

UNIVERSIDAD TÉCNICA DE COTOPAXI
UNIDAD ACADÉMICA DE CIENCIAS DE LA
INGENIERÍA Y APLICADAS



TEMA:

“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL Y BALANCEO DE CARGA, EN ROUTERS MIKROTIK CON CALIDAD DE SERVICIO (QOS), PARA LA RED WLAN (Wireless Local Area Network) DE LOS SISTEMAS DE COMUNICACIÓN LATACUNGA”.

**PREVIO LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
INFORMÁTICA Y SISTEMAS COMPUTACIONALES**

TESISTAS:

- Chancusig Maisincho Omar Estuardo
- Martínez Chilibingua Luis Salomón

DIRECTOR:

Ing. Patricio Navas

Latacunga – Ecuador

2012

AUTORÍA

Las ideas, conclusiones y recomendaciones vertidas en el presente proyecto, cuyo tema es **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL Y BALANCEO DE CARGA, EN ROUTERS MIKROTIK CON CALIDAD DE SERVICIO (QOS), PARA LA RED WLAN (Wireless Local Area Network) DE LOS SISTEMAS DE COMUNICACIÓN LATACUNGA”**. Son exclusiva responsabilidad de los autores.

Chancusig MAisincho Omar Estaurdo

C.I: 050292371-7

Martinez Chiliquinga Luis Salomon

C.I.: 050266565-6

DEDICATORIA

Ha sido el omnipotente, quien ha permitido que la sabiduría dirija y guíe nuestros pasos. Ha sido el todopoderoso, quien ha iluminado nuestro sendero cuando más oscuro ha estado.

Ha sido el creador de todas las cosas, el que me ha dado fortaleza para continuar cuando a punto de caer hemos estado; por ello, con toda la humildad que de nuestro corazón puede emanar, dedicamos primeramente nuestro trabajo a Dios.

De igual forma, a nuestros padres, quienes han sabido formar con buenos sentimientos, hábitos y valores, lo cual nos ha ayudado a salir adelante buscando siempre el mejor camino.

AGRADECIMIENTO

Primeramente damos infinitamente gracias a Dios, por habernos dado fuerza y valor para terminar estos estudios.

A todos los maestros de la Universidad Técnica de Cotopaxi que nos asesoraron, porque cada uno, con sus valiosas aportaciones, nos ayudaron a crecer como personas y como profesionistas.

Un agradecimiento muy especial, a los Sistemas de Comunicación Latacunga, por habernos permitido realizar nuestro trabajo de tesis.

AVAL DEL DIRECTOR DE TESIS

Yo Ing. Patricio Navas en calidad de director de tesis, que tiene por título “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL Y BALANCEO DE CARGA, EN ROUTERS MIKROTIK CON CALIDAD DE SERVICIO (QoS), PARA LA RED WLAN (Wireless Local Area Network) DE LOS SISTEMAS DE COMUNICACIÓN LATACUNGA”, realizado por los postulantes Chancusig Maisincho Omar Estuardo con CI 0502923717 y Martínez Chilibuina Luis Salomón con CI 0502665656, de la carrera de Ingeniería en Informática y Sistemas Computacionales, considero que el presente proyecto de tesis cumple con todos los requerimientos metodológicos, aporte científico y objetivos planteados.

Latacunga, 28 de Junio de 2012

Atentamente:

Ing. Patricio Navas
DIRECTOR DE TESIS

CERTIFICACION DE LA EMPRESA

RESUMEN

El manejo del ancho de banda es una parte esencial del presente trabajo investigativo que proporciona Calidad de Servicio desde el momento en que el internet sale y llega íntegramente a su destino final, que son los usuarios.

Las herramientas utilizadas para la configuración del sistema PingBox-MK v1.0 fueron: Mikrotik RouterOS, Winbox, la plataforma de programación es NetBeans IDE 6.9.1, y para la comprobación del ancho de banda utilizaremos la herramienta Speed Test.

El propósito planteado al inicio del presente trabajo fue alcanzado en su totalidad permitiendo la estructuración y configuración de Router Mikrotik, para establecer el uso de balanceo de carga, a lo cual se diseñó el sistema visual de distribución, determinación, y asignación de conectividad en el uso de ancho de banda en la red WLAN de los sistemas de Comunicación Latacunga.

Mediante estadística descriptiva se realiza la representación en unagráfica de pastel, de la sumatoria obtenida por tabulación de todos los parámetros, se determinó que es de vital importancia la distribución óptima del ancho de banda sin pérdidas de paquetes en cada uno de los usuarios.

El presente diseño e implantación de fácil manejo para el control y balance de carga, permitió obtener calidad de servicio en la red Wlan de los Sistemas de Comunicación Latacunga, comprobando así la hipótesis planteada al inicio, siendo de esta manera beneficiados todos los usuarios de la red como la misma.

ABSTRACT

The management of bandwidth is an essential part of this research project which provides Quality of Service from the time when the internet comes and goes entirely to its final destination, that are the users.

The tools used for system configuration PingBox-MK v1.0 were: Mikrotik RouterOS, Winbox, the programming platform is NetBeans IDE 6.9.1, and for testing the bandwidth use Speed Test tool.

The purpose stated at the beginning of this work was achieved in full allowing the structuring and configuring MikroTik routers to establish the use of load balancing, which was designed to the visual system of distribution, determination, and allocation of connectivity in the use of bandwidth in the network WLAN of the Sistemas de Comunicación Latacunga.

Descriptive statistics were performed using the representation in a pie chart of the sum obtained by tabulating all the parameters, we determined that it is vital the optimal distribution of bandwidth without packet loss in each of the users. This design and implementation of user-friendly for the control and load balancing, quality of service possible to obtain network Wlan of the Sistemas de Comunicación Latacunga, thus proving the hypothesis at the outset, thus being beneficial to all users of the network as the same.

ÍNDICE GENERAL

PORTADA	
AUTORÍA	
AGRADECIMIENTO	
DEDICATORIA	
AVAL DEL DIRECTOR DE TESIS	
AVAL DEL PROFESIONAL DE INGLES	
CERTIFICACION DE LA EMPRESA	
RESUMEN	
ABSTRACT	
INTRODUCCION	1
CAPITULO I	4
1 MARCO TEÓRICO	4
1.1 Redes Inalámbricas	4
1.1.2 Introducción.	4
1.1.2 Clasificación de redes.	5
1.1.2.1 Redes Punto a Punto.	5
1.1.2.1.1 Topología de Anillo.	6
1.1.2.1.2 Topología en Estrella.	6
1.1.2.1.3 Topología de Árbol.	6
1.1.2.1.4 Topología en Malla.	6
1.1.2.2 Redes Multipunto	7
1.1.2.2.1 Topología de Bus	7
1.1.3 Redes según alcance	7
1.1.3.1 Redes Wlan (Wireless Local Area Network)	7
1.1.3.1.1 Ventajas de Wlans Sobre las Redes Fijas	8
1.1.3.1.2 Asignación de Canales	9
1.1.3.1.3 Seguridad	9
1.1.3.1.4 Velocidad	10
1.1.3.2 Red de área metropolitana (MAN)	10
1.1.3.2.1 Aplicaciones	11
1.1.3.2.2 MAN privada	11
1.1.3.2.3 MAN pública	11
1.1.3.2.4 Nodos de red	11

1.1.3.2.5	Extensión de red	12
1.1.3.2.6	Distancia entre nodos	12
1.1.3.2.7	Tráfico en tiempo real	12
1.1.3.2.8	Integración voz/datos/vídeo	12
1.1.3.2.9	Alta disponibilidad	12
1.1.3.2.10	Alta fiabilidad	13
1.1.3.2.11	Alta seguridad	13
1.1.3.3	Red de área amplia (WAN)	13
1.1.3.3.1	Componentes	14
1.1.4	Redes Infrarrojas	16
1.1.5	El uso del Espacio, del Tiempo y del Espectro en redes de radio frecuencia.	16
1.1.6	Factor de Distancia.	17
1.1.7	Puntos de Acceso	18
1.1.7.1	Aislamiento en Sistemas Vecinos.	18
1.1.7.2	Modulación de Radio.	19
1.1.8	Eficiencia del Tiempo	20
1.1.9	Límite de la longitud del Paquete y su Tiempo.	21
1.2	RouterOS Mikrotik	22
1.2.1	Características de los Routers	22
1.2.2	Estructura del RouterOS	22
1.2.3	Router Board Mikrotik	23
1.2.4	Tarjetas mini PCI inalámbricas	23
1.2.4.1	Detalles técnicos de Mini PCI	23
1.2.5	Sistemas Integrados	24
1.2.6	Accesorios	24
1.3	Interfaces	25
1.3.1	Hardware Físico	25
1.3.2	Configuración de interfaces de red	26
1.3.2.1	Interfaz Privada	26
1.3.2.2	Interfaz pública	26
1.3.2.3	Interfaz de marcado a petición	26
1.3.3	Clasificación de Routers	27
1.3.3.1	Según el Alcance	27
1.3.3.2	Según la Conectividad	27
1.3.4	Arquitectura de Routers	27
1.3.5	Comparativa entre Routers y Bridges	28
1.3.5.1	Ventajas de Bridges sobre Routers:	28
1.3.5.2	Ventajas de Routers sobre Bridges	28

1.4	Balanceo de Carga	29
1.4.1	Introducción	29
1.4.2	Definición	29
1.4.3	Balanceador de carga en RouterOS Mikrotik	30
1.4.4	Importancia	30
1.5	Calidad de Servicio (QoS)	30
1.5.1	Historia de QoS	30
1.5.2	Definición de Calidad de Servicio (Qos)	30
1.5.2.1	QoS, CoS y ToS	31
1.5.2.1.1	QoS (calidad de servicio)	31
1.5.2.1.2	CoS (clase de servicio)	32
1.5.2.1.3	ToS (tipo de servicio)	32
1.5.3	Clasificación de QoS.	33
1.5.3.1	Según la sensibilidad del tráfico	33
1.5.3.1.1	QoS muy sensible al retardo.	33
1.5.3.1.2	QoS algo sensible al retardo.	33
1.5.3.1.3	QoS muy sensible a pérdidas.	34
1.5.3.1.4	QoS nada sensible.	34
1.5.3.2	Según quién solicite el nivel de calidad de servicio	34
1.5.3.2.1	QoS Implícita	34
1.5.3.2.2	QoS Explícita	35
1.5.3.3	Según las garantías	35
1.5.3.3.1	QoS garantizada / HardQoS	35
1.5.3.3.2	QoS no garantizada / Lack of QoS	36
1.5.3.3.3	QoS servicios diferenciados/ Soft QoS	36
1.5.3.4	Según el lugar de aplicación	36
1.5.3.4.1	QoS extremo a extremo (end-to-end)	36
1.5.3.4.2	QoS borde a borde (edge-to-edge)	37
1.5.4	Parámetros de QoS	37
1.5.4.1	Tráfico de Red	37
1.5.4.2	Retardo	38
1.5.4.3	Latencia	38
1.5.4.4	JITTER (inestabilidad o variabilidad en el retardo)	38
1.5.4.5	Ancho de Banda	39
1.5.4.6	Pérdida de Paquetes	39
1.5.4.7	Disponibilidad	39
1.5.4.8	Rendimiento	40
1.5.4.9	Priorización	40
1.5.4.10	Encolado	40

1.5.4.11	Planificación	40
1.5.4.12	Flujo	41
1.5.4.13	Acuerdo de Nivel de Servicio o Service Level Agreement (SLA)	41
1.5.4.14	Los SLA	41
1.5.4.15	Especificaciones del condicionamiento del tráfico	42
1.5.5	QoS Basic Framework	
1.5.6.	Algoritmos para la obtención de QoS	42
1.5.6.1	Algoritmos de mejor esfuerzo (Best Effort)	43
1.5.6.2	Algoritmos deterministas	43
1.5.6.3	Algoritmos intermedios	44
1.5.7	Beneficios al aplicar QoS	44
1.5.7.1	Ventajas para las aplicaciones	44
1.5.7.2	Beneficios para las empresas	45
1.5.7.3	Beneficios para los proveedores de servicio	45
1.5.8	Gestión del ancho de banda vs QoS	45
1.5.9	QoS Ofrecida por algunos Sistemas Operativos	46
1.5.9.1	Calidad de servicio en Windows 2000	46
1.5.9.1.1	API GqoS (Generic Quality of Service).	47
1.5.9.1.2	QoS Service Provider (SP)	47
1.5.9.1.3	Infraestructura de control de tráfico con soporte de DiffServ y 802.1p	47
1.5.9.2	Calidad de servicio en IOS de Cisco	47
1.6	NetBeans	49
1.7	Telnet (TELEcommunication NETwork)	49
CAPITULO II		
2	ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS	51
2.1	Análisis de la Encuesta	51
2.2	Entrevista	62
CAPITULO III		
3	APLICACIÓN O VALIDACIÓN DE LA PROPUESTA	65
3.1	Objetivos	66
3.1.1	Objetivo General	66
3.1.2	Objetivos Específicos	66
3.2	Justificación	66
3.3	Diseño de la propuesta	68
3.3.1	Datos informativos de la Empresa	68

3.3.2	Metodología	68
3.3.3	Análisis y Diseño de la aplicación Informática	68
3.3.3.1	Fase de planificación	68
3.3.3.2	Fase de diseño	69
3.3.3.2.1	Manual del Programador	69
3.3.3.2.1.1	Comandos básicos de RouterOS Mikrotik	69
3.3.3.2.1.2	Acceso a Mikrotik mediante acceso remoto.	70
3.3.3.2.1.3	Programación interfaz gráfica de acceso	73
3.3.3.2.1.4	Librerías utilizadas	74
3.3.3.2.1.5	Código Fuente Conexión RouterOS	92
3.3.3.2.1.5.1	Librería Commons-Net	92
3.3.3.2.1.5.2	Acceso de logueo a router mediante puerto telnet	95
3.3.3.2.1.5.3	Librerías fundamentales utilizadas en este proyecto	96
3.3.3.2.1.5.4	Clasificación y separación de datos	105
3.3.3.2.1.5.5	Interface de modificación de usuarios	113
3.3.3.3	Fase de instalación	124
3.3.3.3.1	Instalación y configuración de sistema PingBox-MK v1.0	124
3.3.3.3.2.1	Requerimientos de hardware y software	124
3.3.3.3.2.1.1	Requisitos y recomendaciones de hardware	124
3.3.3.3.2.1.2	Requisitos y recomendaciones de software	125
3.3.3.3.3.2	Proceso de Instalación	125
3.3.3.3.3.3	Arranque de CD-ROM	125
3.3.3.3.3.4	Instalación Java (JM) 6	125
3.3.3.3.3.5	Instalación PingBox-MK v1.0	129
3.3.3.3.3.5.1	Manual de uso de aplicación PingBox-MK v1.0	131
3.3.3.3.3.5.2	Utilización del sistema ping PingBox-MK v1.0	131
3.3.3.3.3.5.3	Acceso de usuario	131
3.3.3.3.3.5.4	Error de acceso	133
3.3.3.3.3.5.5	Conexión exitosa	133
3.3.3.3.3.5.6	Control de ancho de banda y calidad de servicio	134
3.3.3.3.3.5.7	Limitación de colas simple (Queue Simple)	134
3.3.3.3.3.5.8	Selección de balanceo de carga	135
3.3.3.3.3.5.9	Comprobación del cambio de Limitación	136
3.3.4	Fase de prueba	138
3.3.5	Fase de funcionamiento	138
	CONCLUSIONES	139
	RECOMENDACIONES	140

BIBLIOGRAFÍA	141	
BIBLIOGRAFÍA CONSULTADA	141	
BIBLIOGRAFÍA VIRTUAL	141	
ANEXOS	144	
Gráfico III.1	Conexión remota hacia Mikrotik	70
Gráfico III.2	Acceso a RouterOS	71
Gráfico III.3	Consola del RouterOSMikrotik	71
Gráfico III.4	Acceso directorio colas simples	71
Gráfico III.5	Acceso directorio colas simples 2	72
Gráfico III.6	Limitación tráfico de descarga	72
Gráfico III.7	Impresión reglas balanceo de carga	73
Gráfico III.8	Acceso a loguin	74
Gráfico III.9	Librerías	75
Gráfico III.10	Creación Clase Pública	77
Gráfico III.11	Ingreso textbox y asignación de parámetros	93
Gráfico III.12	Carga del proyecto	94
Gráfico III.13	Incorporación del archivo	95
Gráfico III.14	Añadidura de librerías.	96
Gráfico III. 15	Importación de clase	99
Gráfico III.16	Creación de clase	104
Gráfico III.17	Extracción de la información interna	106
Gráfico III.18	Clasificación de datos.	111
Gráfico III.19	Cambio y reingreso de datos	113
Gráfico III.20	Actualización de datos	114
Gráfico III.21	Datos de las limitaciones	115
Gráfico III.22	Asignaciones	124
Gráfico III.23	Versión java en Windows	127
Gráfico III.24	Impresión de Java no instalado	127
Gráfico III.25	Selección y ejecución de archivo jre	128
Gráfico III.26	Aceptación de la licencia	128
Gráfico III.27	Instalación de Java	129
Gráfico III.27	Finalización de la Instalación	130
Gráfico III.28	Selección y ejecución PingBox v1.1	130
Gráfico III.29	Acuerdo de licencia	131
Gráfico III.30	Instalación del sistema	132
Gráfico III.31	Acceso al sistema	133
Gráfico III.32	Error de acceso	134
Gráfico III.32	Conexión exitosa	135

Gráfico III.33	Datos de usuarios	135
Gráfico III.34	Aplicación de cambios	136
Gráfico III.35	Actualización de ancho de banda	136
Gráfico III.36	Limitación de ancho de banda	137
Gráfico III.37	Mensaje de modificación exitosa	137
Gráfico III.38	Ingreso a Speed Test	138
Gráfico III.39	Fase de comprobación de velocidad	139
Tabla III.1	Requisitos y recomendaciones de hardware	125
Tabla III.2	Requisitos y recomendaciones de software	126
Anexo 1	Encuesta a los usuarios	145
Anexo 2	Entrevista	147

INTRODUCCIÓN

En la actualidad las redes informáticas, se han vuelto indispensables, tanto para las personas como organizaciones, les da oportunidad de interactuar con el resto del mundo, ya sea por motivos comerciales, personales o emergencias.

