolos informativos. Tales protocolos pueden en algunos casos también estar recomendados para uso en Internet por IAB.

RFC de Internet

El estándar propuesto, el borrador y los protocolos estándar se describen como constituyentes del Internet Standards Track. El track estándar lo controla el Grupo de

Dirección de Ingenieros de Internet (IESG) del IETF. Cuando un protocolo alcanza el estado de estándar se le asigna un número estándar (STD). El propósito de los números STD es indicar claramente qué RFCs describen los estándares de Internet. Los números STD referencian múltiples RFCs cuando la especificación de un estándar se divide en múltiples documentos. No como con los RFCs, donde el número se refiere a un documento específico, los números STD no cambian cuando un estándar se actualiza. Los números STD, sin embargo, no tienen número de versión dado que todas las actualizaciones se realizan vía RFCs y los números de RFC son únicos. De este modo, para especificar sin ambigüedad qué versión de un estándar único se está refiriendo, se pondría de manifiesto el número estándar y todos los RFCs que incluye. Por ejemplo, el Sistema de Nombres de Dominio (DNS) es STD 13 y se describe en los RFCs 1034 y 1035. Para referenciar el estándar se podría utilizar algo como "STD-13/RFC-1034/RFC-1035". Para una descripción de los Procesos Estándares, ver RFC 1602 -- Los Procesos Estándares de Internet - Revisión 2.

Para algunos estándares RFCs la categoría de status no siempre contiene suficiente información útil. Por lo tanto, se complementa, notablemente por protocolos de enrutamiento por un applicability statement que se da en STD 1 o en un RFC separado.

Cuatro estándares de Internet tienen una importancia particular: STD 1 Estándares de Protocolos Oficiales de Internet

Este estándar da el estado y status de cada protocolo o estándar de Internet y define los significados atribuidos para cada estado o status diferente. Emitió aproximadamente una cuarta parte el IAB. Cuando se escribió este estándar fue el RFC 1780 (Marzo de 1995).

STD 2 - Números Asignados en Internet

Este estándar lista actualmente números asignados y otros parámetros de protocolos en la familia de protocolos de Internet. Lo emitió la Autoridad de Números Asignados de Internet (IANA). La edición cuando se escribió fue el RFC 1700 (Octubre de 1994).

STD 3 - Requerimientos del Host

Este estándar define los requerimientos para el software de host de Internet (a menudo con referencia a los RFCs relevantes). El estándar viene en dos partes: RFC

1122 - Requerimientos para hosts de Internet - capas de comunicaciones y RFC 1123
- Requerimientos para hosts de Internet- aplicación y ayuda.

STD 4 - Requerimientos de Pasarela

Este estándar define los requerimientos para el software de pasarela de Internet (router). Es el RFC 1009.

Para Tu Información (FYI) Un determinado número de RFCs que tienen la intención de ser interesantes a los usuarios de Internet se clasifican como documentos Para Tu Información (FYI). Contienen frecuentemente información introductoria u otro tipo de información útil. Como los números de STD, un número de FYI no cambia cuando se emite la revisión de un RFC. Distintos STDs, FYIs corresponden a un único documento RFC. Por ejemplo, FYI 4 -- FYI sobre Preguntas y Respuestas comunes "Nuevo Usuario de Internet" está actualmente en su cuarta edición. Los números de RFC son 1177, 1206, 1325 y 1594.

2.9.3 Principales protocolos de internet

Para dar una idea sobre la importancia de los principales protocolos, se listan algunos de ellos junto con su estado actual y número de STD donde es aplicable en la tabla que se muestra abajo. La lista completa puede encontrarse en el RFC 1780 Estándares de Protocolos Oficiales de Internet.

Tabla 7.-Estándares de protocolos oficiales de internet

Protocolo	Nombre	Estado	Estado	STD
IP	Protocolo de Internet	Estándar	Requerido	5
ICMP	Protocolo de Control de Mensajes de Internet	Estándar	Requerido	5
UDP	Protocolo de Datagrama de Usuario	Estándar	Recomendado	6
TCP	Protocolo de Control de Transmisión	Estándar	Recomendado	7
Telnet	Protocolo Telnet	Estándar	Recomendado	8
FTP	Protocolo de Transferencia de Ficheros	Estándar	Recomendado	9
SMTP	Protocolo Sencillo de Transferencia de Correo	Estándar	Recomendado	10
MAIL	Formato de Mensajes de Correo Electrónico	Estándar	Recomendado	11
DOMAIN	Sistema de Nombres de Dominio	Estándar	Recomendado	13
DNS-MX	Enrutamiento de Correo y el Sistema de	Estándar	Recomendado	14
MIME	Extensiones Multipropósito de Correo de	Borrador	Electivo	
SNMP	Protocolo Sencillo de Administración de Redes	Estándar	Recomendado	15
SMI	Estructura de Información de Administración	Estándar	Recomendado	16
MIB-I	Base de Información de Administración	Histórico	No Recomendado	
MIB-II	Base de Información de Administración-II	Estándar	Recomendado	17
NetBIOS	Protocolo de Servicios NetBIOS	Estándar	Electivo	19
TFTP	Protocolo de Transferencia de Ficheros Trivial	Estándar	Electivo	33
RIP	Protocolo de Información de Enrutamiento	Estándar	Electivo	34
ARP	Protocolo de Resolución de Direcciones	Estándar	Electivo	37
RARP	Protocolo de Resolución de Direcciones Inversa	Estándar	Electivo	38
GGP	Protocolo Pasarela a Pasarela	Histórico	No Recomendado	
BGP3	Protocolo de Pasarela Exterior 3	Borrador	Electivo	
OSPF2	Abrir Primero la Trayectoria Más Corta	Borrador	Electivo	
IS-IS	IS-IS OSI para Entornos Duales TCP/IP	Propuesto	Requerido	
BOOTP	Protocolo Bootstrap	Borrador	Recomendado	
GOPHER	Protocolo Gopher de Internet	Informativo		
SUN-NFS	Protocolo de Sistema de Ficheros de Red	Informativo		
SUN-RPC	Protocolo de Llamada a	Informativo		
	Procedimiento Remoto Versión 2			

A la hora de escribir no hay RFC asociados con HTTP usado en implementaciones WWW.

Los siguientes RFCs describen el URL y conceptos asociados:

- RFC 1630 - Identificador de Recursos Universal en WWW.
- RFC 1737 - Requerimientos Funcionales para Nombres de Recursos Uniformes.
- RFC 1738 - Localizador de Recursos Uniformes (URL).

2.9.4 Estándares de internet IETF

Internet Engineering Task Force (IETF) es una institución formada básicamente por técnicos en Internet e informática cuya misión es velar porque la arquitectura de la red y los protocolos técnicos que unen a millones de usuarios de todo el mundo funcionen correctamente. Es la organización que se considera con más autoridad para establecer modificaciones de los parámetros técnicos bajo los que funciona la red de redes Internet.

Convierte y promueve estándares de Internet, trabaja de cerca con W3C y ISO/IEC grupos que se ocupan particularmente de estándares TCP/IP y protocolos generales de Internet. Todos los participantes y líderes son voluntarios, aunque su trabajo es financiado generalmente por patrocinadores; por ejemplo, al momento es financiado por la entidad de verificación digital VeriSign y la agencia de seguridad nacional del gobierno de los Estados Unidos.

Se organiza grupos de trabajo y grupos de discusión (BOF) s, los cuales deben ocuparse de un asunto específico, dar solución y cerrar el caso. Cada grupo de trabajo tiene una tarea designada (o a veces varias co-tareas), junto con una carta que describa su enfoque, y qué y cuándo se espera que salga a producción.

Los grupos de trabajo se organizan en áreas por tema. Las áreas actuales incluyen: Casos de uso, casos generales, casos de Internet, operaciones y gerencia, usos e infraestructura en tiempo real, enrutamiento, seguridad, y transporte. Cada área es supervisada por su director del área, teniendo con la mayoría de las áreas dos co-directores. Los directores son responsables de designar tareas al grupo de funcionamiento. Los directores de área, junto con la tarea del IETF, forman Internet Engineering Steering Group (IESG), que es responsable de la operación total del IETF.

El IETF tiene una actividad conjunta con el Internet Society formalmente. El IETF es supervisado por Internet Architecture Board (IAB), que supervisa sus relaciones externas, y relaciones con el Redactor del RFC. El IAB es también responsable del Comité administrativo de descuidos del IETF (IAOC), que supervisa la actividad de la ayuda administrativa del IETF (IASA), que proporciona ayuda y seguimiento logístico para el IETF. El IAB también maneja Internet Research Task Force (IRTF), con que el IETF tiene relaciones de grupo de trabajo y control.

Área de seguridad

El área del IETF para seguridad empieza el 10 de diciembre del 2007, fue dirigido por Tim Polk en el National Institute of Standards and Technology (NIST) de los Estados Unidos y SAM Hartman que en ese entonces era del Instituto de Tecnología de Massachusetts (MIT) también en los Estados Unidos. Fueron apoyados por una cantidad de participantes (activos o no) que tenían direcciones de correo con dominios de bbn.com, de bear.com, de cisco.com, de cmu.edu, de columbia.edu, de comcast.net, de coopercairn.com, de gmx.net, de hactrn.net, de hotmail.com, de hyperthought.com, de ibm.com, de ieca.com, de ihtfp.com, de imc.org, de isode.com, de it.su.se, de iu-bremen.de, de juniper.net, de laposte.net, de ltsnet.net, de microsoft.com, de mit.edu, de motorola.com, de navy.mil, de nec.de, de networkresonance.com, de nokia.com, de nortel.com, de nortelnetworks.com, de opentext.com, de orionsec.com, de qualcomm.com, de rsa.com, de safenet-inc.com, de sendmail.com, de sun.com, de tcd.ie, de tislabs.com, de verisign.com, de vigilsec.com, y de xmission.com. La participación pública ocurre cuando se envía la lista nombrada por SAAG, que es recibido por el MIT y administrado por Polk, Hartman, y Jeffrey I. Schiller, que es el encargado de la seguridad de la red Internet y anterior director del área de seguridad general.

Jonatán Zittrain ha sugerido que los usuarios del Internet acepten la responsabilidad de supervisar el código fuente para evitar el vandalismo por Internet. Él escribió, “la responsabilidad del IETF de un estándar u otro debe ser provechoso y crucial para aumentar la seguridad en el Internet y así la tecnología de red generativa se pueda justificar”.

2.10 Metodología de diseño de red top down

2.10.1 Historia

Según OPPENHEIMER (2010, ed. 3) El diseño de red top-down es una disciplina que creció del éxito de la programación de software estructurado y el análisis de sistemas estructurados. El objetivo principal del análisis de sistemas estructurado es representar de modo más exacto las necesidades de los usuarios, que a menudo son lamentablemente ignoradas. Otro objetivo es hacer el proyecto manejable dividiéndolo en módulos que pueden ser más fáciles de mantener y cambiar.

El diseño de red top-down es una metodología para diseñar redes que comienza en las capas superiores del modelo de referencia OSI antes de mover a las capas inferiores. Esto se concentra en aplicaciones, sesiones, y transporte de datos antes de la selección de routers, switches, y medios que funcionan en las capas inferiores (Ver anexo 2)

El proceso de diseño de red top-down incluye exploración divisional y estructuras de grupo, para encontrar la gente, para quien la red proporcionará servicios y de quien se debería conseguir la información valiosa para hacer que el diseño tenga éxito.

Según Fonseca (2010, Ed. 1) El diseño de red top-down es también iterativo. Para evitar demora en detalles demasiado rápido, es importante conseguir primero una vista total de los requerimientos de un cliente. Más detalle puede ser juntado en comportamiento de protocolo, exigencias de escalabilidad, preferencias de tecnología. El diseño de red top-down reconoce que el modelo lógico y el diseño físico pueden cambiarse cuando más información es juntada.

Como la metodología top-down es iterativa, un acercamiento top-down deja a un diseñador de red ponerse "en un cuadro grande" primero y luego moverse en espiral hacia abajo según exigencias técnicas detalladas y especificaciones.

Con un acercamiento estructurado diseñamos la red, cada módulo es diseñado por separado, aún con relación a otros módulos. Todos los módulos son diseñados usando un acercamiento top-down, que se concentra en los requerimientos, aplicaciones, y una estructura lógica antes de la selección de dispositivos físicos y productos que se implementará en el diseño.

2.10.2 Fase I: Identificar objetivos y necesidades del cliente

Parte 1. Análisis de los objetivos y limitaciones del negocio

Según Huerta (2010, Ed. 3). Los objetivos y limitaciones incluyen la capacidad de correr las aplicaciones de red que reúne los objetivos comerciales corporativos, y la necesidad de trabajar dentro de restricciones comerciales, como paquete, personal limitado que está conectado a una red, y márgenes de tiempo cortos.

El comprender los objetivos comerciales y sus restricciones de sus clientes es un aspecto crítico del diseño de red. Armado con un análisis cuidadoso de los objetivos comerciales de su cliente, se puede proponer un diseño de red que contará con la aprobación de su cliente.

El entendimiento de la estructura corporativa también le ayudará a reconocer la jerarquía de dirección. Uno de sus primeros objetivos en las etapas tempranas del diseño de un proyecto de red debe determinar quiénes son los funcionarios con poder de decisión.

Parte 2. Análisis de los objetivos y limitaciones técnicas

En esta parte se trata de dar algunos alcances para analizar las metas técnicas de los clientes para implementar una nueva red o actualizar una existente. Conociendo las metas técnicas de nuestros clientes podremos recomendar nuevas tecnologías que al implementarlas cumplan con sus expectativas.

Los típicos objetivos técnicos son adaptabilidad, disponibilidad, funcionalidad, seguridad, manejabilidad, utilidad, adaptabilidad, y factibilidad.

Escalabilidad

La escalabilidad se refiere de cuanto es capaz de dar soporte al crecimiento del diseño de la red. Uno de los principales objetivos para muchas empresas es que su red sea altamente escalable, especialmente las empresas grandes que normalmente tienen un crecimiento rápido tanto en usuarios, aplicaciones y conexiones de red. El diseño de red que se propone a un cliente debería ser capaz de adaptarse a aumentos del uso de red y el alcance.

Disponibilidad

La disponibilidad se refiere a todo el tiempo que una red está disponible a usuarios y es a menudo una meta difícil de alcanzar para los que diseñan la red, ésta puede ser expresada en porcentajes por año, mes, semana, día u hora comparado con tiempo total del periodo.

La palabra disponibilidad puede ser mal entendida por los usuarios para lo que se debe ser muy cuidadoso en explicar en qué consiste la disponibilidad de la red para ello se puede usar la palabra fiabilidad que se refiere a varios factores, como la exactitud, rangos de error, estabilidad, y la cantidad de tiempo entre fracasos lo que refleja la disponibilidad de la red.

Disponibilidad también lo asocian con la redundancia que no es un objetivo para el diseño de red, más bien es una solución, se refiere que se duplica los enlaces a la red para reducir tiempos lo que permite continuidad después de fallas o desastres.

Disponibilidad está asociada también con la resistencia que significa cuánto estrés puede manejar la red con rapidez, que la red pueda manejar los problemas incluyendo los de seguridad, brechas, desastres naturales y no naturales, errores humanos, fallas del hardware o software.

Performance

Cuando se analiza los requerimientos técnicos para el diseño de la red, se puede convencer a los clientes para aceptar la performance de la red, incluyendo rendimiento, exactitud, eficacia, tardanza, y tiempo de respuesta.

Analizar el estado actual de la red puede ayudar a ver qué cambios se podrían realizar para que mejore la performance de la red. Las metas de la performance de la red están bastante ligada con las metas de la escalabilidad.

Seguridad

El diseño de la seguridad es uno de los aspectos más importantes en el diseño de red empresarial. Al incrementar las amenazas tanto dentro como fuera de la red de la empresa se debe tener reglas y tecnologías de seguridad actualizadas e incorruptibles.

Las metas más deseadas de muchas empresas es que los problemas de seguridad no afecten a la habilidad de conducir los negocios de la empresa, o sea

que si se presentara algún tipo de problema la empresa debe ser capaz de seguir con sus actividades normales.

La primera tarea para el diseño de la seguridad es planificar. Lo que significa que debemos reconocer las partes más vulnerables de la red, analizando los riesgos y encontrando requerimientos.

Manejabilidad (Administración)

Cada cliente tiene objetivos y una forma de administrar la red diferente. Algunos clientes tienen metas claras de cómo administrar la red y otras metas menos específicas. Si su cliente tiene proyectos definidos, debe asegurarse que se documenten, porque usted tendrá que referirse a los proyectos seleccionando el equipo. En algunos casos, el equipo tiene que ser excluido porque esto no soporta la administración de funciones que el cliente requiere.

La administración de la red debe ser simplificada. Simplificarlos en paquetes de funciones de administración se entienden fácilmente y usados por administradores de red.

Parte 3. Graficar la red existente

Se basa en una ejecución en un diagrama de una red y aprendiendo la localización de la mayoría de los dispositivos y segmentos en el trabajo de la red e identificando algunos métodos establecidos para el direccionamiento y nombramiento y también archivando, investigando los cables físicos, reservas que son muy importante en la característica de la infraestructura de la red.

Ejecución de un diagrama de red

Para la mayoría de los diseñadores de red; la interconexión de dispositivos y segmentar de la red es un buen camino para comenzar la comprensión del flujo circulatorio. El objetivo es obtener un diagrama ya implementado de la red, algunos diseños de los clientes pueden tener diagramas para un nuevo y mejor diseño de la red.

Herramientas para la ejecución de un diagrama de red

Para ejecutar un diagrama de la existencia de la red, deberíamos invertir en una buena herramienta de diagrama de red. Tales como:

- Visio Corporations.
- Visio Profesional.
- Visio Profesional Ships.

Algunas compañías ofrecen esquematizar automáticamente el descubrimiento de la red existente, usando el siguiente software:

- Pinpoint Software's ClickNet Professional.
- NetSuite Development.
- Net Suite Advanced Professional Design.
- NetSuite Professional Audit (similar ClickNet).

Incluir en un diagrama de red

Usando las herramientas mencionadas deberá desarrollar un diagrama de red en la cual deberá contener lo siguiente:

- Conexiones WAN entre países y ciudades.
- Edificios y pisos, y posibilidades cuartos y casetas.
- Conexiones WAN y LAN entre edificios y entre campos.
- Una indicación de la capa de datos (WAN, LANS).
- El Número de servicios proveedor de WANS.
- La localización de las líneas e interruptores, aunque no es necesario en el eje y centro.
- La localización y alcance de redes virtuales (VPN's), que conecta los servicios de los proveedor WAN.
- La localización de las principales estructuras.
- La localización de las mayores estaciones de ejecución de la red.
- La localización y alcance de algunas LAN's Virtuales (VLAN's).
- La topología de algunos sistemas de seguridad Firewall.
- La localización de algunos sistemas de dial- in y dial out.
- Algunas indicaciones de donde residen algunas estaciones de trabajo, aunque no necesariamente la localización explícita de cada estación de trabajo.