Un Sistema de Balanceo de Carga nos permite distribuir equitativamente nuestro internet, por esta razón el motivo principal de la investigación es colaborar con los Sistemas de Comunicación Latacunga para aplicar esta herramienta importante y fundamental en cada uno de los usuarios y llegar a cada uno de ellos con calidad de servicio.

El presente trabajo investigativo se lo realiza en los Sistemas de Comunicación Latacunga, gracias a la gentil colaboración de los entes principales de esta importante institución, durante todo el periodo de la investigación, partiendo desde la presentación de la propuesta hasta la culminación de la misma, ya que su mayor debilidad era no tener un correcto balanceo de carga mediante QoS para los usuarios.

La optimización en el uso de los sistemas informáticos es uno de los elementos de interacción y desarrollo que rige los destinos de la ciencia informática, permitiendo que las redes resulten ser uno de los elementos tecnológicos más importantes al momento de definir un sistema informático en una organización. Con la implementación de calidad de servicio (QoS), es posible ofrecer mayor garantía y seguridad para las aplicaciones avanzadas, cuando el tráfico de estas aplicaciones pasa a tener prioridad en relación con aplicaciones tradicionales.

En relación al objetivo general que es diseñar e implementar un sistema de control, balanceo de carga en Routers Mikrotik con calidad de servicio (QOS), para la red wlan de los sistemas de comunicación Latacunga con los específicos que es analizar la estructuración y configuración de Routers Mikrotik, empleados y diseñados para redes inalámbricas para establecer el uso de balanceo de carga y diseñar un software

o sistema visual de fácil manejo y eficaz en la distribución, determinación, y asignación de conectividad en el uso de ancho de banda aplicado para Routers Mikrotik, fueron fundamentales y alcanzables en su totalidad ya que todo fue enmarcado en base al balanceo de carga.

Siendo la hipótesis el diseño e implantación de un sistema de fácil manejo para el control y balance de carga, permite obtener Calidad de Servicio en la red Wlan de los Sistemas de Comunicación Latacunga, fue comprobada y alcanzada ya que se logró implantar la presente investigación obteniendo Calidad de Servicio en la presente institución.

El presente trabajo tiene como tipo de investigación bibliográfica y de campo, para recolectar información se utilizó la técnica de la encuesta y la entrevista, que nos permitieron obtener información, veraz y sustancial.

En el capítulo I Marco Teórico, se realiza el estudio de los conceptos necesarios para entender el funcionamiento y la implementación del Sistema de Control y Balanceo de Carga en Routers Mikrotik con Calidad de Servicio (QoS), para la Red Wlan (wireless local area network) de los Sistemas de Comunicación Latacunga”, que permitan la configuración y simulación de un router e incorporen Calidad de Servicio, a lo cual se describe aquellas herramientas que se consideraron apropiadas para la implementación de la misma.

En el capítulo II Análisis de Resultados, se examinan los resultados obtenidos en la encuesta realizada a los usuarios y al gerente de los Sistemas de Comunicación Latacunga, que incluyen las cabeceras de los paquetes generados en los escenarios de simulación, obteniendo resultados, que nos permitió realizar el presente proyecto.

En el capítulo III Aplicación o Validación de la propuesta, nos permite desarrollar nuestro valioso aporte a la empresa de una manera metodológica siguiendo procesos sustanciales, como son la fase de planificación, diseño, instalación, prueba y funcionamiento, todas estas fases nos permitió desarrollar el sistema Pinbox.

Y finalmente se culmina con las conclusiones y recomendaciones respectivas, en base al presente trabajo investigativo.

CAPITULO I

1 MARCO TEÓRICO

Diseño e implantación de un sistema de control y balanceo de carga, en Routers Mikrotik con calidad de servicio (QoS), para la red Wlan (wireless local area network) de los “Sistemas De Comunicación Latacunga”.

1.1 *Redes Inalámbricas*

1.1.2 *Introducción.*

El crecimiento tecnológico a un nivel popular, es cada día más elevado con una velocidad inmensa, empresas fabricantes de redes inalámbricas comenzaron a producir aparatos con esta tecnología, los poseedores de estos dotes más usados son los celulares, que permiten conectarse a redes caseras Wireless para compartir sus recursos y tomar de la red otros disponibles, estas se basan en un enlace que utiliza ondas electromagnéticas (radio e infrarrojo) en lugar de cableado estándar, muchas

tecnologías se diferencian por la frecuencia de transmisión que utilizan, el alcance y la velocidad de sus transmisiones.

La traducción al español de Network Wireless, es Redes Inalámbricas, simplemente un cambio de idioma, por tal motivo no existe diferencia, estas redes son tecnologías más prometedoras y discutidas en esta década, y uno de los fines más delimitados es el de poder comunicar computadoras mediante tecnología inalámbrica, con esto, las conexiones de Redes Inalámbricas facilitan la operación en lugares donde la computadora no permanece en un solo lugar, siendo de uso más general las portátiles, como también otros equipos móviles, permitiendo que los dispositivos remotos se conecten sin dificultad, ya se encuentren a unos metros de distancia como a varios kilómetros.

La instalación de estas redes no requiere de ningún cambio significativo en la infraestructura existente como pasa con las redes cableadas, como tampoco hay necesidad de agujerear las paredes para pasar cables ni de instalar portacables o conectores, lo que ha permitido el uso de esta tecnología se extienda con rapidez.

1.1.2 Clasificación de redes.

Desde el punto de vista de la organización de los nodos, se clasifica según la forma en que los nodos de una red utilizan uno o más medios de transmisión, y la forma en que quedan distribuidos los nodos que forma la red.

1.1.2.1 Redes Punto a Punto.

Una red punto a punto es aquella que siempre dos terminales están unidas por una línea o cable, la principal característica es que no es compartido derivando a un enlace puro, tal que su uso es dedicado sólo a esas dos terminales.

Las topologías que soporta esta clasificación son:

1.1.2.1.1 Topología de Anillo.

Esta topología conecta a cualquier terminal, únicamente con sus dos destinos más próximos mediante una línea dedicada, de tal forma que la última de las terminales se conecta con la primera de ellas por uno de los extremos, formando así un ciclo o un anillo a través del cual fluye la información cuando las terminales se comunican, esta es unidireccional o simplex, y viaja de terminal a terminal hasta que encuentra su destino y regresa a su origen, tiene la desventaja de que cualquier fallo entre alguna de las líneas dedicadas genera una falla letal en la red.

1.1.2.1.2 Topología en Estrella.

Ésta topología conecta a todas las terminales entre sí, aunque no en forma directa, para ello utiliza un elemento que organiza el flujo de la información en la red mediante switches que conectan a la terminal destino con la terminal origen, siendo a la vez más robusta que la topología de anillo, ya que si falla una terminal, el resto sigue funcionando, esto surge a una desventaja es que si falla el concentrador entonces irremediablemente fallará toda la red.

1.1.2.1.3 Topología de Árbol.

Es derivada de la topología de estrella, en ésta lo que se tiene, son diferentes estrellas conectadas entre sí, utilizando concentradores como elemento de interconexión, algunas de estas estrellas tienen más prioridad que otras y así es posible encausar la información a través de diferentes estrellas.

1.1.2.1.4 Topología en Malla.

Esta topología busca tener conexión física entre todas las terminales de la red, utilizando conexiones punto a punto, esto permitirá que cualquier terminal se comunique con otras terminales de forma paralela si fuera necesario, la principal ventaja es que este tipo de redes difícilmente falla, pues inclusive, si alguna de estas líneas fallara aún, se podrían encontrar otras rutas para lograr la información, pero la

desventaja de esta topología, es que se requiere demasiado cableado específicamente si existen n terminales en la red.

1.1.2.2 Redes Multipunto

En esta red sólo existe una línea de comunicación cuyo uso está compartido por todas las terminales en la red, la información fluye de forma bidireccional y es discernible para todas las terminales de la red, lo típico es que en una conexión multipunto las terminales compiten por el uso del medio de forma que el primero que lo encuentra disponible lo acapara, aunque también puede negociar su uso.

1.1.2.2.1 Topología de Bus

Los buses lineales son quizás la topología más utilizadas para redes de área local, son las más baratas y una de las más conflictivas, que consiste en conectar todas las terminales a una línea común, utilizando para ello un dispositivo llamado Tap, además de un segundo cable auxiliar (drop line) que conecta a la terminal al Tap y éste a su vez a la línea compartida, también en los extremos del bus se requieren dos elementos terminadores, pero las desventajas en esta topología es la longitud del cable, terminales, el no uso de Taps, por otra parte los mensajes se desgastan cada vez que pasan por un Tap, y si no tubiese terminadores los mensajes se colapsarían y se perderán.

1.1.3 *Redes según alcance*

1.1.3.1 Redes Wlan (Wireless Local Area Network)

En los últimos años las redes WLAN, (Wireless Local Area Network) están ganando mucha popularidad, que se ve acrecentada conforme sus prestaciones aumentan y se descubren nuevas aplicaciones para ellas.

Las WLAN permiten a sus usuarios acceder a información y recursos en tiempo real sin necesidad de estar físicamente conectados a un determinado lugar, es decir móvil

y elimina la necesidad de usar cables, estableciendo nuevas aplicaciones, añadiendo flexibilidad a la red, y lo más importante incrementa la productividad como eficiencia en las empresas donde está instalada.

Puntuando que un usuario dentro de una red WLAN puede transmitir y recibir voz, datos y vídeo dentro de edificios, entre edificios o campus universitarios e inclusive sobre áreas metropolitanas a velocidades de 11 Mbit/s, o superiores y no solamente se encuentra esta aplicación a las empresas, sino que su extensión a ambientes públicos como medio de acceso a Internet o para cubrir zonas de alta densidad de usuarios (hot spots).

Las nuevas posibilidades que ofrecen las WLANs son: permitir una fácil incorporación de nuevos usuarios a la red, ofrecen una alternativa de bajo costo a los sistemas cableados, además de la posibilidad para acceder a cualquier base de datos o cualquier aplicación localizada dentro de la red.

1.1.3.1.1 Ventajas de Wlans Sobre las Redes Fijas

Movilidad, es decir el acceso a la información en tiempo real, en cualquier lugar dentro de la organización o el entorno público (zona limitada) en el que está desplegado, otro aspecto importante es la simplicidad y rapidez en la instalación de una WLAN, ya que es rápida y fácil, elimina la necesidad de tirar cables a través de paredes y techos. La tecnología inalámbrica permite a la red llegar a puntos de difícil acceso para una LAN cableada.

Costo de propiedad reducido con la inversión inicial requerida para una red inalámbrica, puede ser más alta que el costo en hardware de una LAN, pero la inversión de toda la instalación y el costo durante el ciclo de vida puede ser significativamente inferior. Los beneficios a largo plazo son superiores en ambientes dinámicos que requieren acciones y movimientos frecuentes.

Escalabilidad: los sistemas de WLAN pueden ser configurados en una variedad de topologías para satisfacer las necesidades de las instalaciones y aplicaciones específicas. Las configuraciones son muy fáciles de cambiar y además resulta muy fácil la incorporación de nuevos usuarios a la red.

1.1.3.1.2 Asignación de Canales

Los estándares 802.11b y 802.11g utilizan la banda de 2.4 – 2.5 Ghz. En esta se definen 11 canales utilizables por equipos WIFI, los cuales pueden configurarse de acuerdo a necesidades particulares. Sin embargo, los 11 canales no son completamente independientes y en la práctica sólo se pueden utilizar 3 canales en forma simultánea (1, 6 y 11). Esto es correcto para USA y muchos países de América Latina, pues en Europa, el ETSI ha definido 13 canales. En este caso, por ejemplo en España, se pueden utilizar 4 canales no-adyacentes (1, 5, 9 y 13). Esta asignación de canales usualmente se hace sólo en el punto de acceso, pues los “clientes” automáticamente detectan el canal, salvo en los casos en que se forma una red ad hoc o punto a punto cuando no existe punto de acceso.

1.1.3.1.3 Seguridad

Uno de los problemas de este tipo de redes es precisamente la seguridad ya que cualquier persona con una terminal inalámbrica podría comunicarse con un punto de acceso privado si no se disponen de las medidas de seguridad adecuadas. Dichas medidas van encaminadas en dos sentidos: por una parte está el cifrado de los datos que se transmiten y en otro plano, pero igualmente importante, se considera la autenticación entre los diversos usuarios de la red. En el caso del cifrado se están realizando diversas investigaciones ya que los sistemas considerados inicialmente se han conseguido descifrar. Para la autenticación se ha tomado como base el protocolo de verificación EAP (Extensible Authentication Protocol), que es bastante flexible y permite el uso de diferentes algoritmos.

1.1.3.1.4 Velocidad

Otro de los problemas que presenta este tipo de redes es que actualmente no alcanzan la velocidad que obtienen las redes de datos cableadas, así el tener que cifrar toda la información supone que gran parte se transmite sea de control y no información útil para los usuarios, por lo que incluso se reduce la velocidad de transmisión de datos útiles.

(<http://www.maestrosdelweb.com/editorial/redeswlan/>)

1.1.3.2 Red de área metropolitana (MAN)

Una red de área metropolitana MAN(*metropolitan area network*) es una red de alta velocidad que da cobertura en un área geográfica extensa, proporciona capacidad de integración de múltiples servicios mediante la transmisión de datos, voz y vídeo, sobre medios de transmisión tales como fibra óptica y par trenzado (MAN BUCLE), la tecnología de pares de cobre se posiciona como la red más grande del mundo una excelente alternativa para la creación de redes metropolitanas, por su baja latencia (entre 1 y 50ms), gran estabilidad y la carencia de interferencias radioeléctricas, las redes MAN BUCLE, ofrecen velocidades de 10Mbps, 20Mbps, 45Mbps, 75Mbps, sobre pares de cobre y 100Mbps, 1Gbps y 10Gbps mediante Fibra Óptica, siendo una versión más grande que la LAN y que normalmente se basa en una tecnología similar a esta, la principal razón para distinguir una MAN con una categoría especial es que se ha adoptado un estándar para que funcione, que equivale a la norma IEEE, las cuales comprenden una ubicación geográfica determinada "ciudad, municipio", y su distancia de cobertura es mayor de 4 km .

Son redes con dos buses unidireccionales, cada uno de ellos es independiente del otro en cuanto a la transferencia de datos.

1.1.3.2.1 Aplicaciones

Las redes de área metropolitana tienen muchas y variadas aplicaciones, las principales son:

- Interconexión de redes de área local (LAN)
- Interconexión ordenador a ordenador
- Sistemas de Video vigilancia Municipal.
- Transmisión CAD/CAM
- Pasarelas para redes de área extensa (WAN).

(<http://multingles.net/docs/Manual%20-Z20Redes%20WiFi%20inalambricas.pdf>)

1.1.3.2.2 MAN privada

Una MAN privada sería un gran departamento o administración con edificios distribuidos por la ciudad, transportando todo el tráfico de voz, datos y videos entre edificios por medio de su propia MAN y encaminando la información externa por medio de los operadores públicos los cuales podrían ser transportados entre los diferentes edificios, bien en forma de paquetes o sobre canales de ancho de banda fijos.

1.1.3.2.3 MAN pública

Una MAN pública es la infraestructura que un operador de telecomunicaciones instala en una ciudad con el fin de ofrecer servicios de banda ancha a sus clientes localizados en esta área geográfica.

1.1.3.2.4 Nodos de red

Las redes de área ciudadana permiten ejecuciones superando los 600 nodos de acceso a la red, por lo que se hace muy eficaz para entornos públicos y privados con un gran número de puestos de trabajo.

1.1.3.2.5 Extensión de red

Las redes de área metropolitana permiten alcanzar un diámetro en torno a los 500 km, dependiendo el alcance entre nodos de red del tipo de cable utilizado, así como de la tecnología empleada, este diámetro se considera suficiente para abarcar un área metropolitana. Abarcan una ciudad y se pueden conectar muchas entre sí, formando más redes.

1.1.3.2.6 Distancia entre nodos

Las redes de área metropolitana permiten distancias entre nodos de acceso de varios kilómetros, dependiendo del tipo de cable. Estas distancias se consideran suficientes para conectar diferentes edificios en un área metropolitana o campus privado.

1.1.3.2.7 Tráfico en tiempo real

Las redes de área metropolitana garantizan unos tiempos de acceso a la red mínimos, lo cual permite la inclusión de servicios síncronos necesarios para aplicaciones en tiempo real, donde es importante que ciertos mensajes atraviesen la red sin retraso incluso cuando la carga de red es elevada, para lo cual entre nodo y nodo se puede tener en aproximación límite unos 20 km de cable, pero no se sabe en qué momento se puede perder la información o los datos mandados.

1.1.3.2.8 Integración voz/datos/vídeo

Los servicios síncronos requieren una reserva de ancho de banda; tal es el caso del tráfico de voz y vídeo, por este motivo las redes de área metropolitana son redes óptimas para entornos de tráfico multimedia.

1.1.3.2.9 Alta disponibilidad

Referida al porcentaje de tiempo en el cual la red trabaja sin fallos, las redes de área metropolitana tienen mecanismos automáticos de recuperación frente a fallos, en el

caso del cable de cobre se utiliza el bonding EFM, permitiendo la agregación de caudal en múltiples cables, el bonding EFM permite a la red recuperar la operación normal, ante la rotura de uno de los cables, cualquier fallo en un nodo de acceso o cable es detectado rápidamente y aislado. Las redes MAN son apropiadas para entornos como control de tráfico aéreo, aprovisionamiento de almacenes, bancos y otras aplicaciones comerciales donde la indisponibilidad de la red tiene graves consecuencias.

1.1.3.2.10 Alta fiabilidad

Fiabilidad referida a la tasa de error de la red mientras se encuentra en operación, se entiende por tasa de error el número de bits erróneos que se transmiten por la red, en general la tasa de error para fibra óptica es menor que la del cable de cobre a igualdad de longitud, la tasa de error no detectada por los mecanismos de detección de errores es del orden de 10⁻²⁰, esta característica permite a la redes de área metropolitana trabajar en entornos donde los errores pueden resultar desastrosos como es el caso del control de tráfico aéreo.

1.1.3.2.11 Alta seguridad

La fibra óptica y el cable, son un medio seguro, porque no es posible leer o cambiar la señal sin interrumpir físicamente el enlace, la rotura de un cable y la inserción de mecanismos ajenos a la red implican una caída del enlace de forma temporal, además se requiere acceso y actuación sobre el cable físico, aunque este tipo de actuaciones pasen fácilmente desapercibidas.

(<http://multingles.net/docs/Manual%20-%20Redes%20WiFi%20inalambricas.pdf>)

1.1.3.3 Red de área amplia (WAN)

Son redes informáticas que se extienden sobre un área geográfica extensa, que contiene una colección de máquinas dedicadas a ejecutar los programas de usuarios

(hosts), estos están conectados por la red que lleva los mensajes de un host a otro, estas LAN de host acceden a la subred de la WAN por un encaminador.

La subred tiene varios elementos:

- Líneas de comunicación: mueven bits de una máquina a otra.
- Elementos de conmutación: Máquinas especializadas que conectan dos o más líneas de transmisión, que se suelen llamar encaminadores o routers.

Cada host está después conectado a una LAN en la cual está el encaminador que se encarga de enviar la información por la subred, por lo tanto una WAN contiene numerosos cables conectados a un par de encaminadores, si dos de estos no comparten cables desean comunicarse, han de hacerlo a través de routers intermedios. El paquete se recibe completo en cada uno de los intermedios y se almacena allí hasta que la línea de salida requerida esté libre.

Se pueden establecer WAN en sistemas de satélite o de radio en tierra en los que cada encaminador tiene una antena con la cual poder enviar y recibir la información, por su naturaleza, las redes de satélite serán de difusión, lo que hoy en día Internet proporciona WAN de alta velocidad, que normalmente estas son una red punto a punto, es decir, red de paquete conmutado.

1.1.3.3.1 Componentes

Servidor: El servidor es aquel o aquellos ordenadores que van a compartir sus recursos hardware y software con los demás equipos de la red. Sus características son potencia de cálculo, importancia de la información que almacena y conexión con recursos que se desean compartir.

Estación de trabajo: Los ordenadores que toman el papel de estaciones de trabajo aprovechan o tienen a su disposición los recursos que ofrece la red así como los servicios que proporcionan los Servidores a los cuales pueden acceder.

Gateways o pasarelas: Es un hardware y software que permite las comunicaciones entre la red local y grandes ordenadores (mainframes), el Gateway adapta los protocolos de comunicación del mainframe a los de la red, y viceversa.

Bridges o puentes: Es un hardware y software que permite que se conecten dos redes locales entre sí, un puente interno es el que se instala en un servidor de la red, y un puente externo es el que se hace sobre una estación de trabajo de la misma red, los puentes también pueden ser locales o remotos, los puentes locales son los que conectan a redes de un mismo edificio, usando tanto conexiones internas como externas y los puentes remotos conectan redes distintas entre sí, llevando a cabo la conexión a través de redes públicas, como la red telefónica, RDSI o red de conmutación de paquetes.

Tarjeta de red: También se denominan NIC (Network Interface Card), básicamente realiza la función de intermediario entre el ordenador y la red de comunicación, en ella se encuentran grabados los protocolos de comunicación de la red. La comunicación con el ordenador se realiza normalmente a través de las ranuras de expansión que éste dispone, ya sea ISA, PCI o PCMCIA, aunque algunos equipos disponen de este adaptador integrado directamente en la placa base.

El medio: Constituido por el cableado y los conectores que enlazan los componentes de la red, los medios físicos más utilizados son el cable de par trenzado, par de cable, cable coaxial y la fibra óptica.

Concentradores de cableado: Una LAN en bus usa solamente tarjetas de red en las estaciones y cableado coaxial para interconectarlas, además de los conectores, sin embargo este método complica el mantenimiento de la red ya que si falla alguna conexión toda la red deja de funcionar. Para impedir estos problemas las redes de área local usan concentradores de cableado para realizar las conexiones de las estaciones, en vez de distribuir las conexiones el concentrador las centraliza en un único

dispositivo manteniendo indicadores luminosos de su estado e impidiendo que una de ellas pueda hacer fallar toda la red. (<http://es.wikipedia.org/wiki/WAN>)

1.1.4 *Redes Infrarrojas*

La transmisión Infrarroja es actualmente una alternativa para las Redes Inalámbricas, el principio de la comunicación de datos es una tecnología que se ha estudiado desde los 70's, Hewlett-Packard desarrolló su calculadora HP-41 que utilizaba un transmisor infrarrojo para enviar la información a una impresora térmica portátil, actualmente esta tecnología es la que utilizan los controles remotos de las televisiones o aparatos eléctricos que se usan en el hogar, con el mismo principio se usa para la comunicación de Redes, se utiliza un transreceptor que envía un haz de Luz Infrarroja, hacia otro que la recibe, la transmisión de luz se codifica y decodifica en el envío y recepción en un protocolo de red existente, uno de los pioneros en esta área es Richard Allen, que fundó Photonics Corp., en 1985 y desarrolló un transreceptor infrarrojo, los primeros dirigían el haz infrarrojo de luz a una superficie pasiva, generalmente el techo, donde otro transreceptor recibía la señal. (http://www.forpas.us.es/aula/hardware/dia4_redes.pdf)

Estas redes, son de luz infrarroja que están limitadas por el espacio y casi generalmente la utilizan en las estaciones que se encuentran en un solo cuarto o piso, algunas compañías que tienen sus oficinas en varios edificios realizan la comunicación colocando los receptores/emisores en las ventanas de los edificios, las transmisiones de radio frecuencia tienen una desventaja, que los países están tratando de ponerse de acuerdo en cuanto a las bandas que cada uno puede utilizar, al momento de realizar este trabajo ya se han reunido varios países para tratar de organizarse en cuanto a que frecuencias pueden utilizar cada uno.

1.1.5 *El uso del Espacio, del Tiempo y del Espectro en redes de radio frecuencia.*

El método de acceso, como la modulación de radio y el ancho de banda disponible, es importante para determinar la eficiencia y la capacidad de un sistema de radio, los

factores que permiten optimizar la capacidad de comunicación dentro de una área geográfica y del espectro de ancho de banda, son considerados más importantes que la forma como son implementadas, los diseños de alta eficiencia han sido evitados en sistemas de radio y redes porque su utilización no es muy obvia en cuanto a rapidez y conveniencia, uno de los aspectos más importantes de la eficiencia del tiempo es la asignación de frecuencia consolidada y el tráfico de cargas de usuarios no relacionados entre sí, de por lo menos, el punto alto y el promedio de circulación de cada grupo deben de tener diferentes patrones.

Independientemente del rango, un conjunto de enlaces puede únicamente dar servicio a una fracción del área total, para una cobertura total del área, se debe de usar canales independientes, derivados por frecuencia, código o tiempo, no es fácil minimizar el número de canales independientes o conjunto de enlaces para una cobertura total, mientras la distancia incrementa, se origina que la señal de radio disminuya, debido a la curvatura de la tierra o a obstáculos físicos naturales existentes, este diseño es muy utilizado en interferencia limitada, existe una trayectoria normal cuando en el nivel de transferencia, de estaciones simultáneamente activas, no prevén la transferencia actual de datos, para este tipo de diseño, los siguientes factores son importantes:

- Es necesaria una relación señal-interferencia, para una comunicación correcta.
- Se requiere de un margen expresado en estadísticas para generar esta relación, aún en niveles de señal variables
- La posición de las antenas que realizan la transmisión. La cual puede ser limitada por las estaciones y perfectamente controlada por puntos de acceso fijos.

1.1.6 *Factor de Distancia.*

El promedio de inclinación de curva es reconocido por tener un exponente correspondiente a 35-40 dB/Decena para una extensión lejana y de propagación no óptica. Para distancias cortas el exponente es más cerca al espacio libre o 20 dB/Decena, el aislamiento de estaciones simultáneamente activas con antenas

omnidireccionales puede requerir factores de Reuso de 49 o más en espacio libre, la distancia de aislamiento trabaja muy bien con altos porcentajes de atenuación media, dependiendo de lo disperso del ambiente, la distancia de aislamiento en sistemas pequeños resulta ser en algunos casos la interferencia inesperada y por lo tanto una menor cobertura.

1.1.7 *Puntos de Acceso*

La infraestructura de un punto de acceso es simple: “Guardar y Repetir”, son dispositivos que validan y retransmiten los mensajes recibidos, estos dispositivos pueden colocarse en un punto en el cual puedan abarcar toda el área donde se encuentren las estaciones. Las características a considerar son:

- La antena del repetidor debe de estar a la altura del techo, esto producirá una mejor cobertura que si la antena estuviera a la altura de la mesa.
- La antena receptora debe de ser más compleja que la repetidora, así aunque la señal de la transmisión sea baja, ésta podrá ser recibida correctamente.

Un punto de acceso compartido es un repetidor, al cual se le agrega la capacidad de seleccionar diferentes puntos de acceso para la retransmisión.

1.1.7.1 Aislamiento en Sistemas Vecinos.

Con un proyecto basado en puntos de acceso, la cobertura de cada punto de acceso es definible y puede ser instalado para que las paredes sean una ayuda en lugar de un obstáculo, las estaciones están recibiendo o transmitiendo activamente muy poco tiempo y una fracción de las estaciones asociadas, con un punto de acceso, están al final de una área de servicio; entonces el potencial de interferencia entre estaciones es mínimo comparado con las fallas en otros mecanismos de transmisión de gran escala, podemos definir que tendremos dos beneficios del punto de acceso, lo cual estos detalles incrementan materialmente el uso del tiempo:

- El tamaño del grupo de Reuso puede ser pequeño (4 es el valor usado, y 2 es el deseado).
- La operación asíncrona de grupos de Reuso contiguos puede ser poca pérdida, permitiendo así que el uso del tiempo de cada punto de acceso sea aprovechado totalmente.

(http://www.elprisma.com/apuntes/ingenieria_de_sistemas/redesinalambricas/default2.asp)

1.1.7.2 Modulación de Radio.

El espectro disponible es de 40 MHz, según el resultado de APPLE y 802.11, la frecuencia es “Desvanecida” cuando en una segunda o tercera trayectoria, es incrementada o decrementada la amplitud de la señal, la distribución de probabilidad de este tipo de “Desvanecimientos” se le denomina “rayleigh”, el desvanecimiento rayleigh es el factor que reduce la eficiencia de uso del espectro con pocos canales de ancho de banda, si es usada la señal de espectro expandido, la cual es 1 bit/símbolo, la segunda o tercera trayectoria va a causar un “Desvanecimiento” si la diferencia de la trayectoria es más pequeña que la mitad del intervalo del símbolo, entonces el promedio de bits debe de ser constante, en el espacio localizado en el espectro y el tipo de modulación seleccionado, el uso de ciertos símbolos codificados, proporcionarían una mejor resolución a la longitud de trayectoria.

Un espectro expandido de 1 símbolo y cada símbolo con una longitud de 7,11,13,31 bits, permitirá una velocidad de 10 a 2 Mbs promedio, el código ortogonal permite incrementar los bits por símbolo, si son 8 códigos ortogonales en 31 partes y si se incluye la polaridad, entonces es posible enviar 4 partes por símbolo para incrementar la utilización del espacio, entonces el espectro expandido puede proporcionar una reducción del “Desvanecimiento” rayleigh, y una disminución en la interferencia a la señal para que el mensaje sea transmitido satisfactoriamente, lo cual significa que se reduce el factor de Reuso, en una comunicación directa entre

estaciones de un grupo cuando no existe la infraestructura, una frecuencia común debe ser alternada para transmisión y recepción.

1.1.8 *Eficiencia del Tiempo*

El tiempo es importante para poder maximizar el servicio, al momento de diseñar la frecuencia en el espacio. El uso del tiempo está determinado por los protocolos y por los métodos de acceso que regularmente usen los canales de transmisión de la estación, las características del método de acceso para que se considere que tiene un tiempo eficiente, pueden estar limitadas por los métodos que sean utilizados.

Algunas de estas características son:

- Después de completar una transmisión/recepción, la comunicación debe de estar disponible para su siguiente uso, no debe de haber tiempos fijos entre la transmisión-recepción y sobre todo rellenar la longitud de un mensaje para complementar el espacio, es desperdiciarlo.
- La densidad de distribución geográfica y tiempo irregular de la demanda del tráfico deben ser conocidas, un factor de Reuso, es más eficiente por un uso secuencial del tiempo que por una división geográfica del área, para la comunicación en una área, se debe de considerar la posibilidad de que en áreas cercanas existan otras comunicaciones, la dirección del tráfico desde y hacia la estación no es igual, el uso de un canal simple de transmisión y recepción da una ventaja en el uso del tiempo.
- Para tráfico abundante, se debe de tener una “lista de espera” en la que se manejen por prioridades: “El primero en llegar, es el primero en salir”, además de poder modificar las prioridades.
- Establecer funciones para usar todo el ancho de banda del canal de comunicación, para que el tiempo que exista entre el comienzo de la transmisión y la disponibilidad de la comunicación, sea lo más corto posible.

- La conexión para mensajes debe ser más eficiente que la selección, particularmente al primer intento, sin embargo la selección puede ser eficiente en un segundo intento cuando la lista de las estaciones a seleccionar sea corta.

Para transacciones de tipo asíncrona, es deseable completar la transacción inicial antes de comenzar la siguiente. Deben completarse en el menor tiempo posible.

1.1.9 Límite de la longitud del Paquete y su Tiempo.

Cuando el paquete es más pequeño, la proporción del tiempo usado al acceder al canal, es mayor, aunque la carga pueda ser pequeña para algunas funciones, la transferencia y descarga de archivos son mejor administrados cuando la longitud del paquete es de buen tamaño, para minimizar el tiempo de transferencia, por lo contrario cuando son paquetes grandes, se incrementa la posibilidad de que el paquete tenga errores en el envío, en sistemas de radio el tamaño aproximado ideal es de 512 octetos o menos, un paquete con una longitud de 100-600 octetos puede permitir la salida oportuna de respuestas y datagramas prioritarios junto con los datagramas normales, es necesario de proveer formas para dividir los paquetes en segmentos dentro de las redes inalámbricas, las computadoras necesitan varios anchos de banda dependiendo del servicio a utilizar, transmisiones de datos, de vídeo y voz, etc. La opción es, si:

- El medio físico puede multiplexar de tal manera que un paquete sea un conjunto de servicios.
- El tiempo y prioridad es reservado para el paquete y los paquetes relacionados con él, la parte alta de la capa MAC es multiplexada.