Caracterizando el direccionamiento y el nombramiento de la red

La infraestructura lógica de la red envuelve documentar cualquier estrategia que su cliente tiene para el direccionamiento y nombramiento de la red. Cuando dibuje los detalles de los diagramas de la red, deberá incluir los sites, routers, segmentos de la red y servicios. Usted tiene que investigar el direccionamiento de la capa de red que usa, el esquema de direccionamiento que usa su cliente puede influenciar en la habilidad de adaptar su nuevo diseño de red a los objetivos, aquí definirá el mejor método de direccionamiento que se pueda usar para su diseño de red. Entre los cuales tenemos:

- Subnetting.
- Variable Length Subnet Masking (VLSM).
- Supernetting o Aggregations.
- Summarization.
- Estos métodos se explicaran más adelante cuando se seleccione el protocolo y direccionamiento de red.

Características del cableado y el medio

Es importante comprender el cableado y la instalación eléctrica del diseño de la red con el objetivo de identificar posibles y potenciales problemas. Si es posible se deberá documentar el tipo de cableado que usa, la distancia ya que esta información ayudará a determinar la tecnología de la capa de enlace basado en las restricciones de distancia. Cuando el diseño del cableado está en exploración, determine cuáles son los equipos y los cables que están etiquetado en la red existente. Por ejemplo: la red de un edificio debería archivar las conexiones de un número de cable y el tipo de instalación que está en uso en la red.

Probablemente la instalación entre los edificios es unos de los siguientes:

- Single –mode fiber
- Multi –mode fiber
- Shielded twisted pair (STP)
- UTP categoría 5
- Cable coaxial
- Microondas
- Radiofrecuencia.

- Láser
- Infrarrojo

Arquitectura ambiental y restricciones

Cuando se está investigando el cableado hay que poner mucha atención en los problemas ambientales con la posibilidad de que el cableado podría pasar muy cerca donde haya lugares propensos a inundarse, cerca de las carreteras donde el tráfico de los vehículos podría quebrar los cables, calefacciones, etc.

Este seguro que no tenga ningún problema legal a la hora de tender un cableado, por ejemplo al cruzar un cableado por una calle donde tenga que romper pistas. Cuando construya preste atención a la arquitectura si este afecta la implementación de su diseño, este seguro que la arquitectura puede soportar el diseño tales como:

- Aire acondicionado.
- Calefacción.
- Ventilación.
- Protección de interferencias electromagnéticas.
- Puertas que no estén cerradas.

Funcionamiento de la red existente

Estudiar el performance de una red existente te da una línea básica dimensional para poder medir y compara el performance del nuevo diseño de red propuesto el cual le ayudara a demostrar a su cliente cuan mejor es su diseño en performance una vez implementado.

Si la red es muy grande para estudiar todos sus segmentos, preste mayor atención en la red de backbone antigua y las nuevas áreas que se conectan así como redes que se integran al backbone. Por ejemplo capturar la circulación la red con un analizador de protocolo como parte de tu análisis de la línea básica, podría identificar cuáles de los protocolos están realmente trabajando en la red y no contar con la creencia de los clientes (ethereal).

Performance precisa de la Red

Poder identificar la performance precisa de una red no es tarea fácil. Una de las tareas es seleccionar el tiempo para hacer estos análisis para poder determinar el

momento exacto para poder realizarlo y determinar los problemas que presenta la red durante los periodos altos de tráfico, etc.

Los problemas de la red no son usualmente causados por los envíos de malas estructuras de tramas. En el caso token ring (La red Token-Ring es una implementación del standard IEEE 802.5), en el cual se distingue más por su método de transmitir la información que por la forma en que se conectan las computadoras, el problema usualmente esta por estación y problema de cable, en el caso de ethernet, es un difícil precisar la causa del problema. Algunos clientes no reconocen el valor de estudiar las redes existentes antes del diseño y la implementación. Los clientes generalmente se avocan por un diseño rápido por lo cual puede hacer difícil poder dar marcha atrás e insistir en tiempo para desarrollar la performance precisa de la red existente. Un buen entendimiento de los objetivos técnicos y de negocio del cliente pueden ayudar a decidir qué cantidad de tráfico deberá analizar en la red.

Disponibilidad de la red

Para documentar características de disponibilidad de la red existente, junte cualquier estadística que el cliente tiene durante el tiempo medio entre fallas (MTBF) y tiempo medio de reparación (MTTR) para las redes en conjunto así como segmentos de red principales. Compare estas estadísticas con la información en la que usted se ha juntado en MTBF y objetivos MTTR, ¿Espera el cliente que su nuevo diseño aumente MTBF y disminuya MTTR? ¿Son los objetivos del cliente consideración realista del estado corriente de la red?

Parte 4. Características de un diseño de tráfico de red

En parte se describe las técnicas para caracterizar el flujo de tráfico, el volumen de tráfico, y el comportamiento de protocolo. Las técnicas incluyen el reconocimiento de tráfico fuente y almacenaje de datos, documentar las aplicaciones y uso el de protocolo, y evaluar del tráfico de red causado por protocolos comunes.

En la parte anterior se habló de la caracterización de la red existente en términos de su estructura e interpretación. Como el análisis de la situación existente es un paso importante en un acercamiento de análisis de sistemas para diseñar, esta sección se habla de la caracterización de la red existente en términos de flujo de tráfico. Esta sección también cubre nuevas exigencias de diseño de red, añadiendo las dos primeras secciones que cubrieron objetivos de diseño comerciales y técnicos.

Esta sección reenfoca en exigencias de diseño y describe exigencias en términos de flujo de tráfico, carga, y comportamiento; y calidad de servicio (QoS) exigencias.

Características el flujo de trafico

La caracterización del flujo de tráfico implica identificar fuentes y destinos del tráfico de red y analizar la dirección y la simetría de datos que viajan entre fuentes y destinos. En algunas aplicaciones, el flujo es bidireccional y simétrico. (Ambos finales del flujo envían el tráfico en aproximadamente el mismo precio.) En otras aplicaciones, el flujo es bidireccional y asimétrico. Las estaciones de cliente envían pequeñas preguntas y los servidores envían grandes corrientes de datos. Los broadcast de una aplicación, el flujo es unidireccional y asimétrico. Esta sección habla de la caracterización de la dirección y la simetría del flujo de tráfico en una red existente y análisis del flujo para nuevas aplicaciones de red.

Identificación de las principales fuentes de tráfico y almacenamiento

Para entender el flujo de tráfico de red, usted debería identificar primero comunidades de usuario y almacenamiento de datos para las aplicaciones existentes. A comunidad de usuario es un grupo de trabajadores que usan una aplicación particular o un grupo de aplicaciones. Una comunidad de usuario puede ser un departamento corporativo o un grupo de departamentos. Sin embargo, en muchos ambientes, el uso de aplicación cruza muchos departamentos. Cuando más corporaciones usan la dirección de la matriz y forman equipos virtuales para completar un proyecto, se hace más necesario caracterizar comunidades de usuario por aplicación y uso de protocolo más bien que por el límite de departamentos.

2.10.3 Fase II: Diseño de una red lógica

Parte 5. Diseño de una topología de red

Fonseca (2010, Ed. 1) La topología es un diagrama de la red que indican segmentos de red, puntos de interconexión, y comunidades de usuario. Además los sitios geográficos puedan aparecer en el diagrama, el objetivo del diagrama es mostrar la geometría de la red, no la geografía física o implementación técnica. El diagrama es una vista panorámica del alto nivel de la red, análoga a un dibujo arquitectónico que muestra la posición y el tamaño de cuartos para un edificio, pero no los materiales de construcción para fabricar los cuartos.

El diseño de una topología de red es el primer paso en la fase de diseño lógica de la metodología de diseño de red Top Down. Para encontrar los objetivos de un cliente para escalabilidad y adaptabilidad, es importante para el arquitecto una topología lógica antes de seleccionar productos físicos o tecnologías. Durante la fase de diseño de topología, usted identifica redes y puntos de interconexión, el tamaño y alcance de redes, y los tipos de dispositivos de funcionamiento entre redes que serán requeridos, pero no los dispositivos actuales.

Diseño de red jerárquica

Para encontrar los objetivos comerciales y técnicos de un cliente para un diseño de red corporativo, usted podría tener que recomendar que una topología de red que consiste en muchos interrelacionara componentes. Esta tarea es hecha más fácil si usted puede "dividir y triunfar" el trabajo y desarrollar el diseño en capas.

La capa core

La capa core de una topología jerárquica de tres capas es la columna vertebral rápida de las redes. Como la capa core es crítica para la interconectividad, usted debería diseñar la capa core con componentes redundantes. La capa core debería ser muy confiable y debería adaptarse a cambios rápidamente.

La capa de distribución

La capa de distribución de la red es el punto de demarcación entre el acceso y las capas core de la red. La capa de distribución tiene muchos roles, incluso el control del acceso a recursos por razones de seguridad, y control del tráfico de red que cruza el core por motivos de performance. La capa de distribución es a menudo la capa que delinea el dominio de broadcast, (aunque este pueda ser hecho en la capa de acceso también). En diseños de red que incluyen LANs virtuales (VLANs), la capa de distribución puede ser configurada para rutear entre VLANs.

La capa de acceso

La capa de acceso proporciona a usuarios locales del segmento, el acceso a las redes. La capa de acceso puede incluir routers, switches, puentes, hubs para compartir medios, y puntos de acceso inalámbricos. Como los switches frecuentemente son implementados en la capa de acceso, para dividir los dominios de ancho de banda para encontrar las demandas de aplicaciones que necesitan

mucho ancho de banda o no pueden resistir la tardanza variable caracterizada por el ancho de banda compartida.

Parte 6. Diseño de un modelo de direccionamiento

Esta parte proporciona pautas para adjudicar direcciones y nombres a componentes de redes, incluso redes, subredes, routers, servidores, y sistemas de final. En esta parte se enfoca en el Protocolo de Internet (IP) la dirección y el nombramiento. Para beneficiarse más de este capítulo, usted debería tener ya un entendimiento básico de la dirección de IP.

Este parte ilustra la importancia de usar un modelo estructurado para dirección de capa de red y nombramiento. Sin la estructura, es fácil quedarse sin direcciones, desperdiciar direcciones, introducir direcciones duplicadas y nombres, y direcciones de uso y nombres que son difíciles de manejar. Para encontrar los objetivos de un cliente para escalabilidad, performance, y manejabilidad, usted debería asignar direcciones y nombres sistemáticamente.

Este parte también demuestra la importancia de desarrollar políticas y procedimientos para direccionamiento y nombramiento. Las políticas a menudo implican un plan para distribuir autoridades para direccionamiento y nombramiento para evitar que un departamento tenga que manejar todas las direcciones y nombres. Una central de autoridad pueden asignarse por bloques de direcciones y nombres en una manera jerárquica a departamentos y sucursales.

Pautas para asignar direcciones de capa de red

Las direcciones de capa de red deberían ser planeadas, manejadas, y documentadas. Aunque un sistema final pueda aprender su dirección dinámicamente, no existe ningun mecanismos para asignar a la red o números de subnet dinámicamente. Estos números deben ser planeados y administrados. Muchas redes añejas donde todavía existen direccionamiento no son planeadas o documentadas. Estas redes son difíciles cuando fallan y no escalan.

Parte 7. Selección de los switching y protocolo de enrutamiento

El objetivo de esta parte es ayudarle a seleccionar correctamente los protocolos conmutados y de enrutamiento para el diseño de su red al cliente. Las selecciones que usted hace dependerán de los objetivos comerciales y técnicos de su cliente.

Para ayudarle a seleccionar los protocolos correctos para su cliente, la parte cubre los atributos siguientes de conmutación y enrutamiento de protocolos:

Características de tráfico de red.

- Ancho de banda, memoria, y uso de CPU.
- El número aproximado en el tráfico de puntos de routers o switches que soportan.
- La capacidad de adaptarse rápidamente a cambios de una red.
- La capacidad de certificar rutas actualizadas por razones de seguridad.

En este punto en el proceso de diseño de red, usted ha creado una topología de diseño de red y ha desarrollado alguna idea de donde los switches y los routers residirán, pero usted no ha seleccionado ningún switch actual o productos de router. Un entendimiento de la conmutación y enrutamiento de protocolos que un switch o el router deben soportar le ayudará a seleccionar el mejor producto para el trabajo.

Parte 8. Desarrollo de las estrategias de seguridad de red

El desarrollo de estrategias de seguridad que pueden proteger todas las partes de una red complicada teniendo un efecto limitado en la facilidad de uso e interpretación es una de las tareas más importantes y difíciles relacionadas para conectar diseño de red. El diseño de seguridad es desafiado por la complejidad y la naturaleza porosa de redes modernas que incluyen a servidores públicos para el comercio electrónico, extranet conexiones para socios de negocio, y servicios de acceso remoto para usuarios que alcanzan la red de casa, sitios de cliente, cuartos del hotel, cafeterías de Internet, etcétera. Para ayudarle a manejar las dificultades inherentes en el diseño de la seguridad de red para redes complejas, este capítulo enseña un acercamiento sistemático, top down que se concentra en planificación y desarrollo de política antes de la selección de productos de seguridad.

El objetivo de esta parte es ayudarle a trabajar con sus clientes el diseño de red en el desarrollo de estrategias de seguridad eficaces, y ayudarle a seleccionar las técnicas correctas para poner en práctica las estrategias. El capítulo describe los pasos para desarrollar una estrategia de seguridad y cubre algunos principios de seguridad básicos. El capítulo presenta un acercamiento modular al diseño de seguridad que le dejará aplicar soluciones acodadas que protegen una red desde muchos puntos de vista. Las secciones finales describen métodos para asegurar los componentes de una red de empresa típica que son las que más están en peligro,

incluso conexiones de Internet, redes de acceso remoto, redes de servicios de usuario, y redes inalámbricas.

Diseño de seguridad de la Red

Después de desarrollar los pasos del set estructurado y poniendo en práctica la seguridad de red le ayudará a dirigirse a las preocupaciones variadas que juegan una parte en el diseño de seguridad. Muchas estrategias de seguridad han sido desarrolladas de un modo desordenado y han dejado de asegurar realmente activos y encontrar los objetivos primarios de un cliente para la seguridad. La demolición del proceso del diseño de seguridad en los pasos siguientes le ayudará con eficacia a planear y ejecutar una estrategia de seguridad:

1. Identifique activos de red.
2. Analice riesgos a la seguridad.
3. Analice los requerimientos de seguridad y restricciones.
4. Desarrolle un plan de seguridad.
5. Defina una política de seguridad.
6. Desarrolle procedimientos para aplicar políticas de seguridad.
7. Desarrolle una estrategia de realización técnica.
8. Consiga la compra - desde usuarios, gerentes, y personal técnico.
9. Entrene a usuarios, gerentes, y personal técnico.
10. Ponga en práctica la estrategia técnica y procedimientos de seguridad.
11. Pruebe la seguridad y actualícelo si algún problema es encontrado.
12. Mantenga la seguridad programando auditorías independientes periódicas, leyendo los logs de auditoría, respondiendo a incidentes, leyendo literatura corriente y alarmas de agencia, siguiendo probando y entrenarse, y actualizando el plan de seguridad y política.

Parte 9. Desarrollar estrategias de manejo de red

Esta parte concluye la discusión del diseño de red lógico. El manejo de red es uno de los aspectos más importantes del diseño de red lógico. El manejo a menudo es pasado por alto durante el diseño de una red porque es considerada una cuestión operacional más bien que una cuestión de diseño. Sin embargo, si usted considera el manejo al principio, puede evitar escalabilidad y problemas de performance que ocurren cuando el manejo es añadido a un diseño después de que el diseño está completo.

Diseño del manejo de Red

Esto es una idea buena de acercarse al diseño de manejo de red del mismo modo usted se acerca a cualquier proyecto de diseño. Piense en escalabilidad, modelos de tráfico, formatos de datos, y compensaciones de costo/ventaja. Los sistemas de manejo de red pueden ser muy caros. Ellos también pueden tener un efecto negativo en la performance de red.

Preste la atención al principio de incertidumbre Heisenberg, que declara que el acto de observación de algo puede cambiar lo que es observado. Algunos sistemas de manejo de red causan colas en estaciones remotas en una base regular. La cantidad de tráfico causado por la cola puede ser significativa. Usted debería analizar los requerimientos de su cliente para colas temporizadores y no arbitrariamente usar las faltas de un sistema de manejo de red.

El trabajar con su cliente para entender que los recursos deberían ser monitoreados y la métrica para usar midiendo la performance de los dispositivos. Elija los datos para reunirse con cuidado. El ahorro de demasiados datos puede causar un requerimiento para una supercomputadora para tratar y almacenar los datos. Por otra parte, procure no tirar tantos datos que usted es incapaz de usar los datos restantes para manejar la red.

2.10.4 Fase III: Diseño de la red física

Selección de tecnología y dispositivos de red de campus

Oppenheimer (2010, Ed. 3) Se cubre tecnologías para diseños de red de campus. Una red de campus es un juego de segmentos de LAN y redes de construcción en un área que tiene unas millas de diámetro. El siguiente capítulo cubre tecnologías para una red de empresa que incluye WAN y servicios de acceso remoto.

Con respecto a esta fase del diseño descendente se utilizará solo como referencia para determinar las características de los dispositivos, no así como una referencia teórica de lo que se llevará a cabo durante la ejecución de este proyecto.

2.10.5 Fase IV: Testeo, optimización y documentación de la red

Stallings (2004). Indica que las pruebas de su diseño de red son un paso importante en el proceso de diseño que permite que se confirme que el diseño encuentra los objetivos comerciales y técnicos. Probando su diseño, se puede

verificar que las soluciones que usted ha desarrollado proporcionarán la performance y QoS que su cliente espera.

Testear el diseño de red

Las pruebas le ayudarán a demostrar que su diseño de red al cliente es su solución que permite el logro de objetivos comerciales y técnicos.

Construir y testear un prototipo de sistema de red

El objetivo de esta fase es ayudarle a hacer una lista de las tareas para construir un prototipo que verifica y demuestra el comportamiento de un sistema de red. Un objetivo secundario es ayudarle a determinar cuánto de un sistema de red debe ser puesto en práctica en un prototipo para verificar el diseño.