La capacidad de compartir el tiempo de estos dos tipos de servicios ha incrementado la ventaja de optimizar la frecuencia en el espacio y los requerimientos para armar un sistema.

(http://www.elprisma.com/apuntes/ingenieria_de_sistemas/redesinalambricas/default2.asp)

1.2 *RouterOS Mikrotik*

El principal producto es un Linux basado en sistema operativo conocido como Mikrotik RouterOS, este consiste que los usuarios a su vez seleccionado un PC basado en la máquina de un enrutador de software, que permite funciones como firewall normas, VPN Servidor y Cliente, ancho de banda con de calidad de servicio, punto de acceso inalámbrico y otras características comúnmente utilizado para el enrutamiento y la conexión de redes, el sistema también es capaz de servir como un portal cautivo basado en punto de acceso del sistema, el sistema operativo tiene licencia en los niveles de escalada, cada uno la liberación de más de las características disponibles RouterOS medida que aumenta el nivel de número, concesión de licencias es la cuota de base y se intensifica con las características en libertad, existe un software llamado Winbox que ofrece una sofisticada interfaz gráfica de usuario para el sistema operativo RouterOS, este permite conexiones a través de FTP, telnet y SSH, también hay una API que permite crear aplicaciones personalizadas para la gestión y seguimiento.

1.2.1 *Características de los Routers*

Soporta muchas aplicaciones que pueden ser utilizados por el tamaño medio a grande proveedor de Internet, por ejemplo, OSPF, BGP, VPLS / MPLS. Todo en uno, RouterOS es un sistema versátil, y está muy bien apoyado por Mikrotik. (<http://www.mediafire.com/?mymt3jfyet>)

El software ofrece soporte prácticamente para todas las interfaces de red que el Kernel de Linux 2.6.16 apoya, siendo el sistema así muy estable.

1.2.2 *Estructura del RouterOS*

El Router es basado en el Kernel de Linux, y este puede ejecutarse desde discos IDE o módulos de memoria flash, siendo su diseño modular y a la vez estos son actualizables, guiándose en una interface gráfica amigable.

1.2.3 *RouterBoard Mikrotik*

Es la plataforma de hardware hecho por Mikrotik, estos son routers alimentados por el sistema operativo RouterOS inigualable y van desde pequeños CPEs inalámbricos, para routers de núcleo muy potente, cada uno para una tarea diferente y de acuerdo a la situación, esta es combinada con su línea de hardware, conocido como Mikrotik RouterBOARD.

1.2.4 *Tarjetas miniPCI inalámbricas*

Los dispositivos wireless (Routers ADSL) incluyen en su hardware una placa base madre, y un slot miniPCI, en ese slot miniPCI se incorpora una tarjeta Wireless, que puede ser extraída fácilmente y reubicada en otro sitio.

Mini PCI se añadió a la versión 2.2 del PCI para su uso en ordenadores portátiles, sino que utiliza una de 32 bits y 33 MHz de bus con conexiones de potencia y soporte para bus mastering y DMA, el tamaño estándar para tarjetas Mini PCI es de aproximadamente 1/4 de sus contrapartes de tamaño normal, como no se limita el acceso externo a la tarjeta de escritorio en comparación con las tarjetas PCI, hay limitaciones en las funciones que puede realizar, muchos dispositivos Mini PCI se desarrolló como Wi-Fi, Fast Ethernet, Bluetooth, módem, tarjetas de sonido, aceleradores criptográficos, SCSI, IDE - ATA, SATA y los controladores de las tarjetas de combinación, Mini tarjetas PCI se puede utilizar con la interfaz PCI-equipado de hardware estándar, utilizando para Mini PCI a PCI convertidores.

1.2.4.1 *Detalles técnicos de Mini PCI*

Las tarjetas Mini PCI tienen un consumo de 2W de potencia máxima, que también limita la funcionalidad que se puede implementar en este factor de forma, también son necesarias para apoyar la CLKRUN # PCI señal utiliza para iniciar y detener el reloj del PCI para fines de administración de energía, hay tres tarjetas de los factores de forma: Tipo I, Tipo II y tipo III. El conector de la tarjeta utilizada para cada tipo

incluyen: tipo I y II utilizan un conector de 100 pines de apilado, mientras que el Tipo III utiliza un conector de borde de 124 pines, es decir, el conector para los tipos I y II se diferencia de la de Tipo III, donde el conector está en el borde de una tarjeta, al igual que con un SO-DIMM. Los otros 24 pines proporcionan las señales extra que se requiere para dirigir la E / S de nuevo a través del conector del sistema (de audio, AC-Link, LAN, interfaz de línea de teléfono). Tipo II tienen RJ11 y RJ45 conectores montados. Estas tarjetas deben estar ubicadas en el borde de la computadora o estación de acoplamiento para que los puertos RJ11 y RJ45 se puedan montar para el acceso externo.

1.2.5 *Sistemas Integrados*

Contiene por lo general, un RouterBOARD 112 de 16MB además un CPU Mips 175Mhz, con un Soft Level 4, incluyendo placa ROUTER y software de control de usuarios, ancho de banda, QoS, firewall y VPN, estos soportan interfaces inalámbricas, ideal para trabajar como BRIDGE-CLIENTE, la cual incluye una placa MiniPCI R52 y Antena 5.8 de 19d.

1.2.6 *Accesorios*

Los accesorios hay una variedad, los más principales son:

CA/IN1 RB100 indoor case, para routerboard 112, no soporta el rougherboard Rb/564, requiere fuente de 18POW o POE. (2 holes)

CA/IN2 RB500 indoor case, para routerboard 532, no soporta el daughterboard rb/564, requiere fuente de 18pow o poe. (2 holes for Nfemale connector)

CA/IN3 RB500 large indoor case, para rb/532 + daughterboards, requiere fuente de 48v (48POE).

CA/IN4 RB500 indoor case, para routerboard 532, no soporta el daughterboard Rb/564, requiere fuente de 18pow o poe. (2 holes p/antena indoor)

AC/SWI Antena Omni c/Pigtail a Presion, esta viene junto a CA/IN4

CA/OTS outdoor case para usar con RB112, RB133 o RB532

CA/OT3 RB500 large outdoor case para usar con Doughterboard, incluye 1 Port Ethernet RJ45 Insulator externo.

RB/P54 POE passive

18POW low power 18V power supply

48POW high power 48V power supply para RB/500 series y RB/100 series.

(<http://www.adslnet.es/index.php/2006/04/18/tarjetas-minipci-wireless/>)

1.3 *Interfaces*

Las interfaces de red permiten a cualquier servidor que ejecute el servicio, enrutamiento y acceso remoto para comunicarse con otros equipos a través de redes privadas o públicas, las interfaces de red se relacionan con el servicio, enrutamiento y acceso remoto en dos aspectos: el hardware físico y la configuración de las interfaces de red.

1.3.1 *Hardware Físico*

Puede ser cualquier adaptador que se conecte al bus del sistema de un equipo que permita que se conecte a una red, la mayoría de los servidores que ejecutan el servicio enrutamiento y acceso de red disponen de al menos dos adaptadores de red, estos dos adaptadores son necesarios si el servidor que ejecuta el servicio enrutamiento y acceso remoto actúa como enrutador entre dos segmentos de red, un servidor que tiene dos o más adaptadores de red que se conectan con redes distintas se denomina de hosts múltiples, ya que estas son accesibles desde varias redes, tienen un mayor número de requisitos de seguridad que un servidor que se conecta a una sola red, cada

interfaz de red debe configurarse correctamente para proteger el servidor y las redes privadas a las que éste se conecta.

El servidor que ejecuta el servicio enrutamiento y acceso remoto suele detectar automáticamente todos los adaptadores de red cuando se ejecuta el Asistente para la instalación del servidor de enrutamiento y acceso remoto. (<http://technet.microsoft.com/es-es/library/cc784767%28WS.10%29.aspx>)

1.3.2 *Configuración de interfaces de red*

1.3.2.1 Interfaz Privada

Una interfaz privada es un adaptador de red que está físicamente conectado a una red privada, la mayoría de las redes privadas se configuran con un intervalo de direcciones IP de red privada, y la interfaz privada también se configura con una dirección privada. Dado que una red privada está compuesta, teóricamente, de usuarios y equipos, normalmente serán menos los aspectos de seguridad que tendrá que tener en cuenta para una interfaz privada que para una interfaz pública.

1.3.2.2 Interfaz pública

Una interfaz pública es un adaptador de red que está físicamente conectado a una red pública, como Internet, estas se configuran con una dirección IP pública y se puede configurar para que realice la traducción de direcciones de red (NAT), dado que a una interfaz pública teóricamente puede tener acceso cualquier persona, los aspectos sobre seguridad que deben considerarse que esta será mayor que para una interfaz privada.

1.3.2.3 Interfaz de marcado a petición

Se conectan con enrutadores específicos de redes públicas o privadas, esta puede ser una interfaz a petición que se activa sólo cuando es necesario o permanente cuando está siempre conectada, además de configurar cada interfaz de red como una interfaz

pública, privada o de marcado a petición, puede configurar filtros de paquetes, direcciones y otras opciones para las interfaces de red. Algunas opciones para las interfaces públicas, como el Servidor de seguridad básico no se encuentran disponibles para interfaces privadas.

1.3.3 Clasificación de Routers

1.3.3.1 Según el Alcance

Locales. Unen redes LAN.

Remotos. Unen redes LAN-WAN o WAN-WAN. Como son independientes del medio físico no tienen tantos problemas como los bridges para adaptarse a las WAN, a veces deben hacer doble encapsulado y direccionamiento.

1.3.3.2 Según la Conectividad

Routers simples. Sólo permiten un sistema MAC y un protocolo de red.

Router múltiple MAC. Tienen puertos para distintos tipos de red, el funcionamiento de conectividad es el mismo ya que los Routers no utilizan el protocolo MAC.

Routers multiprotocolo. Permiten enrutar diferentes protocolos (IP, IPX, etc.) de paquetes que llegan por cualquier puerto.

Routers multiprotocolo y múltiple MAC. Combina los dos anteriores. (<http://elsitiodetelecomunicaciones.iespana.es/routers.htm>)

1.3.4 Arquitectura de Routers

Se definen de la siguiente manera:

Plana (flat). Todos tienen el mismo nivel lógico y las redes también.

Jerárquica. Tienen varios niveles lógicos, normalmente 2.

Backbone. Se comunican entre Routers, a la vez cada Router tiene su dominio y en cada dominio se pueden formar subdominios.

Red. Routers conectados por las redes a las estaciones.

Mixtas. En grandes redes se dan combinaciones de las dos arquitecturas, como en Internet.

1.3.5 Comparativa entre Routers y Bridges

Estos dos dispositivos son de gran importancia dentro de cualquier red de comunicaciones, por las siguientes ventajas:

1.3.5.1 Ventajas de Bridges sobre Routers:

Los bridges son simples, como transparentes a los usuarios, se configuran con los cambios y las estaciones no se deben programar, estos pueden conectar diferentes protocolos de red, formando una red lógica, o sea, un dominio de la capa de red, esto permite movilidad de las estaciones, todo esto implica bajo coste.

Todos los protocolos admiten bridges. Por ejemplo: NetBios no admite Routers, normalmente siendo estos más rápidos.

1.3.5.2 Ventajas de Routers sobre Bridges

Los Routers aprovechan rutas redundantes, proporcionando más inmunidad a fallos, a la vez pueden solucionar problemas de congestión por los algoritmos de enrutamiento, siendo una gestión más sencilla, se pueden tratar aislados del entorno físico, soportando cualquier topología, estas pueden unir diferentes LAN y LAN-WAN fácilmente, lo que permite la optimización del ancho de banda por el enrutamiento y sobre todo no dependen de las características físicas de las redes, pueden formar redes universales, el Internet siempre necesita de un Router.

1.4 *Balanceo de Carga*

1.4.1 *Introducción*

Es la técnica usada para compartir el trabajo a realizar entre varios procesos, ordenadores, discos u otros recursos, está íntimamente ligado a los sistemas de multiprocesamiento, o que hacen uso de más de una unidad de procesamiento para realizar labores útiles, el balance de carga se mantiene gracias a un algoritmo que divide de la manera más equitativa posible el trabajo, para evitar, los así denominados cuellos de botella.

(<http://www.idg.es/computerworld/%C2%BFQue-es-el-balanceo-de-carga?secciones/articulo-111063>)

1.4.2 *Definición*

El balanceo de carga es la manera en que las peticiones de Internet son distribuidas sobre una fila de servidores. Existen varios métodos para realizar el balanceo de carga, desde el simple "Round Robin" hasta los equipos que reciben las peticiones, recogen información, en tiempo real, de la capacidad operativa de los equipos y la utilizan para enrutar dichas peticiones individualmente al servidor que se encuentre en mejor disposición de prestar el servicio adecuado, a la vez los balanceadores de carga pueden ser soluciones hardware, tales como routers y switches que incluyen software de balanceo de carga preparado para ello, y soluciones software que se instalan en el back end de los servidores.

1.4.3 *Balanceador de carga en RouterOS Mikrotik*

Un balanceador de carga es un dispositivo de hardware o software que se pone al frente de un conjunto de servidores que atienden una aplicación y, tal como su nombre lo indica, asigna o balancea las solicitudes que llegan de los clientes a los servidores usando algún algoritmo, permitiendo conectar hasta cuatro líneas de internet simultáneamente, con esto podemos aumentar “virtualmente” el Ancho de

Banda de nuestro servicio, distribuyendo las conexiones de los diferentes puertos de los equipos a través de las diferentes líneas de internet que tendremos conectadas al Router, es completamente configurable y administrable, el cual asegura de que un servidor no se vea sobrecargado por una gran afluencia de tráfico.

1.4.4 *Importancia*

Este sistema es un procedimiento esencial en las redes LAN o WAN, ya que es difícil determinar el número exacto de visitantes que tiene un servidor. En este caso, el balanceo de carga entre servidores Web es un método muy utilizado por muchas empresas de hosting para garantizar que sus páginas tendrán un despliegue rápido en los navegadores.

1.5 *Calidad de Servicio (QoS)*

1.5.1 *Historia de QoS*

Con el advenimiento de la televisión IP y telefonía IP, los mecanismos de calidad de servicio son cada vez más a disposición del usuario final, ya que añadir etiquetas de calidad de servicio a los datos ha ganado popularidad durante los años, pero luego perdió la atención, ejemplos de ellos son Frame Relay y ATM, recientemente, MPLS (técnica entre la capa 2 y 3) han ganado un poco de atención, sin embargo, Ethernet hoy puede ofrecer calidad de servicio y es, por lejos, la capa más popular de la tecnología 2.

1.5.2 *Definición de Calidad de Servicio (QoS)*

La Calidad de servicio (QoS (Quality of Service)) es un conjunto de requisitos de asistencia que la red debe cumplir para asegurar un nivel de servicio adecuado para la transmisión de los datos los cuales se basan en estándares de funcionalidad QoS, que permite que los programas en tiempo real optimicen el uso del ancho de banda de la red, asegurando cierto nivel de garantía de recursos de red suficientes, ofrece a una

red compartida un nivel de servicio similar al de una red dedicada, una garantía de QoS indica un nivel de servicio que permite que un programa transmita datos a una velocidad especificada y los entregue en un periodo de tiempo dado, como objetivo consigue un sistema de entrega garantizada del tráfico de la red, como los paquetes de protocolo internet (IP).

Las redes inalámbricas ofrecen una peor calidad de servicio que las redes cableadas, que no superan habitualmente los 10 Mbps, frente a los 100 que puede alcanzar una red normal y corriente. Por otra parte hay que tener en cuenta también la tasa de error debida a las interferencias, esta se puede situar alrededor de 10^{-4} frente a las 10^{-10} de las redes cableadas, esto significa que has 6 órdenes de magnitud de diferencia y eso es mucho.

(<http://qos.iespana.es/>)

Concluyendo, QoS o Calidad de Servicio (Quality of Service) son las tecnologías que garantizan la transmisión de cierta cantidad de datos en un tiempo dado, siendo la capacidad de dar un buen servicio, especialmente importante para ciertas aplicaciones tales como la transmisión de vídeo o voz.

1.5.2.1 QoS, CoS y ToS

Los acrónimos terminados en “oS” hacen referencia a la obtención de calidad de servicio en redes, llevando en ocasiones a situaciones equívocas por el mal uso de los mismos, si bien QoS es el único que refiere completamente a la Calidad de Servicio, englobando todas las técnicas que se encuentran en torno a ella, mientras que CoS (clase de servicio) y ToS (tipo de servicio) son, sencillamente, dos de las técnicas utilizadas para su obtención.

1.5.2.1.1 QoS (calidad de servicio)

QoS recoge varios parámetros o atributos que describen un servicio, caracterizándose por la reserva de ancho de banda, retardo extremo a extremo, Jitter y tasa de error.

1.5.2.1.2 CoS (clase de servicio)

Este término implica, dos procedimientos: en primer lugar la priorización de los distintos tipos de tráfico claramente definidos a través de la red y en segundo lugar, la definición de un pequeño número de clases de servicio a las que aplicarla, priorizando en los puntos de congestión de la red, donde las decisiones de priorización pueden ser realizadas por puentes y encaminadores, estas aplicaciones que requieren distinguir clases de servicio incluyen procesos transaccionales, el vídeo y cualquier otro tráfico sensible al tiempo.

No se debe confundir CoS con QoS, pues, a diferencia de QoS, CoS no garantiza ancho de banda o latencia, en cambio permite a los administradores de red solicitar prioridad para el tráfico basándose en la importancia de éste, independientemente de la diferenciación, tanto CoS como QoS categorizan el tráfico para asegurar que el tráfico considerado crítico siempre fluya por la red, a pesar del ancho de banda demandado o de las aplicaciones de menor importancia.

Existen posibles definiciones de tipos de calidad de servicio, pero la mayoría de las empresas definen las clases de tráfico por tipo de aplicación, tipo de dispositivo o por tipo de usuario. Hoy es además posible definir clases separadamente en routers o puentes individuales, pero suele ser poco práctico.

1.5.2.1.3 ToS (tipo de servicio)

El tipo de servicio es equivalente a un carril destinado a coches de uso compartido, se reserva ancho de banda con anticipación y después se asigna el tráfico que necesite preferencia, como el de voz o un CoS con prioridad, de modo que este tráfico pueda utilizar el ancho de banda reservado. ToS no implica, por lo tanto, ningún tipo de garantías, pero a la vez está incluido como uno de los campos en la tecnología de QoS denominada Diffserv (servicios diferenciados), dónde también es conocido como DiffServ codepoint (DSCP o punto de código Diffserv). Es un campo de 8 bits,

estando los dos últimos reservados. Con los otros 6 bits restantes es posible obtener 64 combinaciones o 'codepoint', de ellas, 48 son utilizadas para direccionar el espacio global y 16 son para uso local.

Parte del protocolo IP Versión 4 reserva un campo en el paquete IP para el tipo de servicio (IP TOS). En este campo se pueden especificar los atributos de fiabilidad, capacidad de procesamiento y retardos del servicio.

(<http://qos.iespana.es/capitulo2.htm>)

1.5.3 Clasificación de QoS.

Es posible realizar una clasificación de QoS bajo distintas especificaciones, así podríamos diferenciarla según el tipo de tráfico, dónde aplicarla, la reserva de recursos de la red y otros parámetros:

1.5.3.1 Según la sensibilidad del tráfico

Teniendo en cuenta la variedad de tráfico existente y los requerimientos de retardo, latencia y ancho de banda para cada tipo, nos encontramos con:

1.5.3.1.1 QoS muy sensible al retardo.

Este tipo abarca el tráfico de vídeo comprimido, para este caso es necesario garantizar la disponibilidad de una determinada y gran cantidad de ancho de banda reservado para este tráfico y un valor de retardo mínimo que asegure la correcta transmisión del mismo.

1.5.3.1.2 QoS algo sensible al retardo.

Como la resultante de la aplicación de la emulación de circuito, se garantiza hasta un cierto nivel de ancho de banda, aunque en menor valor, de la misma manera, será necesario asignar prioridades para la transmisión de los datos.

1.5.3.1.3 QoS muy sensible a pérdidas.

Como sucede con el tráfico tradicional si se garantiza un nivel de pérdidas de valor cero entonces nunca se descartarán paquetes ni se desbordarán los buffers de almacenamiento del flujo, lo que facilitará el control de transmisión, por otra parte, esta garantía se hace a nivel de acceso al medio (MAC) o en capas superiores, pero nunca a nivel físico.

1.5.3.1.4 QoS nada sensible.

Se podría hablar en el tráfico de servicios de noticias, la filosofía de este tipo de QoS es usar cualquier oportunidad de transmisión restante y asumir que la capacidad de los buffers posteriores es suficiente para llevarla a cabo, asignándole a este tipo de tráfico la prioridad más baja, a este tipo responden los algoritmos Best Effort o al mejor esfuerzo, utilizado en Internet.

1.5.3.2 Según quién solicite el nivel de calidad de servicio

Teniendo en cuenta que la petición de QoS puede ser realizada por el usuario final o por los conmutadores de la red, nos encontramos con:

1.5.3.2.1 QoS Implícita

En este tipo el router o conmutador asigna automáticamente los niveles de calidad servicio en función del criterio especificado por el administrador, como el tipo de aplicación, protocolo o dirección de origen, hoy en día todos los routers, y algunos conmutadores, ofrecen este tipo de QoS.

El proceso es el siguiente:

En primer lugar están las estaciones finales que son las que transmiten los paquetes, luego el conmutador o router, que le llegan los paquetes, realiza un estudio de los datos entrantes y los prioriza, repartiéndolos en diferentes colas según la prioridad

asignada, estos datos vuelven a ser transmitidos hacia el siguiente conmutador o router, donde se repite el proceso.

Las funciones son:

El Control de red que lo tiene el administrador, el lugar centralmente y sobre todo las técnicas que se realiza según unos patrones de tráfico

1.5.3.2.2 QoS Explícita

Permite al usuario o aplicación solicitar directamente un determinado nivel de servicio que han de respetar los conmutadores y routers. El proceso es:

Las estaciones finales, que en este caso transmiten una petición RSVP, si ésta es aceptada, los paquetes A, C, B, D, son transmitidos, para luego el conmutador o router con los datos entrantes sean priorizados de acuerdo a instrucciones del nodo de destino, pasando al próximo conmutador o router, donde se repetirá el proceso.

Las funciones son:

Control de red, que lo tiene el usuario o la aplicación. Es por lo tanto, más difícil de gestionar, luego las técnicas: IP Type of Service (ToS), RSVP.

1.5.3.3 Según las garantías

En esta clasificación se va a tener en cuenta la reserva de recursos del sistema para proporcionar los servicios.

1.5.3.3.1 QoS garantizada / Hard QoS

Es conocida como “Hard QoS”, en la que se produce una reserva absoluta de los recursos de la red para un tráfico determinado, asegurándose así unos niveles máximos de garantías para este tráfico.

1.5.3.3.2 QoS no garantizada / Lack of QoS

En una calidad de servicio sin garantías, el tráfico es transmitido por la red a expensas de lo que en ella pueda sucederle. Es el tipo de QoS correspondiente a los servicios Best Effort (Mejor servicio).

1.5.3.3.3 QoS servicios diferenciados/ Soft QoS

Conocida como “soft QoS” es el punto medio entre los dos tipos anteriores. Para este tipo se realiza una diferenciación de tráfico, siendo tratados algunos mejor que el resto (expedición más rápida, más ancho de banda promedio, menos tasa de error promedio).

1.5.3.4 Según el lugar de aplicación

Es posible aplicar calidad de servicio en los extremos y en los bordes de la red, por lo tanto tenemos:

1.5.3.4.1 QoS extremo a extremo (end-to-end)

Es la aplicación de las políticas de calidad de servicio entre los extremos de la red. Es viable gracias a productos como el software *Dynamic Access* de 3Com, pero está menos extendida que la QoS entre dos bordes de la red (edge-to-edge), también se la conoce comúnmente como la QoS absoluta, con este tipo de QoS se simplifican, sin embargo, los puentes, cuya función se reduce a observar la marca de los paquetes (en el caso de 802.1p), sin tener que calcular la clase de servicio de cada paquete reducido. Otra ventaja es que las aplicaciones podrían seleccionar dinámicamente el nivel de QoS, almacenándose temporalmente en los directorios de red o en los puentes una información estática de clases de servicio, actualmente, la política de las empresas dedicadas al networking es conseguir una calidad de servicio extremo a extremo, por lo que se están estudiando y aplicando diferentes técnicas para conseguirlo.

1.5.3.4.2 QoS borde a borde (edge-to-edge)

Es la aplicación de las políticas de calidad de servicio entre dos puntos cualesquiera de la red, ejemplo en los puentes, esto tiene varias ventajas: en primer lugar no requiere que los administradores de red toquen ninguno de los extremos, esto es una ventaja para el caso de las empresas en las que la organización responsable de la infraestructura de red está separada del grupo de los servidores y del resto de los puestos de trabajo, otra ventaja es que son menos los dispositivos que tienen que ser manejados para la obtención de la QoS, además, la accesibilidad por parte de un usuario cualquiera de la red o de un hacker para cambiar las especificaciones de QoS es mucho menor, a este tipo también se le conoce como calidad de servicio relativa.

1.5.4 *Parámetros de QoS*

Son muchos los términos manejados en la calidad de servicio, en el ámbito de las telecomunicaciones y de la informática.

1.5.4.1 Tráfico de Red

Son los datos que la atraviesan, dependiente del tipo de aplicación que por ella circulan. De esta manera podríamos establecer una diferenciación del tráfico.

Según el tipo de aplicación

Tendremos: tráfico habitual, multimedia, multicast, broadcast, tiempo real, etc.

Según la sensibilidad al retardo

- Tráfico algo sensible al retardo, los procesos de transacción on-line, la entrada de datos remota y algunos protocolos como SNA. Este tipo de aplicaciones requieren retardos de un segundo o, incluso, menos. Retardos mayores supondrían hacer esperar a los usuarios por la contestación a sus mensajes

antes de que puedan continuar trabajando, disminuyendo así la productividad de los negocios.

- Tráfico muy sensible al retardo, el tráfico en tiempo real es de este tipo, tal y como las conversaciones vocales, la videoconferencia y multimedia en tiempo real. Todos ellos requieren un retraso de tránsito muy pequeño (típicamente menos de una décima de segundo en un sentido, incluyendo el procesamiento en las estaciones finales) y un nivel de variación (jitter) mínimo.
- Tráfico muy sensible a las pérdidas, estos se puede dar en lo datos tradicionales.
- Tráfico nada sensible, puede ser en los servicios de noticias.

1.5.4.2 Retardo

Indica la variación temporal y/o retraso en la llegada de los flujos de datos a su destino. Es una característica que se hace muy evidente en aplicaciones como la video-conferencia, donde todos hemos experimentado alguna vez el retraso en la recepción de algún mensaje vocal enviado por nosotros y, por supuesto, el retardo existente entre la señal de voz y la señal de vídeo. Teniendo en cuenta hacia qué tipo de aplicaciones se están orientando las telecomunicaciones (es evidente la llegada de la voz sobre IP), es necesario que en las políticas de QoS definidas para nuestra red este parámetro sea reducido al mínimo.

1.5.4.3 Latencia

Es el tiempo entre el envío de un mensaje por parte de un nodo y la recepción del mensaje por otro nodo, abarca los retardos sufridos durante el propio camino.

1.5.4.4 JITTER (inestabilidad o variabilidad en el retardo)

Es lo que ocurre cuando los paquetes transmitidos en una red no llegan a su destino en debido orden o en la base de tiempo determinada, es decir, varían en latencia, algo

semejante a la distorsión de una señal, en redes de conmutación de paquetes, Jitter es una distorsión de los tiempos de llegada de los paquetes recibidos, comparados con los tiempos de los paquetes transmitidos originalmente, esta distorsión es particularmente perjudicial para el tráfico multimedia, una solución ante el jitter es la utilización de buffers en el receptor, pero esta es una medida poco eficaz, dado que sería necesario un gran tamaño para los buffers, lo que implica un costo económico en los equipos, y porque estos buffers incrementarían la latencia. El tamaño de uno de estos buffers debería ser al menos dos veces el valor del jitter y la latencia adicional introducida por el buffer podría superar el máximo de latencia permitido por la aplicación.

1.5.4.5 Ancho de Banda

Una medida de la capacidad de transmisión de datos, expresada generalmente en Kilobits por segundo (kbps) o en Megabits por segundo (Mbps). Indica la capacidad máxima teórica de una conexión, pero esta capacidad teórica se ve disminuida por factores negativos tales como el retardo de transmisión, que pueden causar un deterioro en la calidad, pero aumentar el ancho de banda significa poder transmitir más datos (algo así como aumentar el número de carriles de una autopista), pero también implica un incremento económico y, en ocasiones, resulta imposible su ampliación sin cambiar de tecnología de red.

1.5.4.6 Pérdida de Paquetes

Indica el número de paquetes perdidos durante la transmisión. Normalmente se mide en tanto por ciento, por ejemplo 1% o menos de media de pérdida de paquetes mensual de ancho de red

1.5.4.7 Disponibilidad

Indica la utilización de los diferentes recursos, de igual manera se especifica en tanto por ciento.

1.5.4.8 Rendimiento

Mide el rendimiento de la red en relación a los servicios acordados (SLAs o acuerdos de nivel de servicio), el rendimiento es definido también por algunos profesionales como la velocidad teórica de transmisión de los paquetes por la red. Esta depende directamente del ancho de banda y su variación de las posibles situaciones de congestión de la red.

1.5.4.9 Priorización

Priorizar consiste en la asignación de un determinado nivel de QoS al tráfico que circula por una red, asegurando así que las aplicaciones de mayor importancia sean atendidas con anterioridad a las de menor importancia, estando o no ante una situación de congestión. Es necesaria únicamente cuando la red no proporciona la suficiente capacidad para atender todo el tráfico presente en la misma.

1.5.4.10 Encolado

El encolado consiste en dividir y organizar el tráfico ante un determinado dispositivo de red para su posterior retransmisión por la misma según un determinado algoritmo que define a la cola y que permite que determinados paquetes sean reexpedidos antes que otros, es una de las herramientas más utilizadas por la QoS, la idea es ofrecer un mejor servicio al tráfico de alta prioridad al mismo tiempo que se asegura, en diferentes grados, el servicio para los paquetes de menor prioridad, sin embargo, los sistemas de colas, no garantizan que los datos importantes lleguen a su destino a tiempo cuando se produce congestión, lo único que aseguran es que los paquetes de alta prioridad llegarán antes que los de baja prioridad.

1.5.4.11 Planificación

Es el proceso de decidir qué paquetes enviar primero en un sistema de múltiples colas.

1.5.4.12 Flujo

Es el conjunto de datos pertenecientes a una misma secuencia que, debido a su gran tamaño, han de ser enviados mediante distintos paquete, estos tienen la misma dirección IP fuente y destino, el mismo puerto de destino y el mismo protocolo, el flujo, necesita, por tanto, llegar secuencialmente a su destino con una frecuencia constante, por lo tanto, el parámetro más importante para caracterizar un flujo será su frecuencia constante de bit (constant bit rate, CBR), que nos dará la frecuencia a la que debería ser transmitido cada bit de datos.

1.5.4.13 Acuerdo de Nivel de Servicio o Service Level Agreement(SLA)

Es un contrato de servicios entre un proveedor de servicios y su cliente, el cual define las responsabilidades del proveedor en términos del nivel de funcionamiento de la red (rendimiento, tasa de pérdidas, retrasos, variaciones) y la disponibilidad temporal, el método de medida, las consecuencias cuando los niveles de servicio no se consiguen o si los niveles de tráfico definidos son superados por el cliente, así como el precio de todos estos servicios. Evidentemente, y suele ser lo más común, el SLA puede incluir reglas de condicionamiento del tráfico.

1.5.4.14 Los SLA suelen subdividirse en:

SLS: Service Level Specifications o Especificaciones del Nivel de Servicio.

El SLS lleva a cabo el estudio del rendimiento de la red, la probabilidad de 'drop', la latencia, la espera en las entradas y/o salidas de los puntos donde se proporciona el servicio, indicando el 'scope' del mismo, así como de los perfiles del tráfico que se deben adherir para que el servicio solicitado pueda ser proporcionado y de la disposición del tráfico.

SLO: Service Level Objectives u Objetivos del Nivel de Servicio

Un SLO divide un SLA en objetivos individuales, definiendo métricas para hacer cumplir, para limpiar, y/o para vigilar el SLA. , para así determinar en que SLA se están cumpliendo los servicios (ej. Up-time, MTBF, tiempo de respuesta, MTTR).

1.5.4.15 Especificaciones del condicionamiento del tráfico

Aparte del acuerdo de nivel de servicios es necesario adjuntar unas funciones de control de los requisitos del tráfico para estudiar su comportamiento, observando el flujo de las aplicaciones, o cualquier otro subgrupo de tráfico operativo (ej. actualizar tablas de encaminamiento), algunas de estas funciones de control son la medición del tráfico, las políticas, el ‘shaping’ y el uso de marcas en los paquetes, se suelen utilizar en algunas de los protocolos utilizados para proporcionar QoS. En Diffserv, por ejemplo, se usa para hacer cumplir acuerdos entre los dominios. Un Traffic Conditioning Agreement (TCA) o Acuerdo de Condicionamiento del Tráfico, es un acuerdo que especifica las reglas para clasificar el tráfico bajo cualquier perfil, abarca todas las reglas de condicionamiento del tráfico especificadas explícitamente dentro de un SLA, junto con todas las reglas implícitas de los requisitos del servicio.