Un prototipo es una realización inicial de un nuevo sistema que proporciona un modelo en el cual la realización final será modelada. Un prototipo permite que un diseñador valide la operación y la performance de un nuevo sistema. Debería ser funcional, pero no tiene que ser una realización de tamaño natural del nuevo sistema. Esto debería resultar, sin embargo, de un análisis cuidadoso de necesidades como de una revisión de diseño con el cliente de final.

Determinar el alcance de un sistema de prototipo

Basado en un entendimiento claro de los objetivos de su cliente, usted debería determinar cuánto del sistema de red usted debe poner en práctica para convencer a su cliente que el diseño encontrará exigencias. Como no es generalmente práctico poner en práctica un sistema completo, de tamaño natural, usted debería aislar qué aspectos de un diseño de red son los más importantes para su cliente. Su prototipo debería verificar capacidades importantes y funciones que no podrían funcionar suficientemente. Las funciones arriesgadas pueden incluir funciones complejas, intrincadas e interacciones componentes, así como funciones donde el diseño era bajo la influencia de coacciones comerciales o técnicas, y compensaciones con objetivos contrarios.

Documentación de equipo de red y otros recursos

Un plan de prueba debería incluir un dibujo de topología de red y una lista de dispositivos que serán requeridos. El dibujo de topología debería incluir dispositivos principales, direcciones, nombres, enlaces de red, y alguna indicación de

capacidades de enlaces. El dibujo de topología también debería documentar alguna WAN o enlace de LAN que deben unirse a la red de producción o al Internet.

La lista de dispositivos debería incluir hubs, repetidores, switches, routers, estaciones de trabajo, servidores, simuladores de equipo telefónico, puntos de acceso inalámbricos, firewalls, cables, etcétera. La lista debería documentar números de versión para hardware y software, e información de disponibilidad. A veces las pruebas requieren el nuevo equipo que no podría estar disponible aún, o equipo que por otros motivos tiene un tiempo de plomo largo para la consecución. Si es así, debería ser notado en el plan de prueba.

2.11 Casos de éxito

- **Universidad Técnica de Machala**

La Universidad Técnica de Machala, desde el año 2012, ha Implementado la infraestructura de red de datos para laboratorios y biblioteca utilizando la metodología descendente “top down”.

Para ello ha realizado un análisis, diseño e implementación de una Red Híbrida para La Universidad Técnica de Machala, basada en la metodología Top Down y la configuración de un firewall pfSense con los respectivos módulos Snort y Squid. Se han aplicado conocimientos cultivados en la Universidad Técnica de Machala, que garantice el normal y correcto funcionamiento de la red y la configuración respectiva del firewall, ayudando a sacar el mayor provecho de la misma y brindando un mayor rendimiento.

El proyecto está basado en gran parte en la metodología Top-Down Network Designer para redes cableadas, la misma que se la ha acoplado para el desarrollo del proyecto, asociando las necesidades de la institución con la tecnología disponible, generando así una infraestructura de red eficiente y segura.

Dentro de la metodología se ha dado mayor relevancia a la seguridad, implementando un firewall pfSense el cual se usara los módulos Snort (ids) que es un sistema de detección de intrusos y un Squid (proxy). Todo el diseño de la red se lo probó en base a su funcionamiento mediante un programa de simulación packet tracer, probando el rendimiento de la red.

Es así que la implementación del proyecto en la La Universidad Técnica de Machala se dio de manera exitosa cumpliendo con las expectativas de los alumnos y directivos de la institución

CAPÍTULO III. METODOLOGÍA DE LA INVESTIGACIÓN

3.1 Introducción

En este capítulo se muestra el diseño que guio el proceso de ejecución de la tesis, el tipo de investigación en la que se encuentra y asimismo se muestra el análisis de cumplimiento de la metodología.

3.2 Lugar de aplicación

La aplicación y la validación del presente Proyecto de Investigación será en la en la Universidad Nacional de San Martín – Tarapoto (UNSM-T).

3.3 Tipo de investigación

Para el desarrollo del presente proyecto se considera 2 tipos de investigación:

Propositiva

Porque da una solución al problema en el uso del servicio de internet en la UNSM-T.

Aplicativa.

Porque aplica una metodología específica relacionada al diseño y optimización de redes para mejorar los servicios y ahorro de los recursos de la institución.

3.4 Metodología de la investigación

A continuación se presenta la metodología de investigación a utilizar para el desarrollo del proyecto.

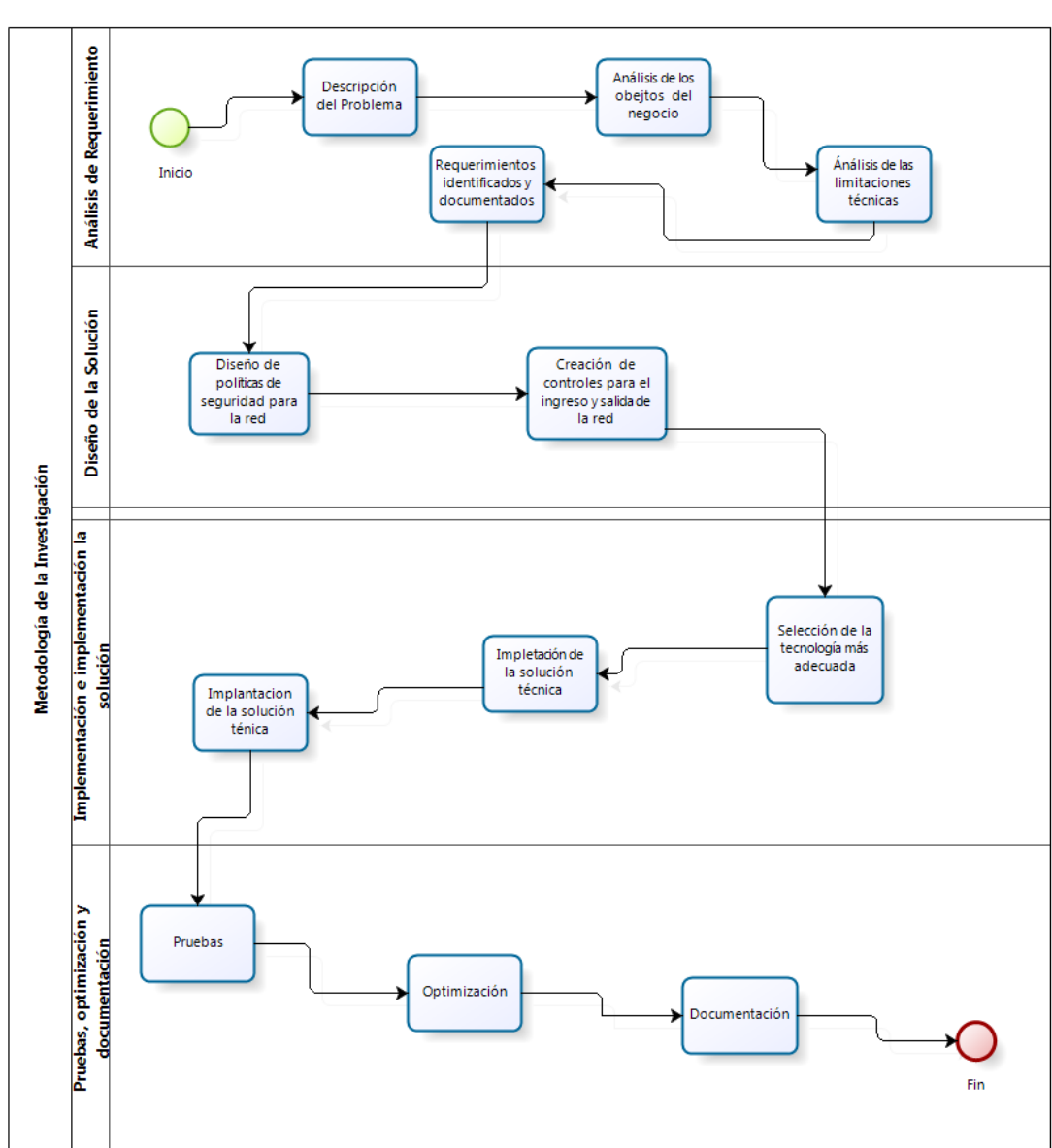


Figura 3.- Diseño de la metodología de la investigación

3.4.1 Descripción del Problema

Se identificarán los problemas de red de la organización, las causas y consecuencias a fin de determinar la mejor manera de plantear la solución.

3.4.2 Análisis de los objetivos del negocio

Esto se refiere a la manera en la que la TI (redes y comunicaciones) apoya al logro de los objetivos de la misma.

3.4.3 Análisis de las limitaciones técnicas

Es importante determinar cuáles serán las limitaciones técnicas a fin de poder realizar los ajustes necesarios a la solución propuesta.

3.4.4 Requerimientos identificados y documentados

Se permite tener documentos que permitan respaldar las modificaciones y aplicación de los cambios basándose en documentos.

3.4.5 Diseño de Políticas de Seguridad para la red

Se diseñan políticas a fin de mejorar la seguridad de la red.

3.4.6 Creación de controles para el ingreso y salida de la red

Específicamente a través de las Listas de Control de Acceso que filtran el contenido permitido y censurado basándose en la URL de destino.

3.4.7 Selección de la tecnología más adecuada

De acuerdo a la realidad de la organización se plantea el uso de software libre como plataforma base para la ejecución de este proyecto.

3.4.8 Implementación de la solución técnica

En un ambiente de laboratorio creado especialmente para el proyecto se implementa la solución.

3.4.9 Implantación de la solución técnica

Luego se procede a la implantación final en el área de producción.

3.4.10 Pruebas

Se realizan las pruebas necesarias a fin de garantizar la correcta implantación de la solución.

3.4.11 Optimización

En función a las pruebas realizadas se realiza un proceso de retroalimentación a fin de optimizar los resultados de la solución.

3.4.12 Documentación

Se documenta la solución y se entrega el producto terminado.

CAPÍTULO IV. DESARROLLO DE LA SOLUCIÓN

4.1 Introducción

El capítulo muestra la solución del plan de mejora del ancho de banda de internet y seguridad aplicados a la red de datos, desarrollando el análisis de requerimientos, diseño físico, diseño lógico, testeo, optimización y documentación de la red implementado, un servidor proxy con las herramientas: squid, webmin, sarg y un corta fuego para mejorar la seguridad de la red, definidos como parte del proyecto.

4.2 Análisis de requerimientos.

En la ciudad universitaria se cuenta actualmente con una línea dedicada de 4 Mbps, a través de la cual se brinda servicio de Internet a aproximadamente al (99%) de los equipos de las oficinas administrativas y académicas, y también a las laptops de estudiantes, docentes y otros visitantes que se conectan a la red inalámbrica de la ciudad universitaria. En conjunto suman alrededor de 750 equipos de cómputo.

Los servicios prestados por la red de fibra óptica son los siguientes:

4.2.1 Acceso a internet.

Se cuenta con 1 línea dedicada a través de los cuales se accede a internet Los locales del Complejo Universitario (OCRA, Biblioteca Central y Oficina de Admisión), Ciudad Universitaria y la Facultad de Turismo de Lamas acceden a internet, a través de la línea dedicada ubicada en el nodo de administración de la red de fibra óptica.

En casos de contingencia cuando en el local central no se cuenta con internet debido a fallas del proveedor del servicio y/u otros acontecimientos, éste muchas veces utiliza como puerta de salida la que se encuentra en el nodo de administración de fibra óptica (gracias a los radio enlaces ubicados en las torres respectivas y que se detallan en la gráfica correspondiente), permitiendo garantizar la continuidad de la operativa diaria de las oficinas administrativas ubicadas en el mismo.

4.2.2 Red inalámbrica.

El objetivo de la red WI-FI es ofrecer conectividad a la comunidad universitaria (alumnos, docentes, profesionales, administrativos e invitados) con el fin de acceder a servicios y sistemas corporativos, y navegar en Internet haciendo uso de equipos móviles (notebook, PDA y similares, para lo cual se cuenta con 30 puntos de acceso.

El servicio WI-FI corporativo de la Universidad es ofrecido y administrado a través de la Oficina de Administración de la Red de Fibra Óptica de la Universidad Nacional de San Martín – Tarapoto.

Las zonas de cobertura corresponden a sectores estratégicos de la Universidad y espacios de masiva concurrencia al aire libre o de interior.

Es una red segura, rápida, monitoreada y no tiene costo para el usuario final.

No está demás mencionar que los vecinos de la ciudad universitaria que cuenten con equipos de enlace inalámbrico, también se ven beneficiados por este servicio y se podría considerar como parte de proyección a la comunidad.

La Oficina de Administración de la Red de Fibra Óptica se reserva el derecho de modificar las políticas de acceso al servicio sin previo aviso, siempre pensando en resguardar la prestación del servicio con los más altos estándares de calidad y seguridad

4.2.3 Compartición de recursos.

Dadas las características en cuanto a velocidad y capacidad de transmisión de datos, la red de fibra óptica, la misma se convierte en el mejor medio para que las oficinas compartan sus recursos como pueden ser impresoras, escáneres, discos duros, lectores de CD/DVD

Se permite optimizar la distribución de los recursos, sobre todo los relacionados a los periféricos y dispositivos de lectura de Cd y Dvd, impresoras, escáneres.

Este beneficio incluye un soporte adicional para el área técnica puesto que a través de la compartición de recursos, se crearon servidores de archivo donde se colocan los drivers, programas y documentos necesarios para brindar un mejor servicio.

4.2.4 Control del servicio de acceso a internet.

Con la finalidad de que se dé buen uso al servicio de acceso a internet se tiene configurado servidor proxy ubicado en la Ciudad Universitaria, a través del cual se controla las salidas del personal administrativo, personal docente y alumnos, con la finalidad de evitar el acceso a páginas que no son de uso académico y/o administrativo.

Se bloquean todas las páginas de porno, chistes, además de restringir el acceso del servicio de mensajería instantánea (pues el mismo sólo sirve para distraer la atención del personal administrativo).

También están bloqueadas páginas de radio, televisión y videos en línea; debido a que las páginas como éstas se caracterizan porque utilizan extensamente el ancho de banda y no sólo eso, sino que priorizan su uso, cómo se sabe, la UNSM-T accede a internet por uno de los servicios de menor costo del mercado, lo que implica que no tenemos un ancho de banda muy bueno, de tal forma que el servicio se satura, en perjuicio de aquellos usuarios que muchas veces necesitan acceder a sus correo o a otros servicios que sí forman parte de su operativa diaria en consecuencia, se reciben múltiples llamadas quejándose.

La solución más adecuada, es justamente el bloqueo de esas páginas y en general, el principal objetivo es que todos puedan acceder al servicio de internet, lo más pronto posible y cuando lo requieran de tal forma que todos estén contentos.

Lo anterior permite dar un uso óptimo de los servicios, de tal forma que no se desaproveche el ancho de banda del acceso a internet.

Para realizar todas las tareas mencionadas, contamos con un servidor proxy SQUID, el mismo que se pasa a describir a continuación:

4.2.5 Servidor de configuración dinámica de host.

Aquellos equipos de cómputo que se conectan a la red inalámbrica, necesita de una configuración especial que les permita acceder a los recursos disponibles en la Red (internet principalmente), es por ello que se ha configurado un servidor que les proporciona una identificación y puerto de enlace para hacer que ese trabajo administrativo sea automático y no se necesita la presencia de personal técnico calificado para esto.

4.2.6 Servidor FTP.

La UNSM cuenta con un servidor FTP, el mismo que está destinado para brindar el servicio de transferencia de archivos a las oficinas administrativas que así lo requiera y para los estudiantes de la Facultad de Ingeniería de Sistemas e Informática, quienes utilizan intensivamente el servicio por la misma naturaleza de la carrera.

4.2.7 Seguridad.

Debido a la gran cantidad de usuarios con los que se cuenta en la red de campus de la ciudad universitaria (más de 250 en fechas y horas punta, según reporte del servidor proxy) y además de contar con puertas de Acceso a Internet, existe un elevado riesgo de sufrir ataques direccionados a detener los servicios brindados por la oficina de administración de la fibra óptica y que esto repercuta en la operatividad diaria de las diferentes oficinas académicas y administrativas de la UNSM-T, se ha implantado un servidor firewall que garantiza la seguridad en el caso de ataques dirigidos hacia la red interna desde internet.

Se cuenta con un antivirus capaz de bloquear aplicaciones y programas no autorizados en los servidores, frente a las amenazas por los ciberdelincuentes, bloqueando virus, gusanos, troyanos, spam, filtrado de mensajes; para proteger la información inapropiada o confidencial que entra o sale de la red, y otros programas potencialmente no deseados. Se actualiza de forma automática y se puede administrar en forma central desde una consola, la plataforma Kaspersky Administration. Lo que le ayudará a cumplir las directivas y los requisitos de cumplimiento de normativas.

4.2.8 Disponibilidad

EL servidor contara con un servicio que consiste en la replicación de los servidores, de tal forma que si falla un equipo, entraría automáticamente en funcionamiento su servidor replicado asumiendo los datos y tareas del servidor que se ha caído.

Las aplicaciones y/o servicios seguirían disponibles para los usuarios cuando el hardware, aplicaciones y/o sistema operativo falle, sin merma de la productividad de la universidad.

4.2.9 Escalabilidad

Tomando en cuenta la cantidad de computadoras en la UNSM-T, este diseño deberá soportar un crecimiento de la red permitiendo que se pueda incluir nuevos host, nodos; dejando puertos adicionales en cada área de la UNSM-T para un posible crecimiento, se estarían dejando para el futuro, para lograr este objetivo, un diseño lógico jerárquico.

Balance de carga

Es necesario considerar balancear la carga de trabajo, usada para dividir el trabajo a compartir entre varios procesos, ordenadores, u otros recursos. Esta muy relacionada con los sistemas multiprocesales, que trabajan o pueden trabajar con más de una unidad para llevar a cabo su funcionalidad. Para evitar los cuellos de botella, el balance de la carga de trabajo se reparte de forma equitativa a través de un algoritmo que estudia las peticiones del sistema y la re direcciona a la mejor opción

Cluster

Introduce la capacidad de unir varios servidores para que trabajen en un entorno en paralelo. Es decir, trabajar como si fuera un solo servidor el existente. En las etapas primigenias del clustering, los diseños presentaban graves problemas que se han ido subsanando con la evolución de este campo

Cluster Balanceado.

Sera capaz de repartir el tráfico entrante entre múltiples servidores corriendo las mismas aplicaciones. Todos los nodos del cluster pueden aceptar y responder peticiones. Si un nodo falla, el tráfico se sigue repartiendo entre los nodos restantes.

Puntos fuertes y débiles del medio, ya que una red es tan eficaz como lo sea su cableado subyacente.

Cable de fibra horizontal

Se deben seguir los estándares. En concreto, el EIA/TIA 568 que establece el cableado a nivel de edificios.

Todo dispositivo conectado a la red deberá estar enlazado con una ubicación central mediante cableado horizontal, con limitación de 100m para UTP cat. 5 y 400m

4.2.10 Performance

La red deberá soportar un control de fallas; como son sobrecargas de voltaje que puedan originar caídas de los servidores que tengan nuestra red y perdida de datos que es atribuida a fallas de disco duro del servidor, la perdida de datos puede ocurrir cuando determinados usuarios borran archivos o se introduce un virus informático destructivo en la red. Esto se puede solucionar implementando un plan de

contingencia en cuanto a posibles caídas de la red que irían desde instalar UPS's para el problema de voltaje hasta seguridad en los discos de los servidores

Durante las horas donde se tendrá demasiada carga de trabajo, la red deberá estar en óptimas condiciones para responder las diferentes solicitudes a los distintos servidores o tráfico de red.

4.2.11 Seguridad

Se establecerán políticas de seguridad en la UNSM-T, con respecto al acceso de recursos, estaciones de trabajo, servidores y opciones de seguridad a nivel usuario, permitiendo de esta manera controlar el uso de los recursos en la red para las diferentes áreas como en el área de Administración.

Estos posibles cambios que podría tener nuestra red en el futuro estarán soportados en el diseño del sistema de cableado estructurado de la red.

Debido a la gran cantidad de usuarios con los que se cuenta en la red de campus de la ciudad universitaria (más de 750 usuarios en fechas y horas punta, según reporte del servidor proxy) y además de contar con puertas de Acceso a, existe un elevado riesgo de sufrir ataques direccionados a detener los servicios brindados por la oficina de administración de la fibra óptica y que esto repercuta en la operatividad diaria de las diferentes oficinas académicas y administrativas de la UNSM-T, se ha implantado un servidor firewall que garantiza la seguridad en el caso de ataques dirigidos hacia la red interna desde internet.

Se cuenta con Listas de Control de Acceso, así como implementación de vlan para segmentar la red y elevar la seguridad de la información sensible.

En casos extremos, se cuenta con un servidor de respaldo, el mismo que está disponible para entrar en operación al momento de cualquier falla del servidor principal.

Gracias a la actual infraestructura de acceso a internet, se han configurados los servidores proxy para que puedan ser administrados de forma remota de tal manera que el administrador de dichos servidores no tenga la necesidad de estar presente físicamente para la solución de los problemas.

4.3 Diseño lógico

Esta red consiste de un “backbone” que cubre el campus de la Universidad y consta de la infraestructura de conectividad y transmisión (equipos y materiales).

La topología física diseñada para la red de la UNSM-T es de tipo estrella, por su facilidad de control y administración y fiabilidad.

La infraestructura de conectividad de dicha estrella, provee de la plataforma y la vía de transporte para la comunicación de voz, datos y video que integra toda la Universidad. Esta consiste de un canal principal de fibra óptica y los equipos de conectividad en cada uno de los pabellones, facultades u oficinas administrativas. El canal principal cubre el campus de la Universidad con una velocidad de transmisión mínima de 1 Gbit/s, la cual garantiza un tráfico fluido para la demanda actual y futura.

A partir del nodo central (core) de comunicaciones sale los enlaces hacia los nodos de distribución (borde), y desde estos nodos a los nodos de acceso.

Cada facultad o área administrativa dispone de puntos terminales de data, voz y video, enlazados al canal principal a través de cableado estructurado con capacidad de transmisión de 1gbps y equipos de conectividad de acceso (borde).

La fibra óptica es el medio de transmisión utilizado para el backbone en la red de datos de la ciudad Universitaria consta de un hilo muy fino de material transparente, vidrio o material plástico, por el que se envían pulsos de luz que representan los datos a transmitir.

Esta plataforma tecnológica los constituye en primer lugar un sistema de transporte (red de datos) eficiente y eficaz, que permita garantizar la comunicación entre los diferentes edificios de la ciudad universitaria.

El diseño lógico de la red, toma en cuenta que a medida que la red aumenta de tamaño, su administración se tornará más compleja, y al compartir los usuarios, el ancho de banda disponible y los dominios de “broadcast”, el tráfico de la red, así como la seguridad de acceso a la información, tiene el potencial impacto de colapsar o hacer inutilizable la red. Para ello, los equipos seleccionados tienen la capacidad de implementar esquemas vlan (redes virtuales), que proporcionen los medios adecuados para solucionar esta problemática, por medio de la agrupación realizada de una forma lógica en lugar de física.

A continuación se muestra el diagrama de la red de datos de la UNSM-T.

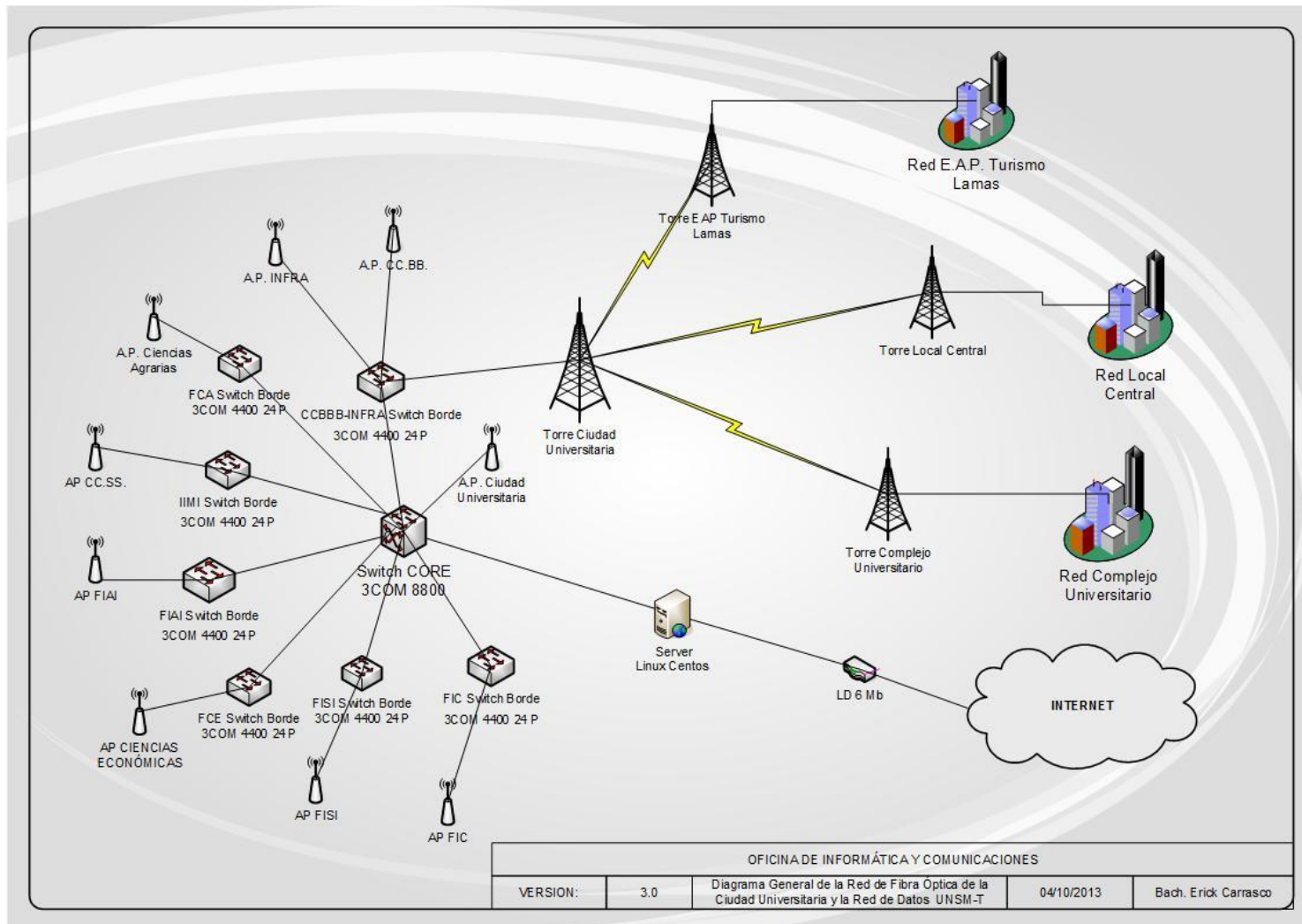


Figura 4.-Diagrama de la red de datos de la UNSM-T

4.4 Diseño físico.

La Red de Datos de la Ciudad Universitaria conecta a 11 edificios (Set/2013) en a través de diferentes medios: fibra óptica, enlaces inalámbricos punto a punto y punto-multipunto de 54 Mbps. Actualmente se está formulando la incorporación de dos nuevos edificios lo que hará un total de 13 edificios en red, por otra parte están en proceso de explotación varios enlaces de fibra óptica.

La red funciona con un esquema de VLANs (Redes virtuales).

Edificaciones conectadas a la red:

- Pabellón de Ingeniería de Sistemas e Informática.
- Pabellón de Ciencias de la Salud.
- Pabellón de Ingeniería Agroindustrial.
- Pabellón de Ciencias Agrarias.
- Pabellón de Ingeniería Civil y Arquitectura.
- Pabellón de Ciencias Económicas.
- Pabellón de Oficina de Infraestructura y Ciencias Básicas.
- Instituto de Investigación Materno Infantil.
- Garita de Control (para el reloj y las cámaras de seguridad).
- Pabellón del laboratorio de Ciencias Básicas.
- Pabellón de Ciencias Básicas y videoconferencia.

Otras instalaciones universitarias conectadas a la red mediante acceso inalámbrico:

- Pabellón Administrativo de la Facultad de Ingeniería Agroindustrial.
- Pabellón Facultad de Idiomas.
- Comedor Universitario.
- Cafetines.

A continuación se presenta un cuadro resumen, con la cantidad de equipos de la ciudad universitaria.

Tabla 8.-Distribución De Equipos De Cómputo UNSM-T

Ubicación	Switchs	Impresoras	PC
Fac. Ing. Sistemas e Informática	10	5	43
Fac. Ing. Agroindustrial	5	6	42
Fac. Ciencias de la Salud	6	4	35
Fac. Ing. Civil y Arquitectura	3	5	35
Fac. Ciencias Económicas	3	5	40
Idiomas	2	3	20
Pabellón Infraestructura	4	4	45
Instituto de Inv. Materno Infantil	4	5	35
Total	37	37	295

Para el caso de la fibra óptica, el proyecto incluye junto a la instalación del backbone de fibra óptica, un sistema de cableado estructurado y de networking para el Campus Universitario de la Universidad Nacional de San Martín-T, que comprende la instalación del Cableado Vertical con Fibra Óptica uniendo 08 Pabellones con el nodo central ubicado en Videoconferencia y el cableado horizontal de 200 puntos de red en las diferentes Facultades de la Ciudad Universitaria.

A continuación se muestran los planos de distribución de la red de datos del Campus Universitario de la Universidad Nacional de San Martín.

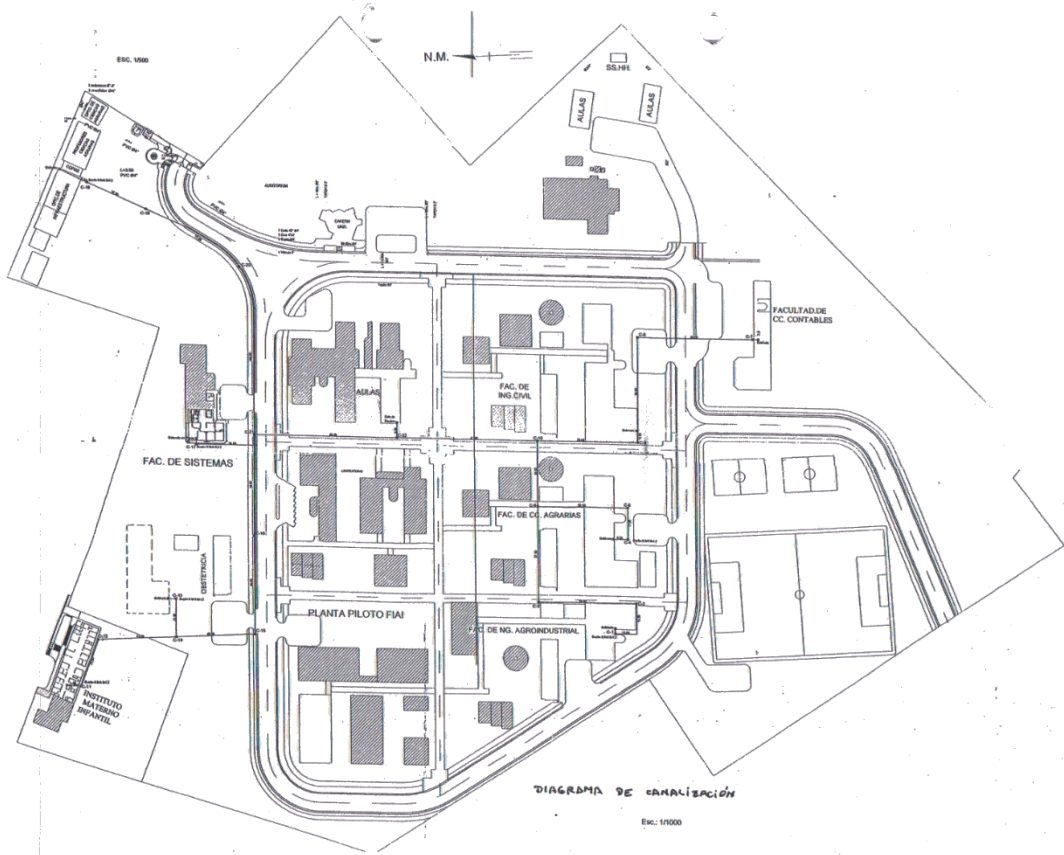


Figura 5.- Diagrama de Canalización de la UNSM-T

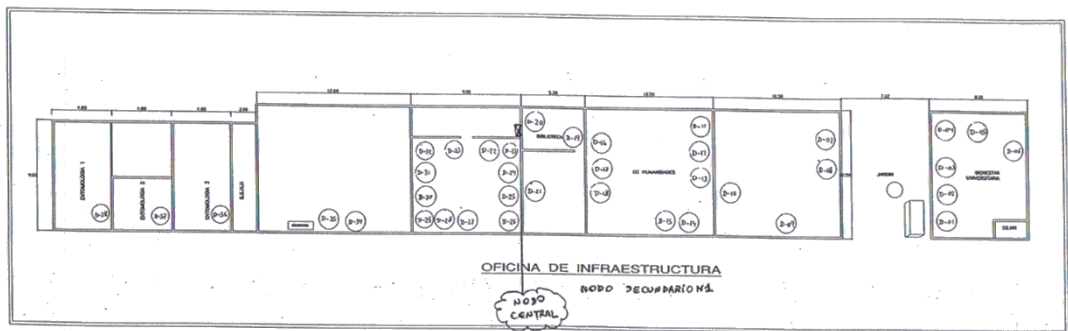
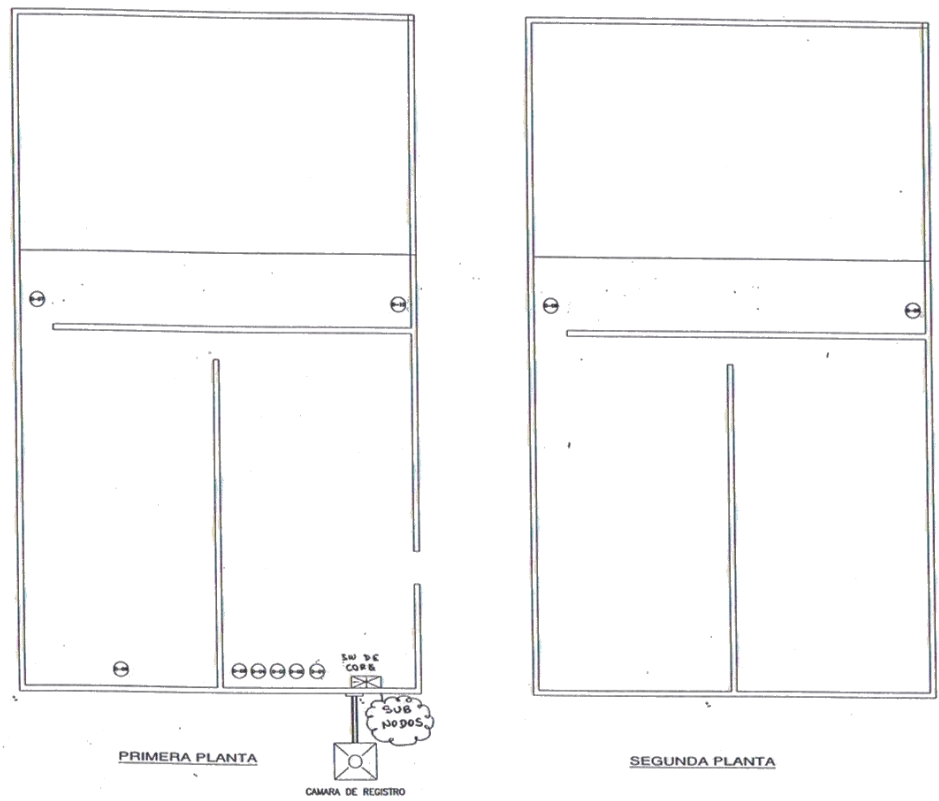