1.5.5 *QoS Basic Framework*

En enero de 1995 se propuso como recomendación/estándar el documento ISO/IEC JTC1/SC21, con título “QoS Basic Framework” en el que se realiza una descripción de los diversos conceptos, campos de aplicación, herramientas y todo un conjunto de definiciones de la aplicación de QoS en redes inteligentes, para proporcionar una base común a todos los posibles estándares nacientes sobre calidad de servicio. Todas las definiciones y estándares de QoS comentados en este proyecto lo utilizan como base.

1.5.6. *Algoritmos para la obtención de QoS*

El tipo de algoritmos utilizados actualmente en la transmisión de paquetes para comprobar cómo estos realizan un control de la congestión y a qué nivel son capaces de proporcionar calidad, así, teniendo en cuenta la clase de servicio que son capaces

de ofrecer los algoritmos de transmisión de paquetes podemos hacer tres divisiones principales:

1.5.6.1 Algoritmos de mejor esfuerzo (Best Effort)

En este tipo de algoritmos se encuentran los algoritmos tradicionales, que no ofrecen ningún tipo de garantías de transmisión, por lo que podría decirse que el nivel de calidad de servicio ofrecido es nulo, un ejemplo muy representativo es el FIFO (First In First Out), pero el principal problema de este tipo de algoritmos es que, si tenemos varios flujos de datos, una ráfaga de paquetes en uno de ellos va a afectar a todos los demás flujos, retardando su transmisión, es decir, que el tiempo de llegada de los paquetes de un flujo puede verse afectado por otros flujos, cuando esto ocurre decimos que el algoritmo utilizado no es capaz de aislar flujos.

1.5.6.2 Algoritmos deterministas

Son aquellos en los que, para evitar la posible congestión, antes de aceptar la transmisión de un flujo, se asegura que podrá transmitirse sin problemas incluso en las peores condiciones, esto se hace reservando ancho de banda, esto es el equivalente a lo que supondría un pico de una transmisión en ráfaga de ese flujo, con lo que se asegura que el flujo nunca se va a salir de su ancho de banda reservado, si suponemos este comportamiento en cada uno de los flujos de la red, podemos ver que la congestión es imposible, puesto que incluso en el caso en el que todos los flujos presentaran un pico al mismo tiempo, tendrían reservado el suficiente ancho de banda para que no hubiera congestión, en caso de que, por límites físicos de la red, no pudiera asegurarse ese ancho de banda, el algoritmo rechazaría la transmisión del flujo, este tipo de algoritmos fueron los primeros en aparecer cuando surgió la necesidad de asegurar las velocidades de transmisión, es obvio que consiguen su objetivo, pero lo consiguen a un precio muy elevado, puesto que son muy ineficientes respecto al uso de la red, las situaciones de ráfaga en un flujo son poco frecuentes y de muy corta duración, con lo que en la mayoría de los casos las necesidades de

ancho de banda del flujo son mucho menores, deduciendo, los algoritmos deterministas aíslan completamente los flujos.

1.5.6.3 Algoritmos intermedios

Aquellos algoritmos cuyo objetivo es ofrecer calidad de servicio y al mismo tiempo hacer un uso eficiente de los recursos. Entre estos podemos diferenciar entre los que ofrecen servicios estadísticos, servicios de degradación limitada y servicios predictivos. Estos algoritmos no aseguran una QoS tan estricta como los deterministas, pero en la mayoría de los casos consiguen un buen comportamiento y aprovechan mucho más los recursos disponibles. Como consecuencia, en estos algoritmos sí que es posible el retraso ocasional de algún paquete, con lo que si el algoritmo en cuestión se da cuenta de que un paquete ha superado su tiempo de expiración puede descartarlo directamente.

1.5.7 *Beneficios al aplicar QoS*

Los beneficios para las aplicaciones, de empresas y para los proveedores de servicio tendremos:

1.5.7.1 Ventajas para las aplicaciones

Hoy en día, todas las empresas están considerando Internet como una nueva vía para incrementar su negocio y, en consecuencia, las expectativas que se tienen para garantizar una calidad son las mismas que si se tratase de una red privada o controlada, el internet está siendo utilizado para la formación y el crecimiento de intranets dentro de la empresa y extranets que permiten el comercio electrónico con los socios del negocio, es evidente, por tanto, que se está incrementando el acercamiento de los negocios hacia la web, siendo cada vez más importante que los administradores de las redes aseguren que éstas entreguen unos niveles apropiados de calidad, es aquí donde las tecnologías de QoS cobran especial importancia.

1.5.7.2 Beneficios para las empresas

Las aplicaciones están consiguiendo ser cada vez más exigentes, las denominadas críticas requieren cada vez más calidad, confiabilidad, y asegurar la puntualidad en la entrega, un ejemplo claro son las aplicaciones de voz o vídeo, éstas deben ser manejadas cuidadosamente dentro de una red del IP para preservar su integridad, es necesario tener en cuenta que el tráfico no es predecible, ni constante, si no que funciona a ráfagas, produciéndose en ocasiones picos máximos de tráfico que son los causantes, en parte, de la saturación de la red, ejemplos clarificadores de este tipo de tráfico es el producido por el mundo web, el correo electrónico y las transferencias de ficheros, que son virtualmente imposibles de predecir.

Las tecnologías de QoS permiten a los administradores de red, manejar las aplicaciones sensibles al jitter, como las que manejan audio y vídeo, además manejar el tráfico sensible al retardo, como la voz en tiempo real y sobre todo el control de pérdidas en los momentos en los que la congestión sea inevitable.

1.5.7.3 Beneficios para los proveedores de servicio

Claramente, las empresas y las corporaciones se están convirtiendo en negocios con requerimientos de “misión-crítica” sobre la red pública, están delegando los servicios de sus redes a proveedores de servicio (outsourcing), lo que les permite centrarse más en el negocio interno y así reducir costosos capitales, esto significa que los proveedores de servicio son quienes podrán ofrecer las garantías de calidad para el tráfico extremo-a-extremo (end-to-end) de la empresa, las tecnologías de QoS permitirán a los proveedores de servicio ofrecer muchas más prestaciones, como el soporte del tráfico en tiempo real.

1.5.8 *Gestión del ancho de banda vs QoS*

La capacidad de cualquier tipo de sistema siempre, o casi siempre, acaba por agotarse; así, los discos duros se llenan o las líneas telefónicas de una centralita se saturan. Pero donde este límite se suele alcanzar con particular rapidez es en la capacidad de la línea que

conecta una organización con Internet (o en general con una red IP) ante el imparable crecimiento de las aplicaciones sobre este medio.

Lo normal es que, cuando las conexiones van lentas, se contrate más capacidad, pero, aun así, las líneas vuelven a saturarse tras un breve período de tiempo y es una solución costosa, esta es la técnica conocida como sobre ingeniería o método de la fuerza bruta, es necesario preguntarse entonces si ésta es la solución correcta y al estudiar otras alternativas vemos que con éstas se pueden obtener mayores capacidades por menos costes mediante la optimización de la gestión del ancho de banda, esto implica que al ampliar el ancho de banda debe utilizarse como una solución puntual para resolver determinadas situaciones de congestión en determinados puntos de la red y para determinados tipos de redes, es medianamente factible para redes LAN y prácticamente imposible para redes WAN, mientras los precios sigan siendo tan elevados, es por tanto, una solución costosa, con durabilidad mínima debido al crecimiento del tráfico de la red y de las necesidades de ancho de banda de determinados tipos de tráfico, sin embargo, la QoS conlleva, entre otras cosas, una correcta gestión del ancho de banda, presentándose como la forma más eficiente, hoy en día, para la mejora de toda red que se precie, en definitiva, la solución por la que deberían apostar todas las empresas para mejorar su red y, en consecuencia, su negocio.

1.5.9 Qos Ofrecida por algunos Sistemas Operativos

Normas aplicadas de QoS en los sistemas operativos, más extendidos: Windows 2000 de Microsoft e IOS de Cisco.

1.5.9.1 Calidad de servicio en Windows 2000

Windows 2000 soporta distintas técnicas de QoS IP que garantizan la transmisión de aplicaciones multimedia en tiempo real y la rápida entrega de grandes volúmenes de datos.

Los componentes QoS incluidos en el sistema operativo Windows 2000 son:

1.5.9.1.1 API GqoS (Generic Quality of Service).

Subconjunto de API Winsock que permite a las aplicaciones invocar servicios QoS del sistema operativo sin necesidad de comprender sus mecanismos subyacentes.

1.5.9.1.2 QoS Service Provider (SP)

Responde a las peticiones de API GqoS y proporciona señalización RSVP y soporte de políticas QoS, con Kerberos. Además invoca los mecanismos de control de tráfico.

Control de admisión ADS y protocolo SBM (Subnet Bandwidth Manager).

Para servicio de control de admisión sobre medios compartidos. ACS es un servicio de políticas que corre por encima de Windows 2000 Server y funciona en conjunción con SBM. ACS combina la funcionalidad de control de admisión basada en recursos de un SBM con el control de admisión basado en políticas que permite Active Directory.

1.5.9.1.3 Infraestructura de control de tráfico con soporte de DiffServ y 802.1p

El control de tráfico de Windows 2000 incluye además mecanismos adicionales como ATM (modo de transferencia asíncrono), en general Windows 2000 da soporte a los principales estándares QoS, como RSVP, 802.1p y Diffserv.

(http://es.wikipedia.org/wiki/Calidad_de_servicio)

1.5.9.2 Calidad de servicio en IOS de Cisco

El software de QoS IOS de Cisco permite controlar redes complejas y predecir los servicios de una gran variedad de aplicaciones de red y de tipos de tráfico, este software proporciona los siguientes beneficios:

Control de recursos. Permite tener control sobre cualquier recurso que está siendo utilizado, siendo posible, por ejemplo, limitar el ancho de banda consumido por una unión backbone, por transferencias FTP o dar prioridad al acceso de una importante base de datos.

Uso más eficiente de los recursos de red. Será posible conocer qué elementos está usando la red y cómo se están sirviendo al tráfico más importante de mi negocio.

Servicios adaptados. El control y la visibilidad proporcionadas por la QoS habilitan al proveedor de servicios de Internet a ofrecer diferentes tipos de servicios adaptados a sus clientes.

Coexistencia de aplicaciones de misión-crítica. Las tecnologías de QoS de Cisco permiten que la red sea usada eficientemente para este tipo de aplicaciones, disponiéndose el ancho de banda y los retardos mínimos requeridos por las aplicaciones sensibles al tiempo, así como su coexistencia con otras aplicaciones menos críticas, sin interferir.

El software de QoS IOS de Cisco utiliza, además, algoritmos de encolado para ordenar el tráfico y determinar así algún método de priorización para su retransmisión; Incluye los siguientes algoritmos:

FIFO: First in, first out (primero en entrar, primero en salir).

PQ: Priority Queuing (encolamiento por prioridad). Este tipo realiza priorización de tráfico.

CQ: Custom queuing (encolamiento por costumbre). Garantizando ancho de banda.

WFQ: Weighted fair queuing (encolamiento justo pesado). Es un algoritmo de encolamiento inteligente para las nuevas tecnologías de redes.

Además, este sistema operativo dispone de utilidades para evitar y solucionar la congestión, utilidades basadas en gestión de políticas y mecanismos de eficiencia en las uniones, sin olvidar los mecanismos de señalización como el IP-Precedence (utilizando el campo ToS), RSVP, 802.1p y Diffserv.

(http://es.wikipedia.org/wiki/Calidad_de_servicio)

1.6 *NetBeans*

Es un proyecto exitoso de código abierto, sun Microsystems fundó el proyecto de código abierto NetBeans en junio 2000 y continúa siendo el patrocinador principal de los proyectos, hoy en día hay disponibles dos productos: el NetBeans IDE y NetBeans Platform.

NetBeans IDE; es un entorno de desarrollo, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas, está escrito en Java, pero puede servir para cualquier otro lenguaje de programación, existe además un número importante de módulos para extender el NetBeans IDE, además NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

NetBeans Platform; una base modular y extensible usada como estructura de integración para crear grandes aplicaciones de escritorio, empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones.

Ambos productos son de código abierto y gratuito, para uso tanto comercial como no comercial, el código fuente está disponible para su reutilización de acuerdo con la Common Development and Distribution License (CDDL) v1.0 and the GNU General Public License (GPL) v2.

(http://netbeans.org/index_es.html)

1.7 *Telnet (TELEcommunication NETwork)*

Creada por Postel en 1980 es el nombre de un protocolo de red que sirve para acceder mediante una red a otra máquina para manejarla remotamente como si estuviéramos sentados delante de ella, también es el nombre del programa informático que implementa el cliente, para que la conexión funcione, como en todos los servicios de

Internet, la máquina a la que se acceda debe tener un programa especial que reciba y gestione las conexiones, el puerto que se utiliza generalmente es el 23.

Sirve para acceder en modo terminal, es decir, sin gráficos, pero fue una herramienta muy útil para arreglar fallos a distancia, sin necesidad de estar físicamente en el mismo sitio que la máquina que los tenía, también se usaba para consultar datos a distancia, como datos personales en máquinas accesibles por red, información bibliográfica, etc.

Aparte de estos usos, en general telnet se ha utilizado (y aún hoy se puede utilizar en su variante SSH) para abrir una sesión con una máquina UNIX, de modo que múltiples usuarios con cuenta en la máquina, se conectan, abren sesión y pueden trabajar utilizando esa máquina. Es una forma muy usual de trabajar con sistemas UNIX.

Este protocolo se usa para acceder a los BBS (*Bulletin Board System*), que inicialmente eran accesibles únicamente con un módem a través de la línea telefónica, para acceder a un BBS mediante telnet es necesario un cliente que dé soporte a gráficos ANSI y protocolos de transferencia de ficheros, los gráficos ANSI son muy usados entre los BBS, con los protocolos de transferencia de ficheros (el más común y el que mejor funciona es el ZModem) se podrá enviar y recibir ficheros del BBS, ya sean programas o juegos o ya sea el correo del BBS (correo local, de FidoNet u otras redes).

Algunos clientes de telnet (que soportan gráficos ANSI y protocolos de transferencias de ficheros como Zmodem y otros) son mTelnet!, NetRunner, Putty, Zoc, etc.

CAPITULO II

2 ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

2.1 Análisis de la Encuesta Aplicada a los Usuarios de Internet, de los Sistemas de Comunicación Latacunga, ubicados en los cantones de Latacunga, Pujilí, Saquisilí y Sigchos.

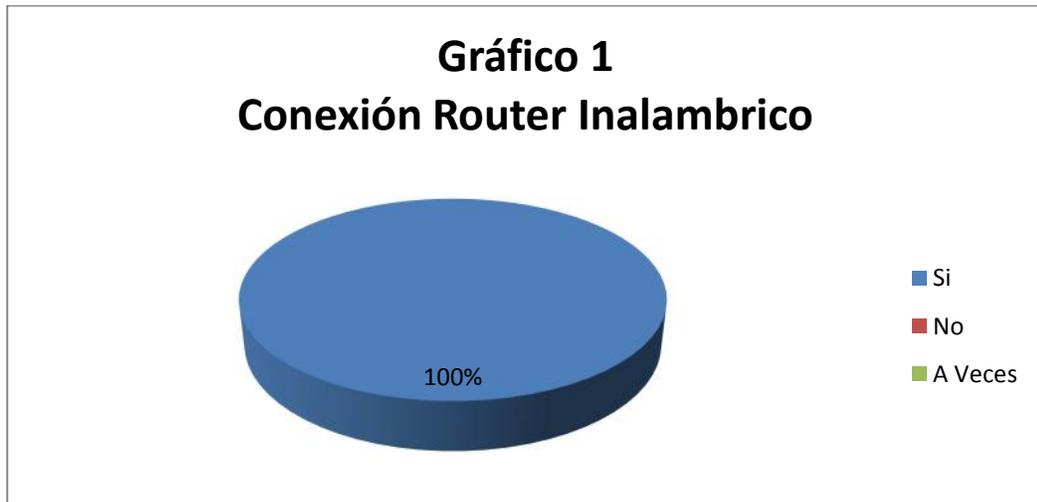
1.- ¿La conexión de su Router inalámbrico es permanente?

TABLA N° 1

Indicadores	F	%
Si	43	100%
No	0	0%
A veces	0	0%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

En la tabla numero 1 nos proporciona datos de un 100%, aseguran que la conexión del router inalámbrico es permanente.

Esto permitirá que el presente proyecto investigativo a emplear, funcione correctamente sin inconvenientes, y así permitirá tener en los usuarios calidad de servicio y clientes satisfechos.

2.- ¿Está conforme con la calidad del equipo inalámbrico instalado en su domicilio?

TABLA NUMERO 2

Indicadores	F	%
Si	37	86.05%
No	6	13.95%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

Los datos pertenecientes a la pregunta numero dos abordan el 86.05% a favor positiva, por otra parte el 13.95% en desacuerdo.