Figura 6.- Nodo Central



 : SW DE CORE
 2COM 8600

VIDEO CONFERENCIA
 NODO CENTRAL

Figura 7. Nodo central - switch core

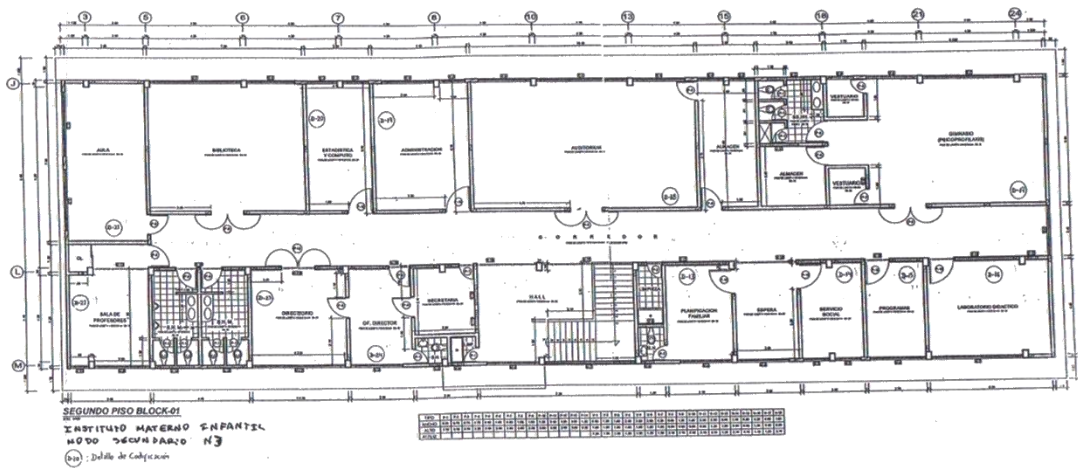


Figura 8.- Nodo Secundario - Instituto Materno Infantil

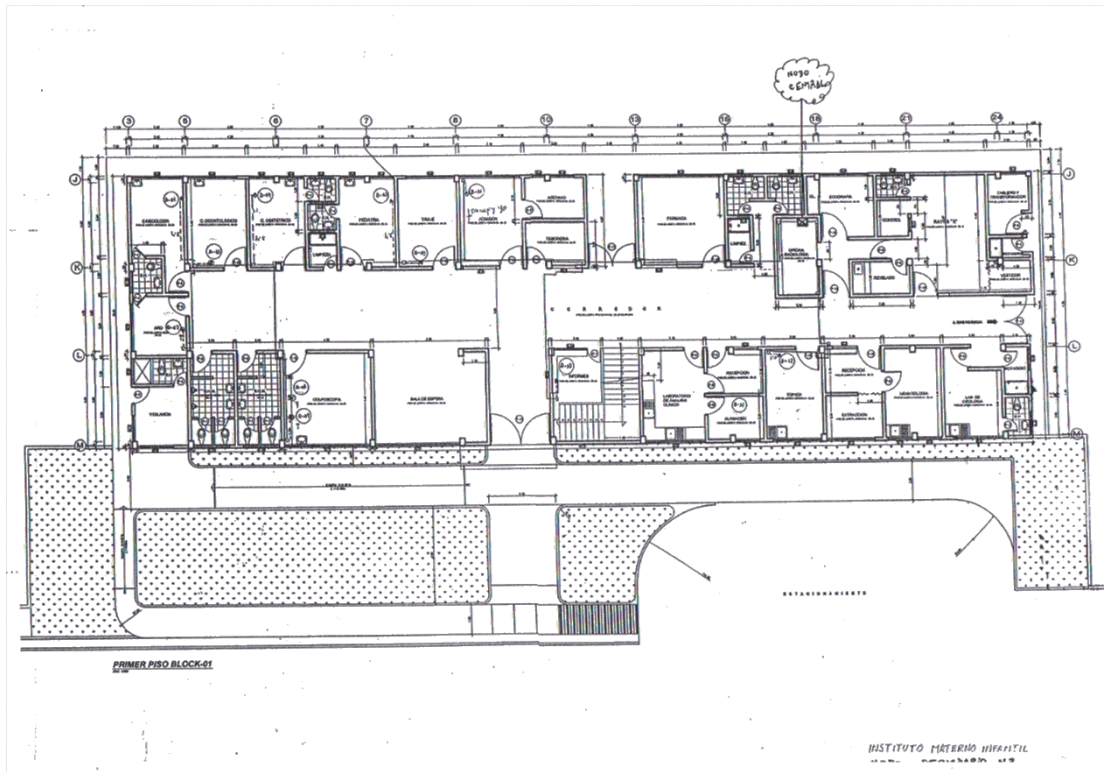


Figura 9.- Nodo Secundario switch de borde - Instituto Materno Infantil

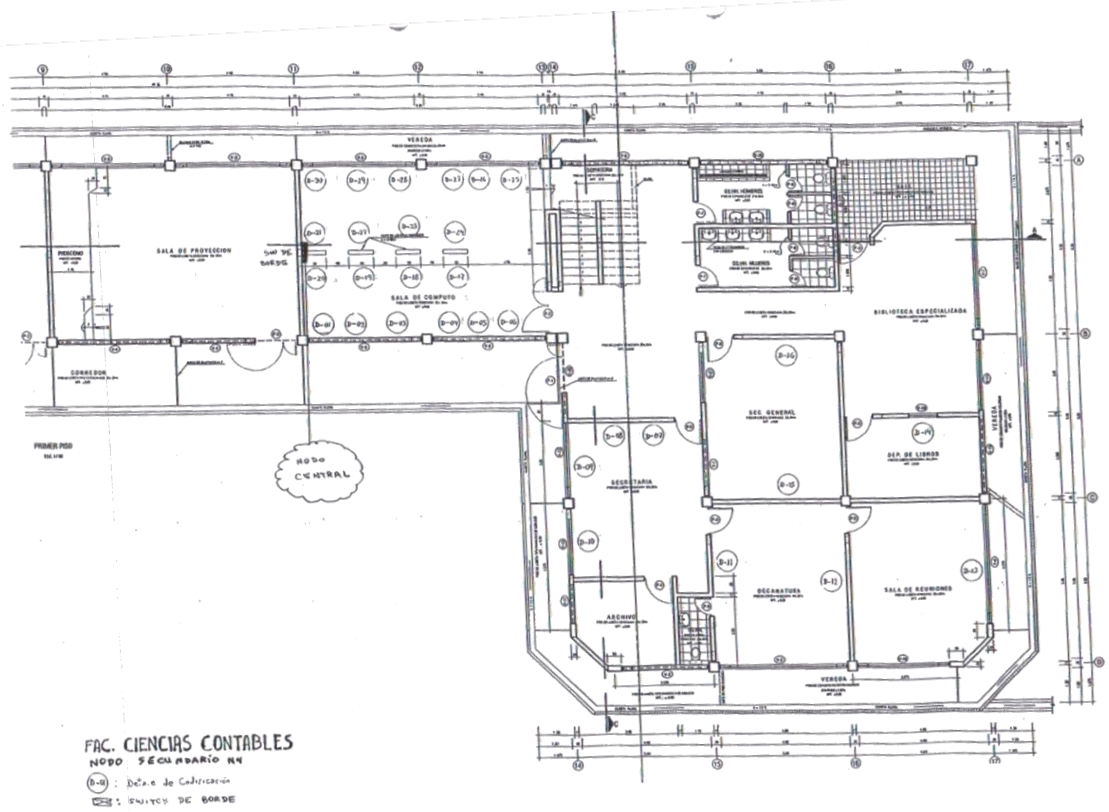


Figura 10.-Nodo Secundario - Ciencias Contables

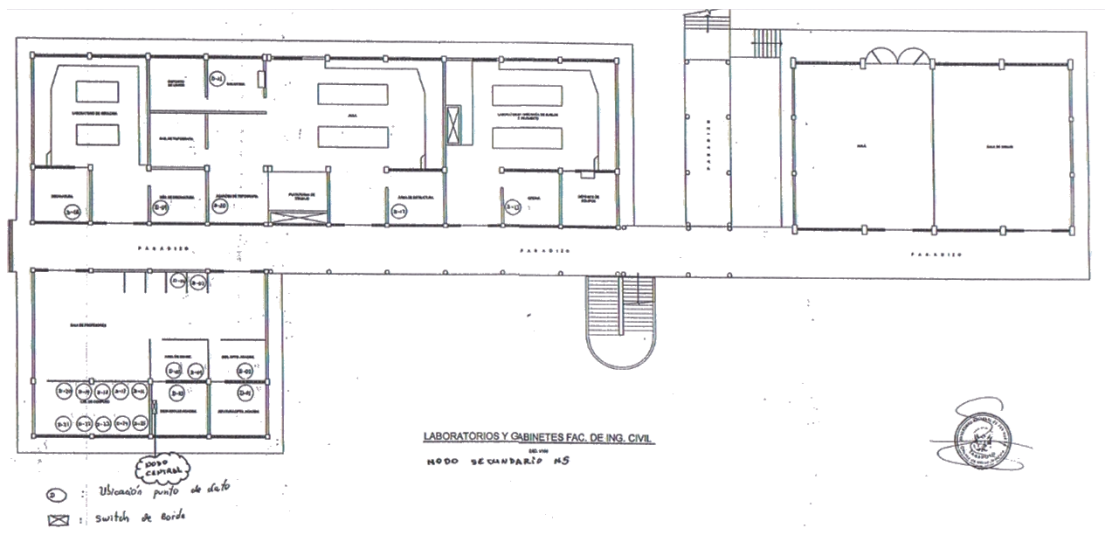


Figura 11.- Nodo Secundario switch de borde - ingeniería Civil

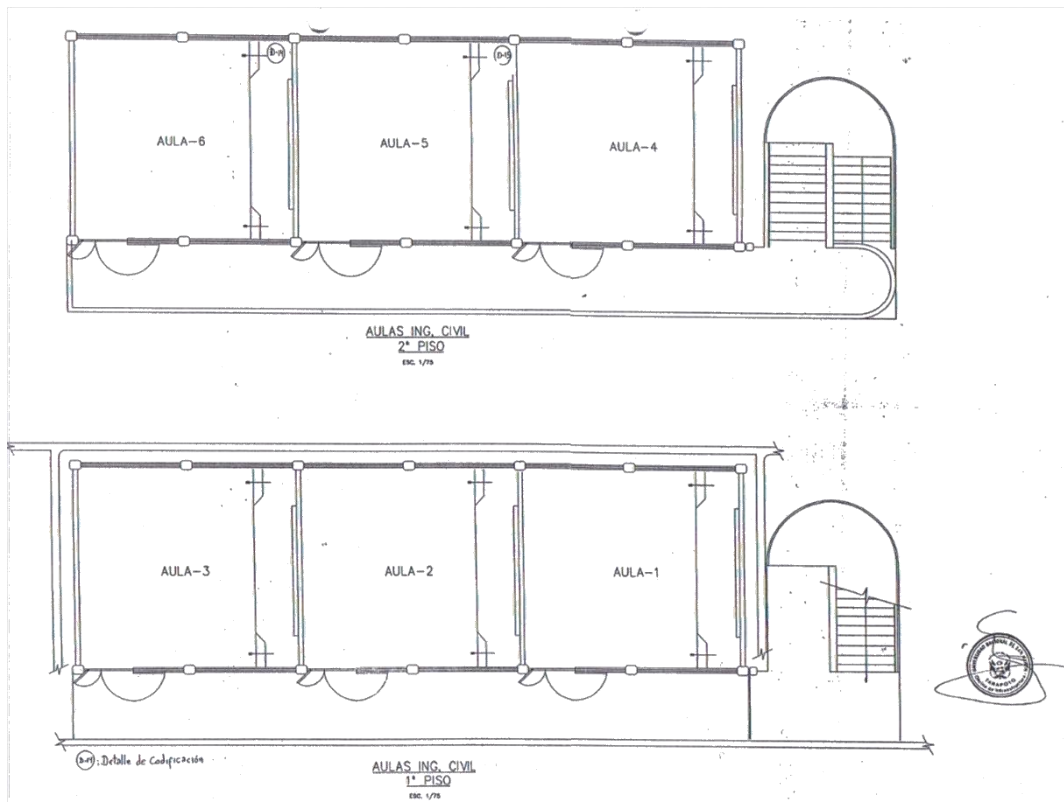


Figura 12.-Nodo Secundario detalle de codificación de Ingeniería Civil

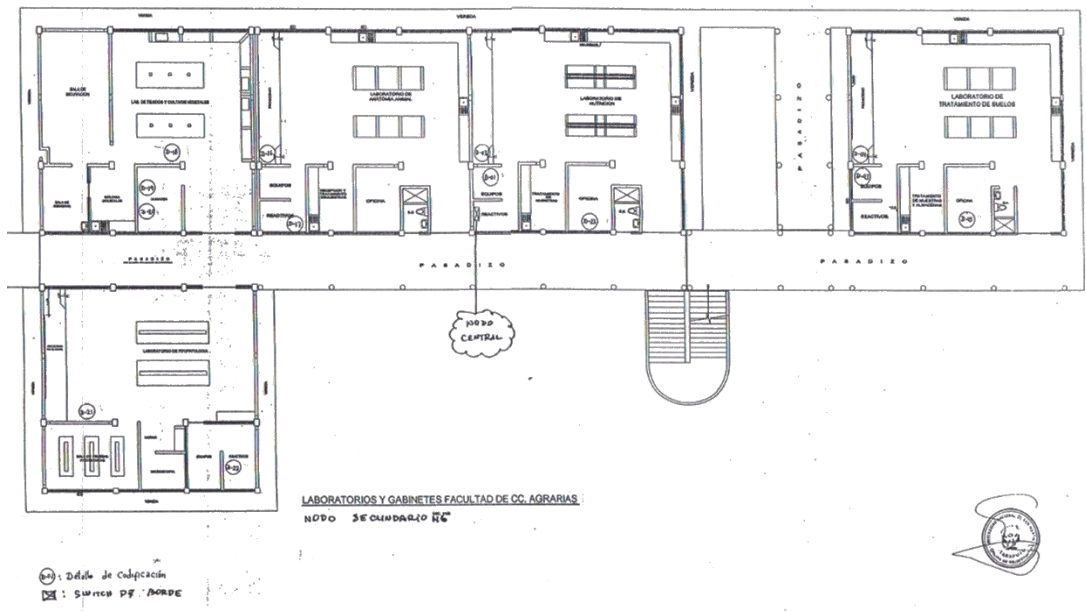


Figura 13.- Nodo Secundario switch de borde - Ciencias Agrarias

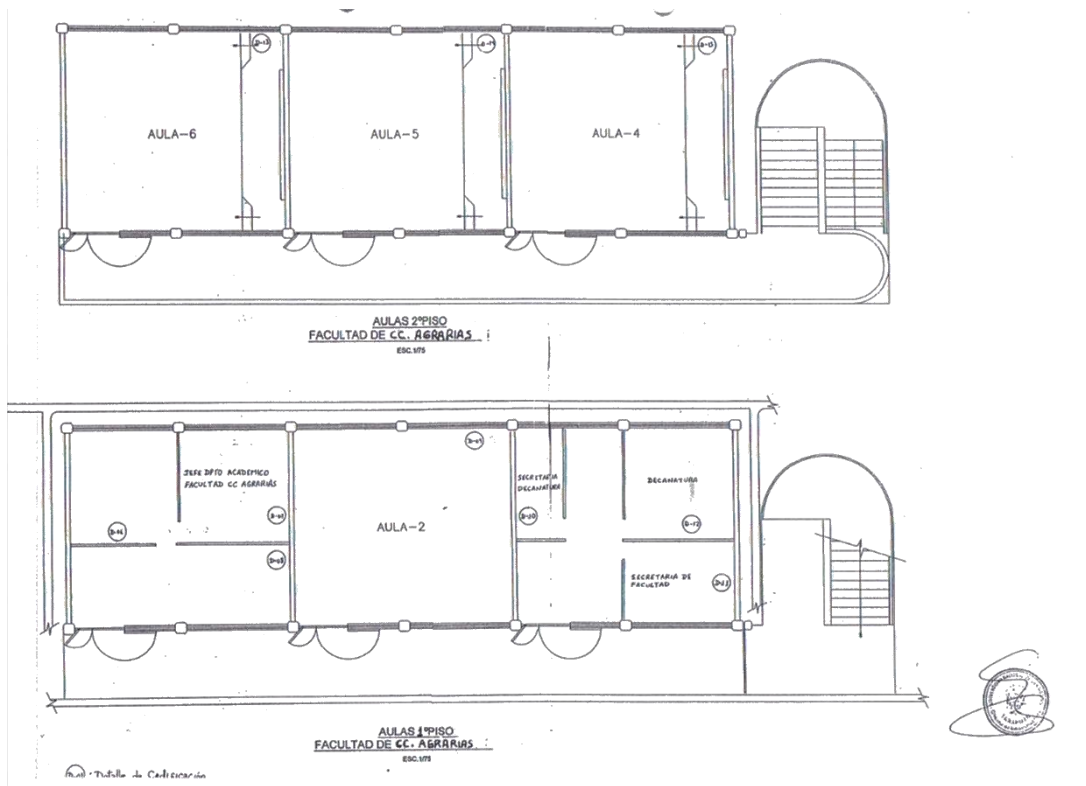


Figura 14.-Nodo Secundario detalle de codificación Ciencias Agrarias

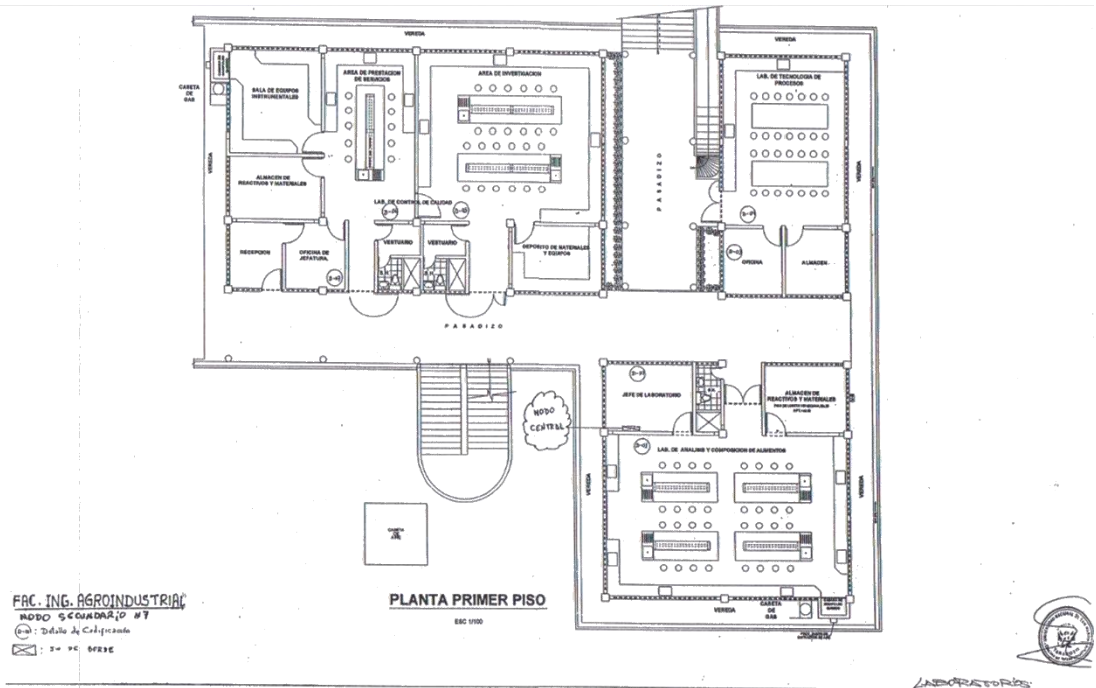


Figura 15.- Nodo Secundario switch de borde - Agroindustrial

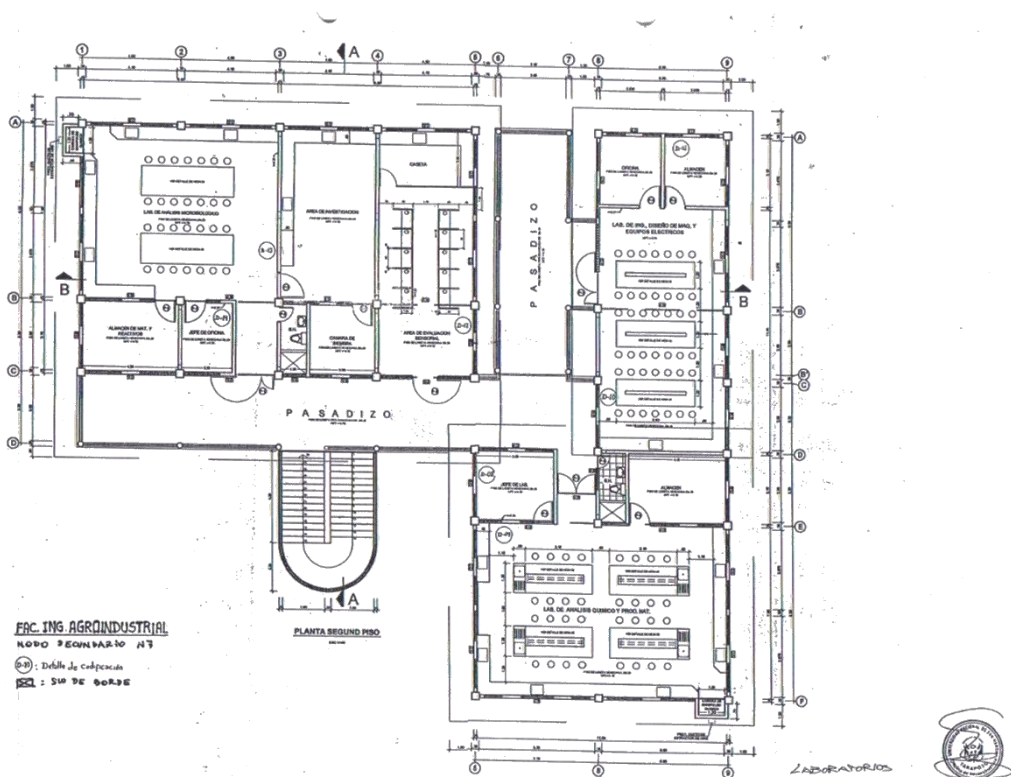
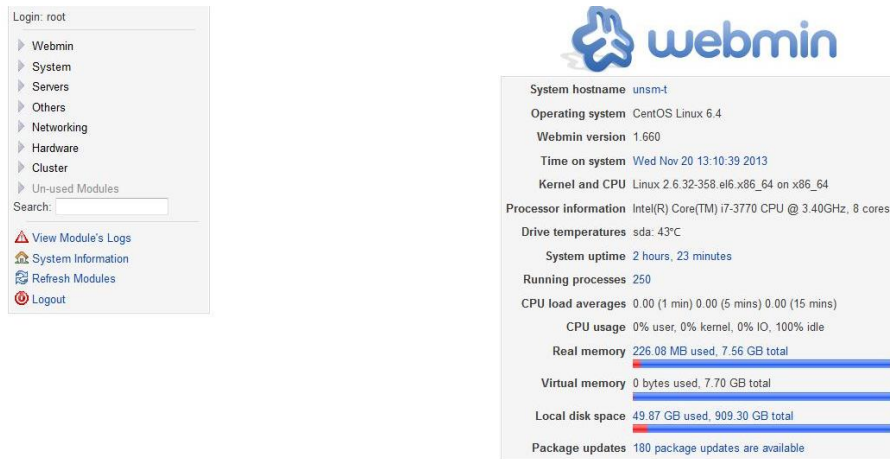


Figura 16.- Nodo Secundario switch de borde -Agroindustrial- Laboratorios

4.5 Testeo, optimización y documentación de la red

4.5.1 Servidor Proxy

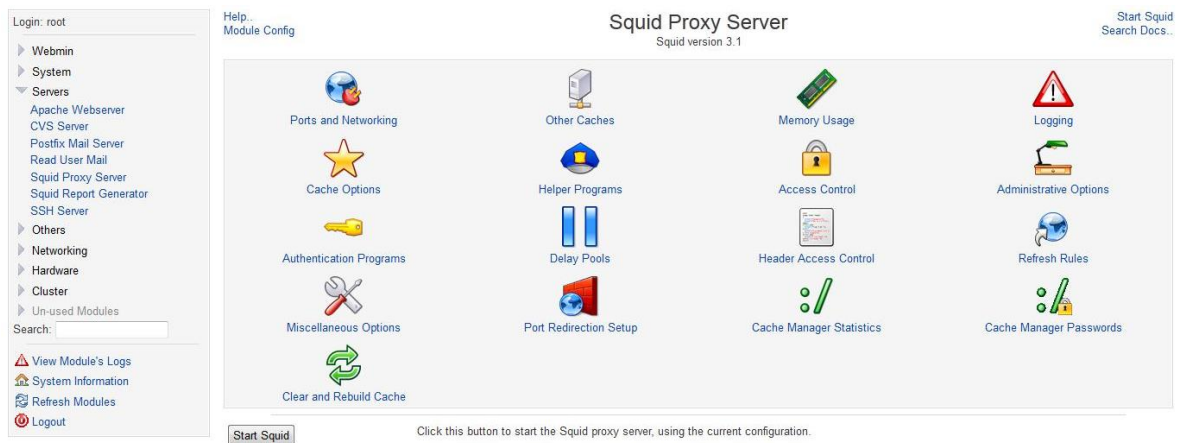
Una vez iniciado la pantalla principal de esta herramienta, nos muestra: nombre del servidor, la versión de sistema operativo, versión del webmin, información sobre el servidor como el procesador, el uso de la memoria y el espacio en disco.



The screenshot shows the Webmin interface. On the left is a navigation menu with categories like Webmin, System, Servers, and Hardware. The main area displays system information for 'unsm-t' running CentOS Linux 6.4. Key details include: Processor information (Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz, 8 cores), System uptime (2 hours, 23 minutes), Running processes (250), CPU load averages (0.00), CPU usage (0% user, 0% kernel, 0% IO, 100% idle), Real memory (226.08 MB used, 7.56 GB total), Virtual memory (0 bytes used, 7.70 GB total), Local disk space (49.87 GB used, 909.30 GB total), and Package updates (180 package updates are available).

Figura 17.-Gestor administrador del webmin

Proxy Server nos muestra varias opciones mediante las cuales se utiliza el squid



The screenshot shows the Squid Proxy Server administration interface. The title is 'Squid Proxy Server' with version 3.1. The interface is divided into several sections: 'Ports and Networking', 'Other Caches', 'Memory Usage', 'Logging', 'Cache Options', 'Helper Programs', 'Access Control', 'Administrative Options', 'Authentication Programs', 'Delay Pools', 'Header Access Control', 'Refresh Rules', 'Miscellaneous Options', 'Port Redirection Setup', 'Cache Manager Statistics', 'Cache Manager Passwords', and 'Clear and Rebuild Cache'. A 'Start Squid' button is located at the bottom left, with a note: 'Click this button to start the Squid proxy server, using the current configuration.'

Figura 18.-Opciones de administración del squid

Opción: ports and network, se utiliza para configurar los puertos de entrada y salida del servidor desde el puerto 3128 hasta el 80, así como la dirección de red del servidor.

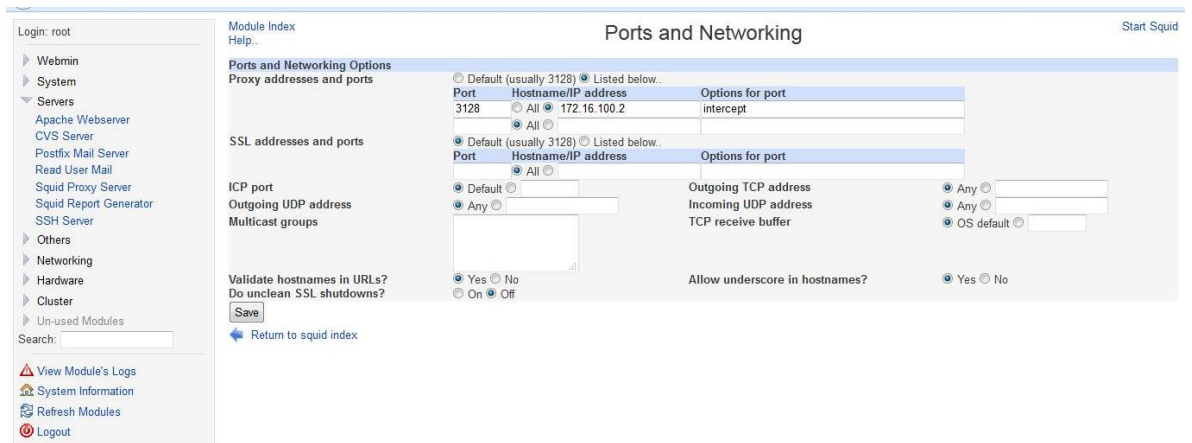


Figura 19.- Configuración de puertos de entrada y salida

Opción: other caches, se ingresa las direcciones mostradas, que estas busquen, directamente en la nube y no utilice la cache.



Figura 20.- Ingreso direcciones del cache

Opción: memory usage, muestra la cantidad de memoria utilizada y el máximo que tiene el equipo, así como el porcentaje del mismo.

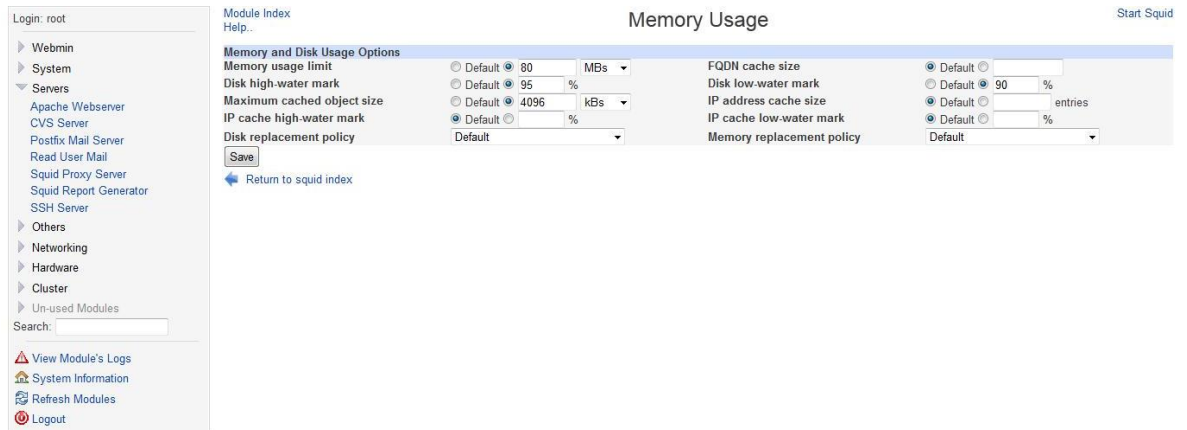


Figura 21.- Cantidad de memoria y porcentaje del mismo

Opción: logging, donde se guarda el archivo .log del servidor, el cual nos servirá para almacenar los datos históricos de los ingresos de cada ordenador a la red.

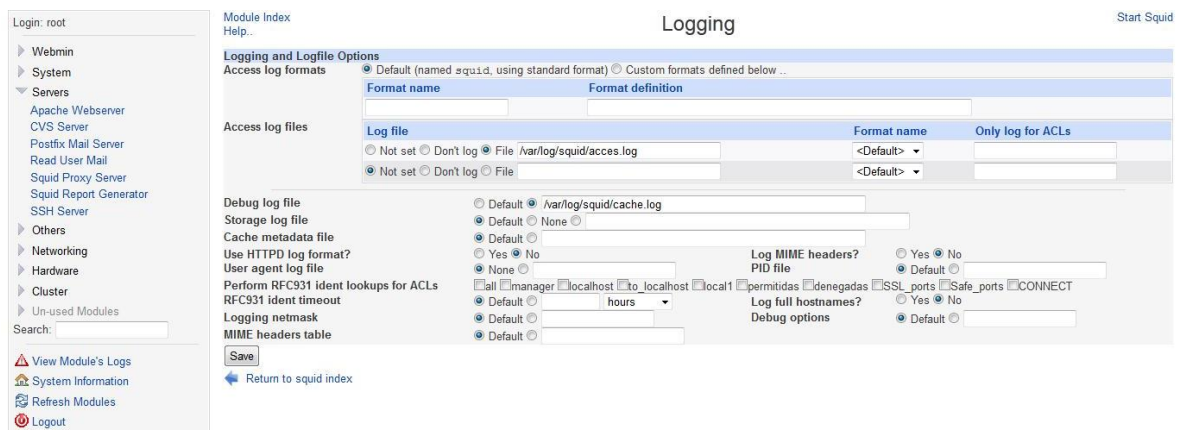


Figura 22.- Almacenar datos en el servidor

Opción: cache, utilizada para configurar, la dirección, el tamaño promedio que este debe tener, el tiempo que debe reiniciar y el tipo de usuario.

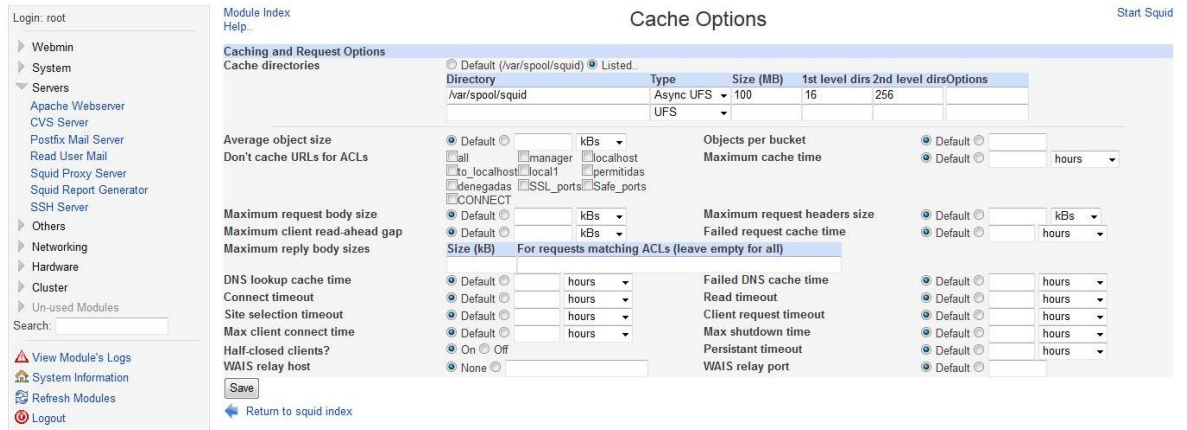


Figura 23.- Configurar la dirección del cache

Opción: helper programs, ayudante de programas de acuerdo al programa que se está utilizando, valores por defecto.

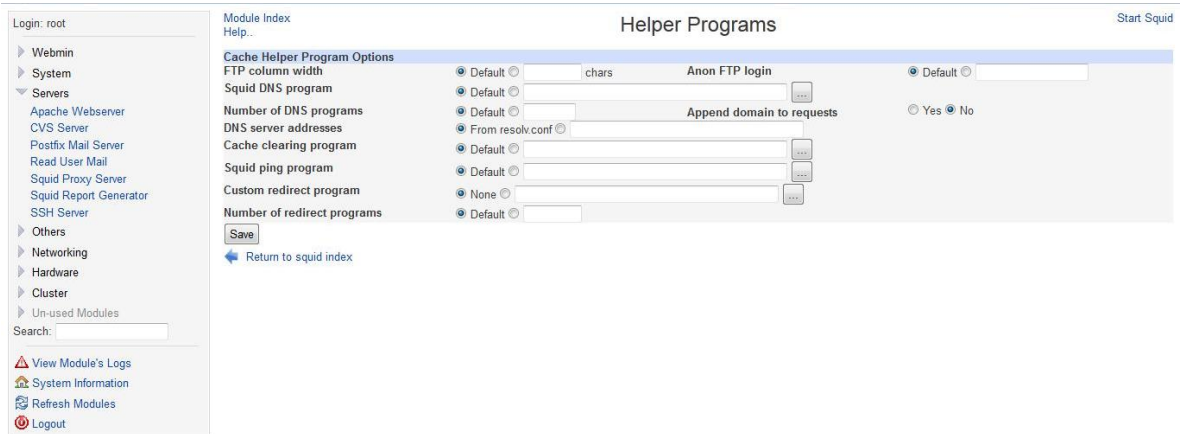


Figura 24.-Opción: Helper Programs

Opción: access control, nos muestra todos los accesos tanto para los alumnos como para los administrativos.

Module Index Help. Access Control Start Squid

Access control lists Proxy restrictions ICP restrictions External ACL programs Reply proxy restrictions

Name	Type	Matching..
manager	URL Protocol	cache_object
localhost	Client Address	127.0.0.1/32 ::1
to_localhost	Web Server Address	127.0.0.0/8 0.0.0.0/32 ::1
alumnos	Client Address	From file /etc/squid/listas/alumnos
permitidas	URL Regexp	From file /etc/squid/listas/permitidas
denegadas	URL Regexp	From file /etc/squid/listas/denegadas
SSL_ports	URL Port	443
Safe_ports	URL Port	80
Safe_ports	URL Port	21
Safe_ports	URL Port	443
Safe_ports	URL Port	70
Safe_ports	URL Port	210
Safe_ports	URL Port	1025-65535
Safe_ports	URL Port	280
Safe_ports	URL Port	488
Safe_ports	URL Port	591
Safe_ports	URL Port	777
CONNECT	Request Method	CONNECT
administrativos	Client Address	From file /etc/squid/listas/administrativos

Create new ACL Browser Regexp

Return to squid index

Figura 25.-Opción: Access Control

A continuación la lista de IP de las máquinas de los alumnos.