Conforme a esta realidad, los clientes saben de la calidad del equipo inalámbrico, pero inconformes con la calidad del servicio, un buen equipo es primordial para dar alcance con el sistema realizado, así que el 86.05% se siente conforme con la adquisición de los dispositivos siendo esto muy favorable a que el presente trabajo sea efectivo y seguro.

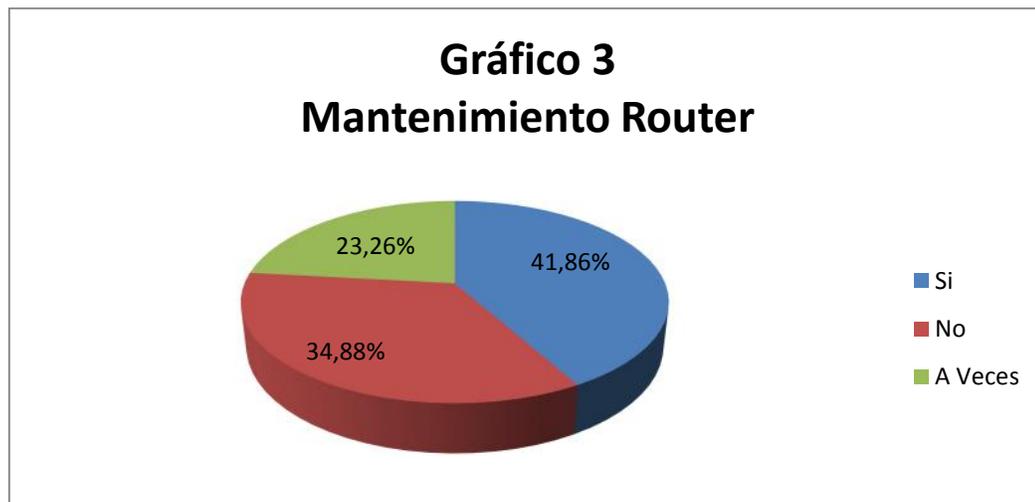
3.- ¿Es periódico el mantenimiento de su Router inalámbrico?

TABLA NUMERO 3

Indicadores	F	%
Si	18	41.86%
No	15	34.88%
A veces	10	23.26%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

Los datos pertenecientes a la tabla número tres, mencionan que el 41.86% afirma que es periódico el mantenimiento del router, el 34.88% mantiene la opción del no y el 23.26% indica que su Router es a veces dado mantenimiento.

El mantenimiento del router inalámbrico es casi periódico, es esencial para la durabilidad de este equipo, siendo mejor un mantenimiento preventivo a un correctivo, permitirá que el sistema no tenga pérdidas en el tráfico de datos.

4.- ¿Se debe aumentar el Ancho de Banda en los servidores por parte de Radio Latacunga?

TABLA NUMERO 4

Indicadores	F	%
Si	38	88.37%
No	5	11.63%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

Los resultados de la tabla numero 4 corresponde a un 88.37% que refleja el si, por otra parte el no aborda el 11,63%.

Tener un Ancho de Banda suficientemente, hoy en día es apropiado para trabajar acorde a nuestras necesidades, dado así el 88.37% menciona que se debe aumentar el ancho de banda, nuestro trabajo permitirá que el mismo ancho de banda, se entregue pero con una balanceo de carga y sobre todo con calidad de servicio, esto permitirá tener clientes satisfechos y la empresa no tener que invertir mas recursos.

5.- ¿Un servicio cableado mejoraría sus expectativas, al servicio inalámbrico que recibe?

TABLA NUMERO 5

Indicadores	F	%
Si	0	0%
No	43	100%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

En la presente tabla correspondiente a la pregunta 5, nos proporciona un resultado del 100% con la alternativa del sí y el 0% corresponde el no.

Hay que tomar en cuenta que al hablar de equipo inalámbrico se refiere al radio que se encuentra de enganche para recibir internet, el servicio cableado proporciona algunas ventajas en lo que se refiere a calidad de servicio, pero inalámbricamente podremos llegar a sitios extremadamente solitarios, pero es aquí donde nosotros necesitamos un balanceo de carga constante.

6.- ¿Es satisfactorio el Internet en su Equipo?

TABLA NUMERO 6

Indicadores	F	%
Si	17	39.53%
No	23	53.48%
A veces	3	6.97%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

Los datos correspondientes a la tabla nos verifica que el 39.53% abordan el si, por otra parte el 53.48% pertenece al no y el 6.97% menciona que a veces.

Este resultado permite determinar en síntesis que el Internet es malo, pero un sistema que controle y balancee la carga, ayudará a que el servicio sea óptimo y puedan recibir en su totalidad el plan contratado.

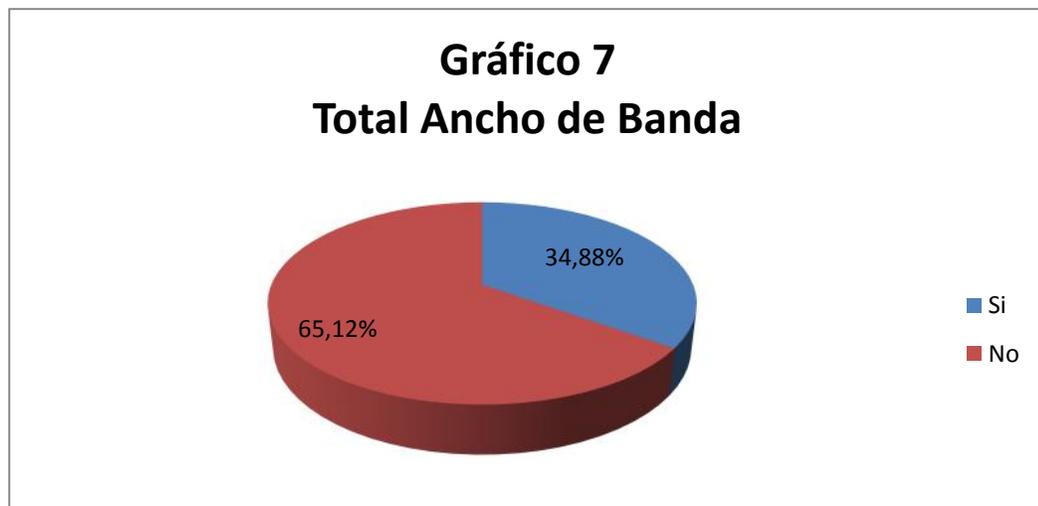
7.- ¿Recibe en su totalidad el ancho de banda deseado?

TABLA NUMERO 7

Indicadores	F	%
Si	15	34.88%
No	28	65.12%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

En referencia a la tabla número 7 el 34,12% opta por la opción del sí y el 65,88% el no.

El ancho de banda es importante, pero más importante es un correcto balanceo de carga, dado así que el 65.88% no está conforme con el ancho de banda que es emitido por parte de Radio Latacunga, pero nuestra propuesta ayudara inmediatamente a cubrir las necesidades de cada uno de los usuarios.

8.- ¿Tiene problemas de descarga en el Internet?

TABLA NUMERO 8

Indicadores	F	%
Si	23	53.49%
No	8	18.60%
A veces	12	27.91%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

Los datos abarcados en la tabla número 8 proporciona que el 53.49% hace referencia al sí, el 18.60% menciona que no y el 27.91% también tienen inconvenientes, pero estos son ocasionales.

Mencionando que el 53,49% corresponde a un porcentaje alto, los usuarios tienen dificultad con las descargas, a sabiendas que es la parte fundamental del internet, por lo tanto implementar el balanceo de carga permitirá que la descarga sea siempre equitativa y que el tamaño del archivo no sea un obstáculo.

9.- ¿El servicio de Internet le permite distribuir en varios equipos?

TABLA NUMERO 9

Indicadores	F	%
Si	40	93.02%
No	3	6.98%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

En la tabla numero 9 nos proporciona datos de un 93.02% que corresponde al si y tan solo un 6.98% menciona que no.

Un Internet Inalámbrico se puede distribuir en algunos equipos, claro está esto implica distribuir equitativamente el tráfico de datos, con este pequeño preámbulo nos damos cuenta que algunos clientes son cybers y al distribuir baja el ancho de banda, estos es dado porque no se tiene una correcta configuración de los QoS, ya sea en el sistema principal como en sus terminales.

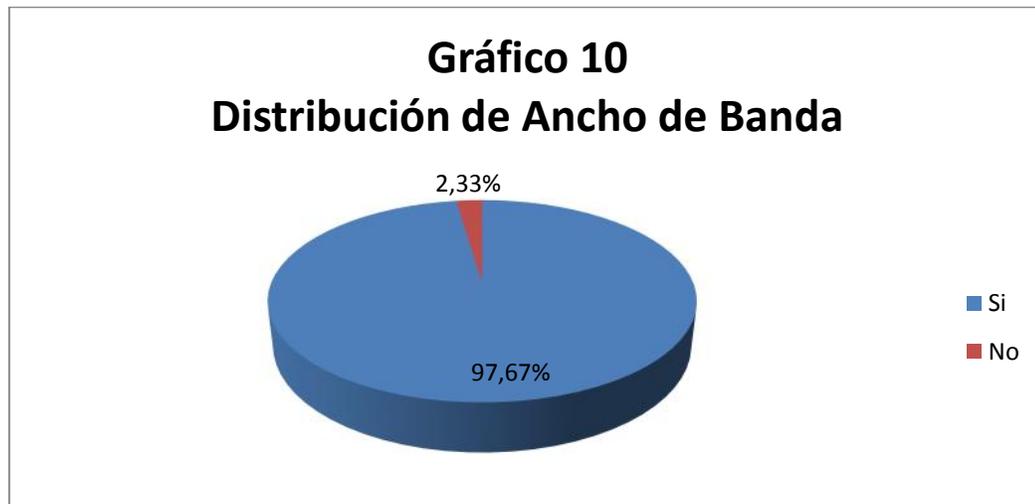
10.- ¿La distribución adecuada de Ancho de Banda mejorara el servicio de Internet que usted recibe?

TABLA NUMERO 10

Indicadores	F	%
Si	42	97.67%
No	1	2.33%
TOTAL	43	100%

Fuente: Encuesta usuarios

Elaborado por: Investigadores



Análisis e interpretación

Los resultados que abordan la tabla numero 10 corresponde a un 97.67% que mencionan el sí y el 2.33% el no.

Obviamente mejorará, el 97.67% lo asegura, pero más aún, un correcto balanceo de carga superará todas las expectativas tanto comercial, administrativa y técnica, que implementar el presente sistema se abarcara a solucionar todos los inconvenientes que se tiene con el tráfico de la red.

2.2Entrevista al Gerente de Radio Latacunga, Lic. Eduardo Guerrero.

Lunes 10H30, 19 de abril de 2011

1 ¿Cuál es su apreciación por parte de usted, frente a la implementación del Sistema de control de manejo de banda para la empresa?

La empresa posee 5 puntos de acceso los cuales permite acceder desde varios rincones de nuestros queridas parroquias, como cantones, esto nos ha permitido crecer a pasos agigantados despertando aún más el interés por mejorar las cosas ya sea en la parte administrativa, comercial y fundamentalmente en la parte técnica la cual se viene trabajando constantemente para optimizar aún más la red, un trabajo muy esencial e importante, como de magnitud es el trabajo investigativo que se da por parte de los postulantes en la realización de la tesis, al implementar un sistema de control, que me permita el balanceo de carga y ancho de banda en nuestros Routers Mikrotik y sobre todo lo más fundamental con calidad de servicio (QoS), cabe recalcar este es un aporte muy importante a la empresa, porque primeramente es una de las pioneras en poseer Routers Mikrotik en la Provincia, y optimizar esta tecnología con estándares de calidad.

2 ¿Cómo mejorara la implementación del Sistema de Control, Balanceo de Carga y Ancho de Banda por Routers Mikrotik con Calidad de Servicio (QoS), para la red wlan (wireless local area network) en los Sistemas de Comunicación Latacunga el servicio de internet a los diferentes usuarios de la presente empresa?

La implementación del presente trabajo investigativo permitirá interactuar de forma directa de una manera fácil y efectiva en nuestro sistema, facilitando de esta manera controlar, modificar el ancho de banda, con un balanceo de carga correcto, persistente, pero más aún un sistema que permita dar calidad de servicio a los usuarios.

Conclusiones

- La presente encuesta permitió conocer el estado actual en que se encontraba la empresa, observando claramente que se necesitaba un balanceo óptimo de carga en la red Wlan de los Sistemas de Comunicación Latacunga, mejorando tanto el área administrativa, comercial y técnica, gracias a un consumo equitativo.
- La insatisfacción de los clientes fue notoria, porque no existía un control de tráfico en la red Wlan, esto permitió que la red se vea de una forma imperfecta o en el peor del caso pésimo, pero gracias a la implementación de este sistema, mejoro la transmisión de datos.
- Se concluye que los momentos para realizar nuestra investigación parten de un estudio bibliográfico y de campo, y las técnicas utilizadas para el estudio de campo fueron la entrevista y la encuesta, que dichos resultados permitieron obligaron a implementar el presente trabajo investigativo.
- En base a los resultados obtenidos, podemos finiquitar qué, el control y balance de carga, permite obtener Calidad de Servicio en la red Wlan de los Sistemas de Comunicación Latacunga, obteniendo que la hipótesis planteada sea comprobada de una manera clara y precisa.

Recomendaciones

- Al implementar el presente sistema mejora el tráfico de red hacia el equipo inalámbrico que se tiene instalado en cada uno de los usuarios, pero la distribución interna es decir la intranet del cliente es responsabilidad del mismo.
- El sistema Pinbox-MK v1.0 permite de una manera amplia solucionar cada uno de los inconvenientes presentados al momento de la entrevista, pero su correcta utilización será responsabilidad única del administrador.

- La utilización de las técnicas adecuadas para un estudio de campo son primordiales ya que representan la parte fundamental para ser objeto de investigación.

CAPITULO III

3 APLICACIÓN O VALIDACIÓN DE LA PROPUESTA

Tema: Diseño e implementación de un sistema de control y balanceo de carga, en Routers Mikrotik con calidad de servicio (QoS), para la red Wlan (wireless local area network) de los “Sistemas de Comunicación Latacunga”.

La realización de un sistema siempre va a ir en función de seguir mejorando, agilitando y sobre todo facilitando las ideas de las empresa, permitiendo que siempre evolucionen los sistemas, en función de esta situación el presente trabajo investigativo menciona administrar una red Wlan de una manera fácil, amigable y sobre todo exacta al momento de un control y balanceo de carga.

Al desarrollar el presente sistema nos podemos regir a herramientas fundamentales con la que está desarrollado, principalmente utilizamos RouterOS Mikrotik, para ingresar a este, necesitaremos de una herramienta básica pero importante como es Telnet y la programación del presente software nos basaremos en Java con una estructuración en el Netbeans.

3.1 *Objetivos*

3.1.1 *Objetivo General*

Diseñar e Implementar un sistema de control, balanceo de carga en Routers Mikrotik con calidad de servicio (QOS), para la red wlan de los sistemas de comunicación Latacunga.

3.1.2 *Objetivos Específicos*

- Analizar la estructuración y configuración de Routers Mikrotik, empleados y diseñados para redes inalámbricas para establecer el uso de balanceo de carga.
- Diseñar un software o sistema visual de fácil manejo y eficaz en la distribución, determinación, y asignación de conectividad en el uso de ancho de banda aplicado para Routers Mikrotik.
- Implantar, el sistema de control y balanceo de carga y pruebas de ejecución en la red WLAN (Wireless Local Área Network) de los sistemas de Comunicación Latacunga.

3.2 *Justificación*

Un sistema de control de ancho de banda con balance de carga para Router Mikrotik permite agilizar el nivel de enlace de internet e los equipos de radio Latacunga ya que el Balanceo de Carga es un recurso muy importante porque es un algoritmo que distribuye la carga de forma uniforme entre los enlaces disponibles para conseguir máxima eficacia y alta disponibilidad, es decir podemos aumentar virtualmente el ancho de banda de nuestro Mikrotik, siendo este configurable y administrable, permitiendo que este sistema distribuya equilibrada y estrictamente el internet hacia

los usuarios, lo que provocara mayor facilidad a los Sistemas de Comunicación Latacunga distribuir el servicio de acuerdo a las necesidades de los clientes.

Los beneficiarios directos de la propuesta serán los usuarios de Radio Latacunga, y los mismos Sistemas de Comunicación, además de la comunidad latacungeña y los investigadores que ponen en práctica sus conocimientos adquiridos.

3.3Diseño de la propuesta

3.3.1Datos informativos de la empresa

"Los Sistemas de Comunicación Latacunga" están ubicados en la Región Interandina de Ecuador, Provincia de Cotopaxi, ciudad de Latacunga, a 90 Km. al sur de Quito capital de la República; está a 2700 m. sobre el nivel del mar.

Eclesiásticamente se ubica en la Diócesis de Latacunga, siendo el Director General Lic. Luis Eduardo Guerrero Guevara

Esquema general de la red Wlan de Radio Latacunga.