Module Index Edit ACL Start Squid

Client Address ACL alumnos

ACL Name	From IP	To IP	Netmask
alumnos	172.16.100.10		
	172.16.100.100		
	172.16.100.101		
	172.16.100.102		
	172.16.100.103		
	172.16.100.104		
	172.16.100.105		
	172.16.100.106		
	172.16.100.107		
	172.16.100.108		
	172.16.100.109		
	172.16.100.11		
	172.16.100.110		
	172.16.100.111		
	172.16.100.112		
	172.16.100.113		
	172.16.100.114		
	172.16.100.115		
	172.16.100.116		
	172.16.100.117		
	172.16.100.118		
	172.16.100.119		
	172.16.100.12		
	172.16.100.120		
	172.16.100.121		
	172.16.100.122		
	172.16.100.123		

Figura 26.-IP de las máquinas de los alumnos conectado

Las direcciones IP de los Administrativos dentro del servidor

The screenshot shows the Squid web interface for editing an ACL. The left sidebar contains a navigation menu with categories like Webmin, System, Servers, and Others. The main content area is titled 'Edit ACL' and shows the configuration for a 'Client Address ACL' named 'administrativos'. A table lists IP addresses from 172.16.200.24 to 172.16.200.28. Below the table, there are options for 'Failure URL' and 'Store ACL values in file', with the latter set to 'Separate file /etc/squid/listas/administrativos'. There are 'Save' and 'Delete' buttons and a 'Return to ACLs' link.

From IP	To IP	Netmask
172.16.200.24		
172.16.200.25		
172.16.200.26		
172.16.200.27		
172.16.200.28		

Figura 27.-IP de los Administrativos dentro del servidor

Lista de las direcciones denegadas por el servidor proxy

The screenshot shows the Squid web interface for editing an ACL. The left sidebar is the same as in the previous image. The main content area is titled 'Edit ACL' and shows the configuration for a 'URL Regexp ACL' named 'denegadas'. The 'Regular Expressions' field contains a list of patterns: '¡resonline', '0 12 73 116', '0-0-0-0', '1000culos', '1-1-1-1', '173.193.138.138', and '190.254.22.44'. Below the field, there are options for 'Failure URL' and 'Store ACL values in file', with the latter set to 'Separate file /etc/squid/listas/denegadas'. There are 'Save' and 'Delete' buttons and a 'Return to ACLs' link.

Figura 28.-Direcciones denegadas por el servidor proxy

Opción: administrative options, se configura el mensaje para los usuarios que intenten ingresar a páginas que se encuentran en la lista denegada además dejar un correo para enviar su comentario o sugerencias.

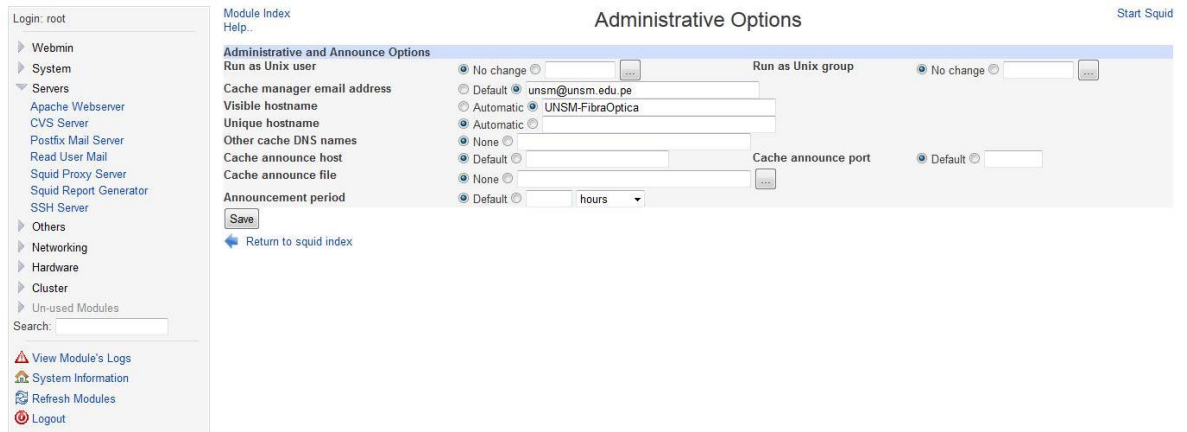


Figura 29.-Opción: Administrative Options

Opción: authentication programs, verifica las acciones de los programas, de acuerdo al uso, en esta oportunidad se utilizara los valores por defecto.

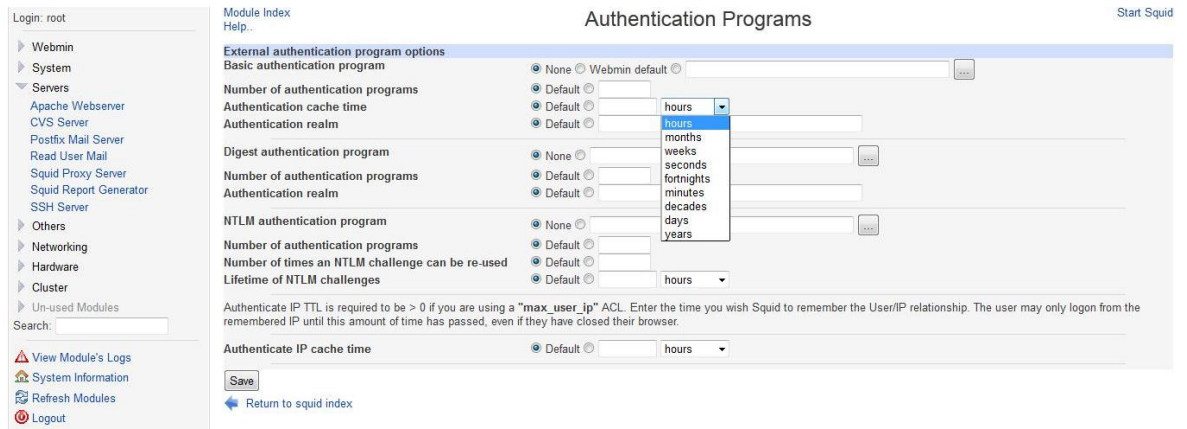


Figura 30.-Opción: authentication programs

Muestra la tarjeta de red existente y opción de elegir la red.

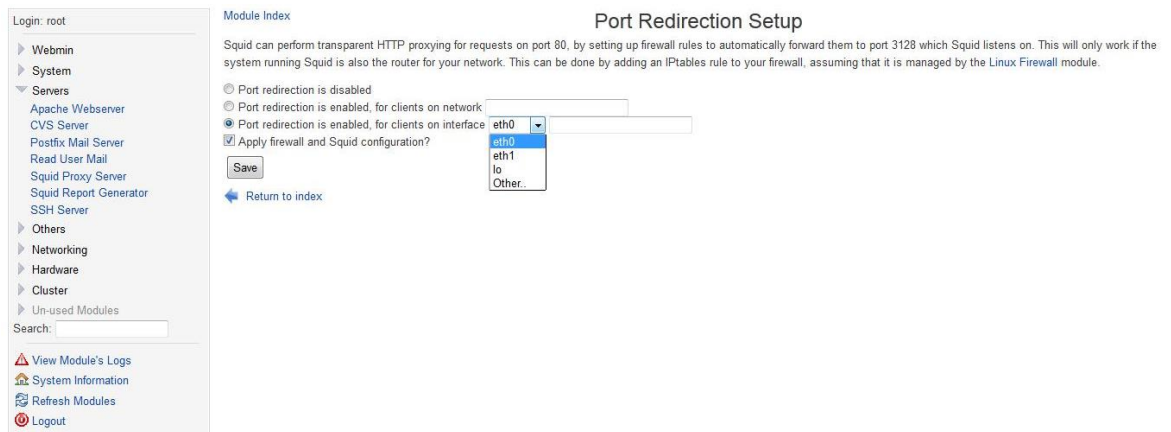


Figura 31.- Squid Report Generator

Una vez configurado el servidor proxy se ingresa al Squid Report Generator, donde se generan los reportes de manera manual, además esta herramienta cuenta con la opción de poder ver los reportes anteriormente generados

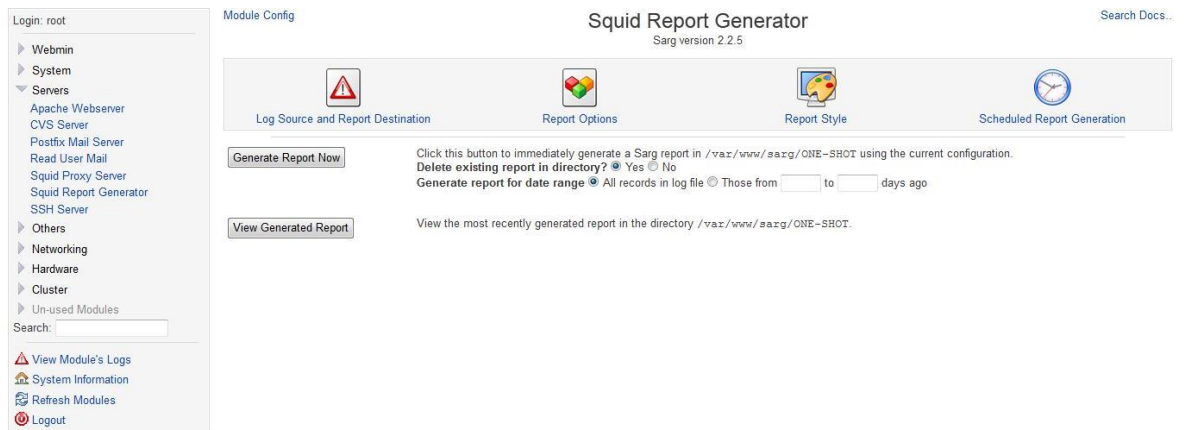


Figura 32.-Squid Report Generator

Los reportes muestra la lista con todos los reportes generados hasta ese momento ya sea de forma manual y manera periódica.

The screenshot shows the SARG (Squid Analysis Report Generator) web interface. On the left is a navigation menu with categories like Webmin, System, Servers, and Others. The main content area is titled 'Sarg Report' and displays a 'Squid User Access Report'. The report is a table with columns for 'ARCHIVO/PERIODO', 'FECHA CREACION', and 'USUARIOSBYTESPROMEDIO'. It lists several reports generated between 2013 and 2014, including details on file names, dates, and user statistics. A 'Return to module index' link is visible below the report list.

Figura 33.-Lista con todos los reportes generados

Al observar el reporte nos muestra las direcciones IP que se encuentran conectadas a la red, el tiempo de conexión. Las megas utilizadas, entre otros valores de suma importancia.

This screenshot shows a more detailed report from SARG. The main content area is titled 'Sarg Report' and displays a 'Squid User Access Report' for the period '2013Oct19-2013Oct19'. The report is classified by 'BYTES, reverse' and 'Topuser'. Below the title, there is a table titled 'Topsites' and 'Sitios y Usuarios Denegado'. The table has columns for 'NUM', 'USERID', 'CONEXIONBYTES', '%BYTESENTRADA-CACHE-SALIDA', 'TIEMPO UTILIZADO', 'MILISEC', and '%HORA'. It lists three top sites, with the first being 192.168.5.3. A 'TOTAL' row shows aggregate statistics. A 'Return to module index' link is also present.

Figura 34.-Direcciones IP que se encuentran conectadas a la red

Ahora nos muestra las direcciones de páginas que intentaron ingresar, si la dirección se encuentra en la lista de denegadas nos mostrara los intentos de ingreso a esa página.

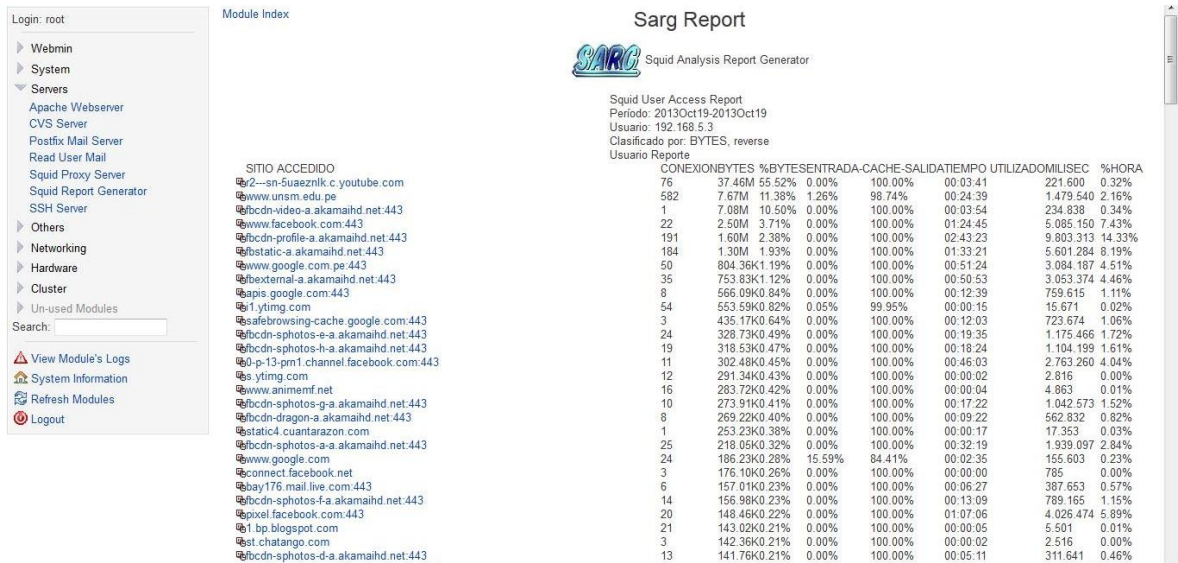


Figura 35.-Intento de ingresar a páginas denegadas

Observamos otros reportes generados desde nuestro servidor



Figura 36.- Reportes generados

Lista de reportes efectuados hasta las fecha.

The screenshot shows the Sarg Report interface. On the left is a navigation menu with options like Webmin, System, Servers, and others. The main content area displays the 'Sarg Report' title and a 'Squid User Access Report' table. The table lists various reports with columns for 'ARCHIVO/PERIODO', 'FECHA CREACION', and 'USUARIOS/BYTES/PROMEDIO'. Below the table, it says 'Generado por sarg-2.2.5 Mar-03-2008 el Nov/19/2013 03:09'.

ARCHIVO/PERIODO	FECHA CREACION	USUARIOS/BYTES/PROMEDIO
2013Nov17-2013Nov18	Tue Nov 19 02:51:31 PET 2013 6	1.75M 292.90K
2013Nov17-2013Nov18	1Tue Nov 19 01:51:07 PET 2013 6	1.75M 292.90K
2013Nov10-2013Nov17	Tue Nov 19 02:51:45 PET 2013 135	1.64G 12.21M
2013Nov10-2013Nov17	1Tue Nov 19 01:51:22 PET 2013 135	1.64G 12.21M
2013Oct27-2013Nov01	Fri Nov 1 11:17:17 PET 2013 141	1.31G 9.29M
2013Oct19-2013Oct25	4Fri Nov 1 11:02:06 PET 2013 1	1.64G 0
2013Oct19-2013Oct25	3Fri Nov 1 10:49:05 PET 2013 55	2.91G 53.01M
2013Oct19-2013Oct25	1Sun Oct 27 10:40:24 PET 2013 55	2.91G 53.01M
2013Oct19-2013Oct20	Sun Oct 20 09:58:48 PET 2013 47	1.91G 40.79M
2013Oct19-2013Oct25	Fri Nov 1 11:08:30 PET 2013 2744	1.49G 0
2013Oct19-2013Oct25	2Fri Nov 1 10:34:10 PET 2013 55	2.91G 53.01M

Figura 37.- Lista de reportes efectuados hasta las fecha

Reportes generados dependiendo de la configuración anteriormente

The screenshot shows a detailed Sarg Report. The left navigation menu is the same. The main content area displays the 'Sarg Report' title and a 'Squid User Access Report' for the period '2013Oct19-2013Oct19' by user '192.168.5.3'. The report is classified by 'BYTES, reverse' and shows a list of 'Usuario Reporte' with columns for 'SITIO ACCEDIDO', 'CONEXION', 'BYTES', '%BYTES', 'ENTRADA-CACHE', 'VALIDATIEMPO', 'UTILIZADOMI', 'USEC', and '%HORA'.

SITIO ACCEDIDO	CONEXION	BYTES	%BYTES	ENTRADA-CACHE	VALIDATIEMPO	UTILIZADOMI	USEC	%HORA
2--sn-5uaeznk.c.youtube.com	76	37.46M	55.52%	0.00%	100.00%	00:03:41	221.600	0.32%
www.untsm.edu.pe	582	7.67M	11.36%	1.26%	98.74%	00:24:39	1.479.540	2.16%
bcdn-video-a.akamaihd.net:443	1	7.08M	10.50%	0.00%	100.00%	00:03:54	234.839	0.34%
www.facebook.com:443	22	2.50M	3.71%	0.00%	100.00%	01:24:45	5.085.150	7.43%
bcdn-profile-a.akamaihd.net:443	191	1.60M	2.38%	0.00%	100.00%	02:43:23	9.803.313	14.33%
bstatic-a.akamaihd.net:443	184	1.30M	1.93%	0.00%	100.00%	01:33:21	5.601.284	8.19%
www.google.com.pe:443	50	804.36K	1.19%	0.00%	100.00%	00:51:24	3.084.187	4.51%
bexternal-a.akamaihd.net:443	35	753.83K	1.12%	0.00%	100.00%	00:50:53	3.053.374	4.46%
apis.google.com:443	8	566.09K	0.84%	0.00%	100.00%	00:12:39	759.615	1.11%
1.ytimg.com	54	553.59K	0.82%	0.05%	99.95%	00:00:15	15.671	0.02%
safebrowsing-cache.google.com:443	3	435.17K	0.64%	0.00%	100.00%	00:12:03	723.674	1.06%
bcdn-sphotos-e-a.akamaihd.net:443	24	328.73K	0.49%	0.00%	100.00%	00:19:35	1.175.466	1.72%
bcdn-sphotos-h-a.akamaihd.net:443	19	318.53K	0.47%	0.00%	100.00%	00:18:24	1.104.199	1.61%
0-p-13-prn1.channel.facebook.com:443	11	302.48K	0.45%	0.00%	100.00%	00:46:03	2.763.260	4.04%
s.ytimg.com	12	291.34K	0.43%	0.00%	100.00%	00:00:02	2.816	0.00%
www.animamf.net	16	283.72K	0.42%	0.00%	100.00%	00:00:04	4.863	0.01%
bcdn-sphotos-g-a.akamaihd.net:443	10	273.91K	0.41%	0.00%	100.00%	00:17:22	1.042.573	1.52%
bcdn-dragon-a.akamaihd.net:443	8	269.22K	0.40%	0.00%	100.00%	00:09:22	562.832	0.82%
static4.cuantarazon.com	1	253.23K	0.38%	0.00%	100.00%	00:00:17	17.353	0.03%
bcdn-sphotos-a-a.akamaihd.net:443	25	218.05K	0.32%	0.00%	100.00%	00:32:19	1.939.097	2.84%
www.google.com	24	186.23K	0.28%	15.59%	84.41%	00:02:35	155.603	0.23%
connect.facebook.net	3	176.10K	0.26%	0.00%	100.00%	00:00:00	785	0.00%
ay176.mail.live.com:443	6	157.01K	0.23%	0.00%	100.00%	00:06:27	387.653	0.57%
bcdn-sphotos-f-a.akamaihd.net:443	14	156.98K	0.23%	0.00%	100.00%	00:13:09	739.165	1.15%
pixel.facebook.com:443	20	148.46K	0.22%	0.00%	100.00%	01:07:06	4.026.474	5.89%
1.bp.blogspot.com	21	143.02K	0.21%	0.00%	100.00%	00:00:05	5.501	0.01%
st.chatango.com	3	142.36K	0.21%	0.00%	100.00%	00:00:02	2.516	0.00%
bcdn-sphotos-d-a.akamaihd.net:443	13	141.76K	0.21%	0.00%	100.00%	00:05:11	311.641	0.46%

Figura 38.- Direcciones a las cuales se decidió ingresar.

Observamos el reporte general de las máquinas que se conectaron a internet por medio de este servidor proxy.

Observamos el reporte general de las máquinas que se conectaron a internet por medio de este servidor proxy.

Topsites									
Sitios y Usuarios									
Bajados									
Denegado									
NUM	USERID	CONEXION	BYTES	%BYTESEN	TRADADA-CACHE-	SALIDATIEMPO	UTILIZADO	MILISEC	%HORA
1	172.16.100.1281.39K	72.33M	4.39%	4.49%	95.51%	02:00:57	7.257.099	0.89%	
2	172.16.100.1271.99K	65.25M	3.96%	3.89%	96.11%	10:08:00	36.480.486	4.48%	
3	172.16.100.36.6.26K	45.16M	2.74%	52.03%	47.97%	04:18:47	15.527.306	1.91%	
4	172.16.100.98.7.95K	43.09M	2.61%	64.02%	35.98%	04:40:24	16.824.567	2.07%	
5	172.16.100.38.1.36K	39.79M	2.41%	4.63%	95.37%	02:46:58	10.018.833	1.23%	
6	172.16.100.94.6.53K	35.34M	2.14%	69.31%	30.69%	03:22:37	12.157.333	1.49%	
7	172.16.100.54.2.29K	27.54M	1.67%	31.92%	68.08%	03:02:37	10.957.834	1.35%	
8	172.16.100.1102.27K	26.74M	1.62%	19.72%	80.28%	02:11:48	7.908.148	0.97%	
9	172.16.100.95.1.28K	26.42M	1.60%	7.45%	92.55%	02:52:11	10.331.951	1.27%	
10	172.16.100.40.1.75K	24.52M	1.49%	10.99%	89.01%	01:34:49	5.689.525	0.70%	
11	172.16.100.1031.44K	24.33M	1.48%	26.93%	73.07%	03:37:58	13.078.032	1.61%	
12	172.16.100.1072.19K	24.31M	1.47%	30.80%	69.20%	03:42:16	13.336.998	1.64%	
13	172.16.100.93.1.54K	24.30M	1.47%	7.67%	92.33%	04:50:21	17.421.402	2.14%	
14	172.16.100.74.5.11K	23.55M	1.43%	11.34%	88.66%	03:20:55	12.055.496	1.48%	
15	172.16.100.33.1.29K	22.39M	1.36%	7.58%	92.42%	02:36:00	9.360.005	1.15%	
16	172.16.100.50.1.48K	21.68M	1.32%	13.16%	86.84%	03:16:13	11.773.618	1.45%	
17	172.16.100.71.1.26K	21.03M	1.28%	12.80%	87.20%	02:53:42	10.422.269	1.28%	
18	172.16.100.44.1.16K	20.79M	1.26%	8.23%	91.77%	02:18:49	8.329.371	1.02%	
19	172.16.100.1311.13K	20.32M	1.23%	12.15%	87.85%	02:26:30	8.790.258	1.08%	
20	172.16.100.91.4.80K	20.02M	1.21%	11.19%	88.81%	03:20:47	12.047.236	1.48%	
21	172.16.100.41.1.41K	19.48M	1.18%	10.38%	89.62%	03:11:35	11.495.250	1.41%	
22	172.16.100.1051.51K	19.00M	1.15%	43.16%	56.84%	02:03:36	7.415.521	0.91%	
23	172.16.100.1081.34K	18.33M	1.11%	8.06%	91.94%	02:29:58	8.998.896	1.11%	
24	172.16.100.46.1.09K	18.28M	1.11%	10.07%	89.93%	01:55:01	6.901.411	0.85%	
25	172.16.100.64.885	18.21M	1.10%	8.80%	91.20%	02:21:16	8.476.761	1.04%	
26	172.16.100.66.1.26K	17.82M	1.08%	6.28%	93.72%	02:25:54	8.754.186	1.08%	
27	172.16.100.35.1.37K	17.79M	1.08%	11.03%	88.97%	02:28:55	8.935.697	1.10%	
28	172.16.100.34.1.15K	17.32M	1.05%	25.22%	74.78%	02:38:14	9.494.922	1.17%	
29	172.16.100.65.1.25K	17.22M	1.04%	6.16%	93.84%	03:10:20	11.420.919	1.40%	
30	172.16.100.72.1.72K	16.91M	1.02%	19.33%	80.67%	03:44:38	13.478.740	1.66%	
31	172.16.100.30.1.03K	16.78M	1.02%	17.02%	82.98%	02:19:58	8.398.407	1.03%	
32	172.16.100.45.1.36K	16.53M	1.00%	16.67%	83.33%	02:21:23	8.483.922	1.04%	
33	172.16.100.62.1.10K	16.44M	1.00%	10.90%	89.10%	02:22:13	8.533.171	1.05%	
34	172.16.100.1001.57K	16.19M	0.98%	15.42%	84.58%	03:20:45	12.045.844	1.48%	
35	172.16.100.60.1.46K	15.99M	0.97%	31.78%	68.22%	02:26:39	8.799.275	1.08%	

Figura 39.- Reporte general que se conectaron a internet

4.5.2 Corta fuego.

A continuación se muestra la configuración del servicio iptables, el mismo que se encarga de garantizar la seguridad de la red interna, evitando que paquetes no autorizados accedan a esta red y generen problemas derivados de la denegación de servicios.

Configuración de los paquetes entrantes. Donde se puede apreciar los puertos a los cuales se tienen acceso, previa configuración de los servicios brindados.

Incoming packets (INPUT) - Only applies to packets addressed to this host			
Select all Invert selection			
Action	Condition	Move	Add
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 7777	↓ ↓↑
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 7777	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is ICMP	↓↑ ↓
<input type="checkbox"/>	Accept	If input interface is lo	↓↑ ↓
<input type="checkbox"/>	Accept	If state of connection is RELATED,ESTABLISHED	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 3128	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 3128	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 80	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 80	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is UDP and input interface is eth0 and destination port is 67:68 and source port is 67:68 and state of connection is NEW	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 25000	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 25000	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 3350	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 3350	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 22	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 22	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 443	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 443	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and input interface is eth0 and destination port is 1194	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and input interface is eth0 and destination port is 1194	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 10000	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and destination port is 443	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and source port is 443	↓↑ ↓
<input type="checkbox"/>	Accept	If protocol is TCP and input interface is eth0 and destination port is 443 and state of connection is NEW,ESTABLISHED	↓↑ ↓

Figura 40.- Paquetes Entrantes

Configuración del comportamiento de los paquetes reenviados. Permitiendo al servidor funcionar como puerta de reenvío de los paquetes desde y hacia la red interna.



Figura 41.- Paquetes Reenviados

A continuación se muestra la configuración de los paquetes salientes.

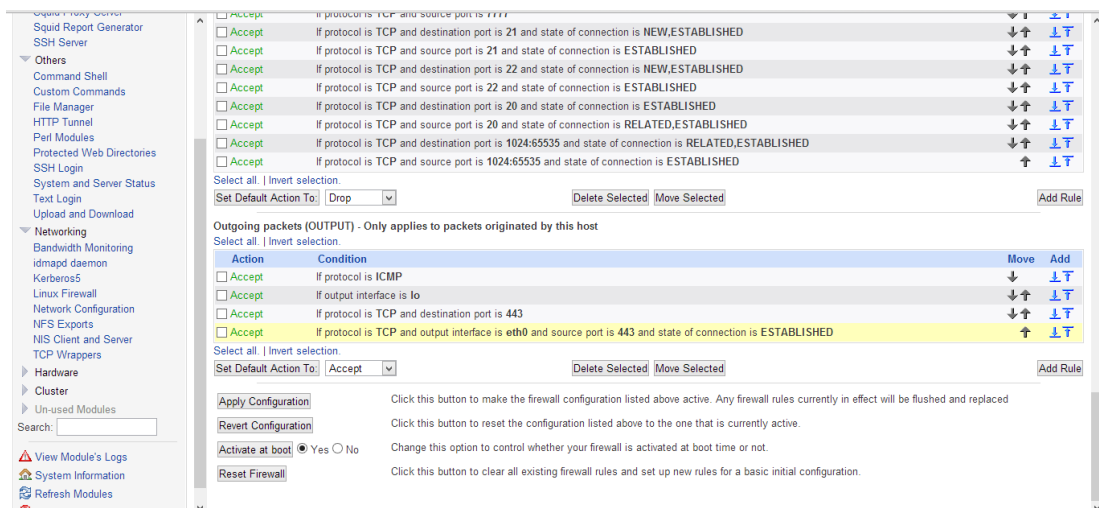


Figura 42.- Paquetes salientes

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

Gracias a la ejecución de este proyecto, se logró implantar el plan de mejora del ancho de banda de internet y seguridad en la red de datos basado en la metodología del diseño descendente de redes top down en la Universidad Nacional de San Martín -Tarapoto.

Al aplicar la metodología del diseño descendente, en la fase 4: Testeo, optimización y documentación de la red, se lograron definir las listas de control de acceso y filtrado de contenido, permitiendo definir los límites de acceso y uso del servicio de internet y todos los relacionados al mismo.

Con respecto a la seguridad, en la etapa de Desarrollo de la Solución, se construye un producto que permite garantizar la seguridad de la red interna de ataques de la red externa mediante un cortafuegos software que se personalizó de acuerdo a los requerimientos de la red misma (por ejemplo con la apertura de puertos para el uso del SIAF y páginas web que personalizan los puertos configurados por defecto).

Se conoce ahora más sobre la metodología del diseño descendente de redes, con lo cual se pueden realizar proyectos orientados a otras instituciones que tengan requerimientos parecidos a la Universidad Nacional de San Martín – Tarapoto.

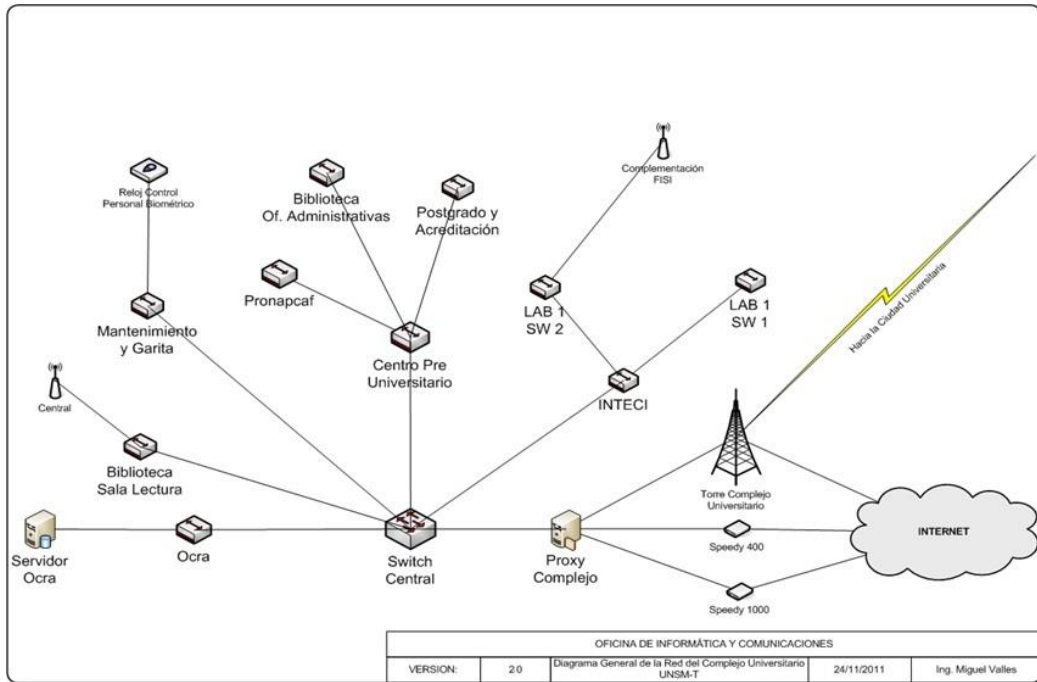
Se recomienda utilizar esta investigación como base teórica y práctica para próximas investigaciones.

REFERENCIAS

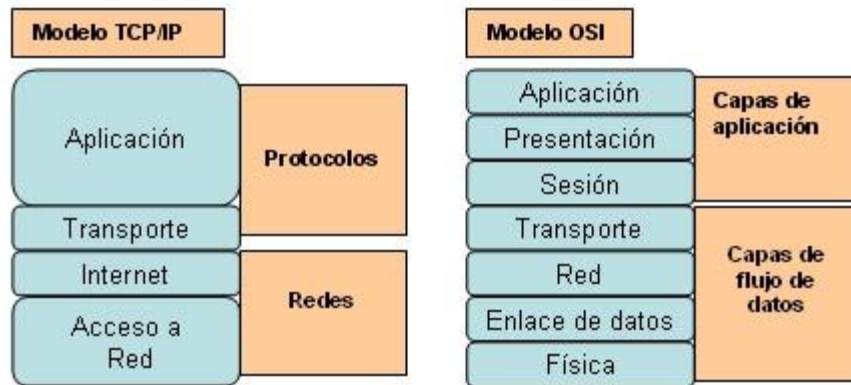
- Adrformacion. "Comparación entre el modelo OSI y el TCP/IP". 2013. España
Formato html. Disponibilidad libre en:
<http://www.adrformacion.com/cursos/wserver082/leccion1/tutorial6.html>
- BARROS D, Joel "Configuración de Squid: Opciones básicas". 2013. México.
formato html. Disponibilidad libre en:<<http://www.alcancelibre.org/>>
- DREW H, Prentice P Redes con Microsoft TCP/IP, 3er edición
- FONSECA, Rodrigo y Patricio Esteban ÁVILA SANTACRUZ. "Optimización y administración para el uso del Internet en la red de Policia de Migración a nivel nacional utilizando herramientas bajo Linux". 1era Edición. Editorial Escuela Politécnica del Ejército. 2010. España.
- Gobierno de Chile – Ministerio de educación. "Orientaciones Técnicas Implementación de Red de Datos y Conectividad a Internet". 2010. Chile.
- HUERTA, Marcos. "Metodología Top Down". 3era Edición. Editorial Pearson. México.
- OPPENHEIMER, Priscilla. "Top-Down Network Design". 3era Edición. Editor Pearson Education, 2010. USA.
- PEREZ, Fernando "Administración de redes en la Universidad Privada Antenor Obrego" 2013 Trujillo Formato html. Disponibilidad libre en:
http://share.pdfonline.com/b83a57c1a4324bbd82bbc5e1448f5525/VPN_I.E.P%20WILLIAM%20HARVEY%20-%20II.htm
- ROUSSKOV, Alex "Squid: Optimising Web Delivery". 2013. USA. formato html.
Disponibilidad libre en: <<http://www.squid-cache.org>>
- STALLINGS, W.: "Comunicaciones y Redes de Computadores". Séptima Edición, España, Pearson Educación, 2004.
- TANENBAUM, A.S. "Redes de Computadoras". Cuarta Edición, México, Pearson Educación, 2003.
- VALLES M, Entrevista "Administrador de la red Universidad Nacional San Martin" 2013 Tarapoto.

ANEXOS

Anexo 1.- Diagrama General de la Red del campus UNSM-T



Anexo 2.- Comparación entre el modelo OSI y el TCP/IP



Anexo 3.- Constancia de reconocimiento y respaldo de la UNSM-T

“UNIVERSIDAD NACIONAL DE SAN MARTÍN”.- TARAPOTO

EL SEÑOR ING. MIGUEL ÁNGEL VALLES CORAL, ADMINISTRADOR DE LA FIBRA ÓPTICA DE LA UNIVERSIDAD NACIONAL DE SAN MARTIN, QUE SUSCRIBE:

HACE CONSTAR:

Que, el señor, Don **Erick Carrasco Guerrero**, identificado con DNI N° 45077502, presto servicio **AD HONOREM** como **Bachiller en Ingeniería de Sistemas**, desarrollando un Plan de mejora del ancho de banda de internet y seguridad aplicados a la red de datos en la “**UNIVERSIDAD NACIONAL DE SAN MARTÍN**”, habiendo concluido satisfactoriamente la implementación de esta herramienta.

Se expide la presente constancia a solicitud del interesado.

Morales, 20 de noviembre de 2013



Anexo 4.-Evaluación Costo Beneficio

			MARZO	ABRIL	MAYO	JUNIO		AGOSTO	SEPTIEMBRE	OCTUBRE	NOVIEMBRE	COSTO
REAL	13	TRABAJADORES	9,750.00	9,750.00	9,750.00	9,750.00		9,750.00	9,750.00	9,750.00	9,750.00	78,000.00
PROPUESTA	3	TRABAJADORES	2,250.00	2,250.00	2,250.00	2,250.00		2,250.00	2,250.00	2,250.00	2,250.00	18,000.00
AHORRO TOTAL	10	TRABAJADORES	7,500.00	7,500.00	7,500.00	7,500.00		7,500.00	7,500.00	7,500.00	7,500.00	60,000.00

PROPUESTA DE TRABAJO

	SEPTIEMBRE	OCTUBRE	NOVIEMBRE	TOTAL
MATERIALES	85.73	85.73	85.73	257.20
HARDWARE	1,147.50	1,147.50	1,147.50	3,442.50
SOFTWARE				
HONORARIOS	1000	1000	1000	3,000.00
INVESTIGADOR				
COSTO TOTAL	2,233.23	2,233.23	2,233.23	6,699.70

BENEFICIO DE REDUCIR DE 13 A 03 TRABAJADORES PARA EL AÑO 2013

AHORRO TOTAL	60,000.00
(-) PRPUESTA DE TRABAJO	-6,699.70
BENEFICIO	53,300.30