La red Wlan de radio Latacunga está compuesta por un ISP, un servidor con sistema operativo Linux, seis radios de marca Mikrotik. Todos estos radios están enlazados mediante accesos wireless, de manera que se encuentran dentro de una red alcanzable y registrada cada una de las rutas hacia dichos equipos.

Para la utilización del sistema Pingbox-MK utilizaremos rutas destinadas y asignadas a cada una de la interfaces de los equipos de la red de radio Latacunga.

3.3.2 Metodología

La Metodología para aplicar el presente sistema de control y balanceo de carga en Routers Mikrotik con calidad de servicio (QoS) para la red Wlan (Local Area Network) de los Sistemas de Comunicación Latacunga, consiste en un proceso que se ha denominado de la siguiente manera: Fase de planificación, fase de diseño, fase de instalación, fase de prueba y por último la fase de funcionamiento.

3.3.3 Análisis y Diseño de la aplicación Informática

3.3.3.1 Fase de planificación

La presente tiene como prioridad Diseñar e Implementar un sistema de control y balanceo de carga y ancho de banda para RouterOS Mikrotik con calidad de servicio (QOS), para la red wlan de los sistemas de Comunicación Latacunga, que permita llegar a cada uno de los usuarios de una manera eficaz el ancho de banda solicitada, como también ayudara a la empresa a controlar este servicio.

3.3.3.2 Fase de diseño

3.3.3.2.1 Manual del Programador

En el siguiente apartado detallaremos el funcionamiento de RouterOS Mikrotik y su acceso a las colas simples, así como la administración del mismo mediante acceso remoto a través del Puerto 23 perteneciente a Telnet.

3.3.3.2.2 Comandos básicos de RouterOS Mikrotik

Describiremos algunos de los comandos para realizar las configuraciones básicas de Mikrotik desde la interfaz de línea de comando, y los cuales serán objeto del estudio para el diseño del proyecto.

Algunas cosas a tener en cuenta son las siguientes:

1. Mikrotik posee una estructura jerárquica de directorio (al igual que Linux), esto quiere decir que cada instrucción tiene su propio lugar dentro de esta estructura.
2. Mikrotik posee diferentes tipos de usuario, y dependiendo del tipo de permisos que sean asignados a ellos podrán o no realizar ciertas tareas.
3. Todos los usuarios entran al mismo modo (no existe un modo de usuario privilegiado) y estos usuarios se diferencian por sus permisos.
4. La tecla de tabulación (TAB) sirve para autocompletar los comandos (igual a Linux).
5. Para ver la lista de comandos factibles de utilizar digita un signo de interrogación “?” (Observa que nuestra CLI reconoce el teclado el ingles, por

lo tanto para ingresar este símbolo la combinación de teclas que debes realizar es “SHIFT” + “-“)

6. Para retroceder un directorio en la jerarquía el comando es “..” (esto reemplaza al “cd ..” de Linux).
7. Para volver directamente a la raíz de nuestro árbol de directorio el comando es “/” (al igual que Linux).
8. Todas las modificaciones que realicemos quedan activas inmediatamente y estarán activadas para el próximo booteo de nuestra máquina (NO EXISTE EL COMANDO WRITE).
9. El comando “print” de Mikrotik muestra la configuración del archivo o driver.
10. Las interfaces ethernet reciben el nombre de “etherx” (donde x es el número de identificación), y las interfaces Wireless reciben el nombre de “wlanx” (donde x es el número de identificación).

3.3.3.2.3 Acceso a Mikrotik mediante acceso remoto.

Ejecutamos una consola en nuestro Windows, verificamos si tenemos habilitado el “Telnet” y procedemos a realizar la conexión remota hacia Mikrotik digitando el siguiente comando

>telnet “Dirección IP RouterOS Mikrotik”, ejemplo;

```
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.
C:\Users\Fix>telnet 192.168.1.1
```

Gráfico III.1 Conexión remota hacia Mikrotik

Luego de teclear “ENTER” e ingresara al sistema de logueo para acceder a RouterOS Mikrotik, e ingresamos el usuario y contraseña de acceso al equipo;

```
MikroTik v3.20
Login: admin
Password: _
```

Gráfico III.2 Acceso a RouterOS

Luego de autenticar la conexión estaremos dentro del sistema RouterOS de Mikrotik; listo ya ingresamos en el sistema:

```
MMM      MMM      KKK      TTTTTTTTTT      KKK
MMMM     MMMM     KKK      TTTTTTTTTT      KKK
MMM MMMM MMM III  KKK  KKK  RRRRRR  000000  TTT      III  KKK  KKK
MMM  MM  MMM  III  KKKKK  RRR  RRR  000 000  TTT      III  KKKKK
MMM      MMM  III  KKK  KKK  RRRRRR  000 000  TTT      III  KKK  KKK
MMM      MMM  III  KKK  KKK  RRR  RRR  000000  TTT      III  KKK  KKK

MikroTik RouterOS 3.20 <c> 1999-2009      http://www.mikrotik.com/

[admin@MikroTik] >
```

Gráfico III.3 Consola del RouterOS Mikrotik

Ahora nos dirigiremos hacia el directorio de colas simples atreves de los siguientes comandos:

```
[admin@MikroTik] > queue simple
```

Gráfico III.4 Acceso directorio colas simples

La forma más sencilla de limitar la velocidad de datos de direcciones IP y/o subredes, es el uso de colas de simple.

También puede utilizar las colas simples para construir aplicaciones avanzadas de QoS. Tienen útiles funciones integradas,

```
[admin@MikroTik] /queue simple> _
```

Gráfico III.5 Acceso directorio colas simples 2

Esto no dará acceso hacia el directorio de colas simples, un ejemplo de balanceo de carga mediante comandos se ingresa una regla de cola simple, lo que limitará el tráfico de descarga de 512 kbps y 256 kbps de subida a la red 10.1.1.0/24, servido por la interfaz Ether2, “No aplica calidad de servicio”:

```
[admin@MikroTik] /queue simple> add name=private target-  
addresses=10.1.1.0/24 max-limit=256K/512K \interface=ether2
```

Gráfico III.6 Limitación tráfico de descarga

Donde:

- Add: comando para agregar una nueva regla de balanceo de carga.
- Name: nombre designado por el usuario hacia la dirección a limitar el consumo de ancho de banda.
- Target-addresses: dirección IP a la cual se aplicara la regla de balanceo de carga.
- Max-limit: el limite maxi permitido de subida y bajada al momento del consume de internet.
- Interface: interface designada aplicar la regla.

Ejecutamos el comando “print” donde se imprimirá en pantalla todas las reglas de balanceo de carga ingresadas en las colas simples.

```
[admin@MikroTik] /queue simple> print  
Flags: X - disabled, I - invalid, D - dynamic  
0 name="private" target-addresses=10.1.1.0/24 dst-address=0.0.0.0/  
interface=ether2 parent=none direction=both priority=8  
queue=default-small/default-small limit-at=0/0 max-limit=256k/512k  
burst-limit=0/0 burst-threshold=0/0 burst-time=0s/0s
```

```
total-queue=default-small
```

Gráfico III.7 Impresión reglas balanceo de carga

Existen otros parámetros de configuración que no nos extenderemos en este proyecto.

Como ya hemos analizado la forma de limitar y realizar el control y balanceo de carga aplicaremos este tipo de comando interpretándolos en base a una interfaz gráfica.

La plataforma de programación seleccionada es NetBeans IDE 6.9.1 de donde partiremos nuestro proyecto.

El acceso hacia el equipo se lo realizo con la herramienta “Telnet”, de manera que buscaremos una librería para java que me pueda permitir iniciar una instancia dentro de la aplicación usando “Telnet”. Explicaremos cada una de las herramientas, variables, formularios, y código de programación de la aplicación.

3.3.3.2.4 Programación interfaz gráfica de acceso

Analizaremos el entorno grafico que hemos diseñado para el acceso a login del sistema RouterOS de Mikrotik.

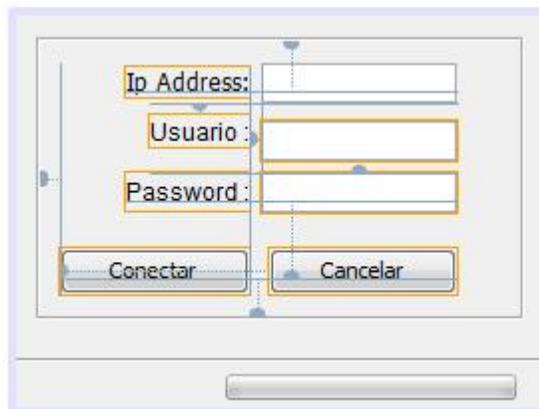


Gráfico III.8 Acceso a login

El código fuente está distribuido y programado para cada una de las funciones, se utilizó:

Tres etiquetas: para identificar las cajas de texto

Tres cajas de texto: para el ingreso de información

Dos botones: para ejecutar la conexión y cancelar o cerrar el programa.

3.3.3.2.5 Librerías utilizadas

```
/*
 * PingboxView.java
 */

package pingbox;

import java.util.logging.Level;
import java.util.logging.Logger;
import org.jdesktop.application.Action;
import org.jdesktop.application.ResourceMap;
import org.jdesktop.application.SingleFrameApplication;
import org.jdesktop.application.FrameView;
import org.jdesktop.application.TaskMonitor;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.ArrayList;
import java.util.List;
import javax.swing.Timer;
import javax.swing.Icon;
import javax.swing.JDialog;
import javax.swing.JFrame;
```

```
import javax.swing.JOptionPane;
```

Gráfico III.9 Librerías

Ya con las librerías creamos una clase pública para asignar los valores a ingresar

```
public class PingboxLogin extends JFrameView {

    public PingboxLogin(SingleFrameApplication app) {
        super(app);

        initComponents();

        // status bar initialization - message timeout, idle icon and busy animation, etc
        ResourceMap resourceMap = getResourceMap();
        int messageTimeout =
resourceMap.getInteger("StatusBar.messageTimeout");
        messageTimer = new Timer(messageTimeout, new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                statusMessageLabel.setText("");
            }
        });
        messageTimer.setRepeats(false);
        int busyAnimationRate =
resourceMap.getInteger("StatusBar.busyAnimationRate");
        for (int i = 0; i < busyIcons.length; i++) {
            busyIcons[i] = resourceMap.getIcon("StatusBar.busyIcons[" + i + "]");
        }
        busyIconTimer = new Timer(busyAnimationRate, new ActionListener() {
            public void actionPerformed(ActionEvent e) {
```

```

        busyIconIndex = (busyIconIndex + 1) % busyIcons.length;
        statusAnimationLabel.setIcon(busyIcons[busyIconIndex]);
    }
});
idleIcon = resourceMap.getIcon("StatusBar.idleIcon");
statusAnimationLabel.setIcon(idleIcon);
progressBar.setVisible(false);

// connecting action tasks to status bar via TaskMonitor
TaskMonitor taskMonitor = new
TaskMonitor(getApplication().getContext());
    taskMonitor.addPropertyChangeListener(new
java.beans.PropertyChangeListener() {
        public void propertyChange(java.beans.PropertyChangeEvent evt) {
            String propertyName = evt.getPropertyName();
            if ("started".equals(propertyName)) {
                if (!busyIconTimer.isRunning()) {
                    statusAnimationLabel.setIcon(busyIcons[0]);
                    busyIconIndex = 0;
                    busyIconTimer.start();
                }
                progressBar.setVisible(true);
                progressBar.setIndeterminate(true);
            } else if ("done".equals(propertyName)) {
                busyIconTimer.stop();
                statusAnimationLabel.setIcon(idleIcon);
                progressBar.setVisible(false);
                progressBar.setValue(0);
            } else if ("message".equals(propertyName)) {

```

```

        String text = (String)(evt.getNewValue());
        statusMessageLabel.setText((text == null) ? "" : text);
        messageTimer.restart();
    } else if ("progress".equals(propertyName)) {
        int value = (Integer)(evt.getNewValue());
        progressBar.setVisible(true);
        progressBar.setIndeterminate(false);
progressBar.setValue(value);
    }
    }
});
}

```

Gráfico III.10 Creación Clase Pública

Ingresamos los textbox y asignamos parámetros

```

public void showAboutBox() {
    if (aboutBox == null) {
        JFrame mainFrame = PingboxApp.getApplication().getMainFrame();
        aboutBox = new PingboxAboutBox(mainFrame);
        aboutBox.setLocationRelativeTo(mainFrame);
    }
    PingboxApp.getApplication().show(aboutBox);
}
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">

```

```

private void initComponents() {

    mainPanel = new javax.swing.JPanel();
    jPanel2 = new javax.swing.JPanel();
    jLabel3 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jBConectar = new javax.swing.JButton();
    jBCancelar = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();
    jPasswordField = new javax.swing.JPasswordField();
    jTextField1 = new javax.swing.JTextField();
    jTextField2 = new javax.swing.JTextField();
    statusPanel = new javax.swing.JPanel();
    javax.swing.JSeparator statusPanelSeparator = new javax.swing.JSeparator();
    statusMessageLabel = new javax.swing.JLabel();
    statusAnimationLabel = new javax.swing.JLabel();
    progressBar = new javax.swing.JProgressBar();
    jProuterdatos = new javax.swing.JPanel();
    jScrollPane2 = new javax.swing.JScrollPane();
    jTdatosrouter = new javax.swing.JTable();
    label1 = new java.awt.Label();

    mainPanel.setCursor(new
java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));

    mainPanel.setDebugGraphicsOptions(javax.swing.DebugGraphics.NONE_OPTIO
N);

    mainPanel.setInheritsPopupMenu(true);
    mainPanel.setName("mainPanel"); // NOI18N

```

```

jPanel2.setBorder(javax.swing.BorderFactory.createEtchedBorder());
jPanel2.setName("jPanel2"); // NOI18N

org.jdesktop.application.ResourceMap resourceMap =
org.jdesktop.application.Application.getInstance(pingbox.PingboxApp.class).getC
ontext().getResourceMap(PingboxLogin.class);
jLabel3.setFont(resourceMap.getFont("jLabel3.font")); // NOI18N
jLabel3.setText(resourceMap.getString("jLabel3.text")); // NOI18N
jLabel3.setName("jLabel3"); // NOI18N

jLabel2.setFont(resourceMap.getFont("jLabel2.font")); // NOI18N
jLabel2.setText(resourceMap.getString("jLabel2.text")); // NOI18N
jLabel2.setName("jLabel2"); // NOI18N

jBConectar.setText(resourceMap.getString("jBConectar.text")); // NOI18N
jBConectar.setName("jBConectar"); // NOI18N
jBConectar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jBConectarActionPerformed(evt);
    }
});

javax.swing.ActionMap actionMap =
org.jdesktop.application.Application.getInstance(pingbox.PingboxApp.class).getC
ontext().getActionMap(PingboxLogin.class, this);
jBCancelar.setAction(actionMap.get("quit")); // NOI18N
jBCancelar.setText(resourceMap.getString("jBCancelar.text")); // NOI18N
jBCancelar.setName("jBCancelar"); // NOI18N

```

```

jBCancelar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jBCancelarActionPerformed(evt);
    }
});

jLabel1.setFont(resourceMap.getFont("jLabel1.font")); // NOI18N
jLabel1.setText(resourceMap.getString("jLabel1.text")); // NOI18N
jLabel1.setName("jLabel1"); // NOI18N

jPassword.setText(resourceMap.getString("jPassword.text")); // NOI18N
jPassword.setName("jPassword"); // NOI18N

jTextField1.setName("jTextField1"); // NOI18N

jTextField2.setName("jTextField2"); // NOI18N

javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addGap(10, 10, 10)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addContainerGap(10, true)
            .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .add(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.HORIZONTAL)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .add(jPassword, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.HORIZONTAL)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .add(jTextField1, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.HORIZONTAL)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .add(jTextField2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.HORIZONTAL)
                )
            )
        )
    )
);

```

```

        .addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.TRAILING)
        .addComponent(jLabel1)
        .addComponent(jLabel3)
        .addComponent(jLabel2)
        .addComponent(jBConectar,
javax.swing.GroupLayout.PREFERRED_SIZE, 95,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.LEADING, false)
        .addComponent(jTextField2)
        .addComponent(jPassword)
        .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 98,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(jPanel2Layout.createSequentialGroup()
        .addGap(105, 105, 105)
        .addComponent(jBCancelar,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
        .addContainerGap(32, Short.MAX_VALUE))
    );
    jPanel2Layout.setVerticalGroup(

```

```

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addContainerGap()

        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jTextField1,
                javax.swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jLabel1))

        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addGroup(jPanel2Layout.createSequentialGroup()

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel2)

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 13,
                    Short.MAX_VALUE))
            .addGroup(jPanel2Layout.createSequentialGroup()

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jTextField2,
                    javax.swing.GroupLayout.PREFERRED_SIZE,

```

```

javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.BASELINE)
    .addComponent(jPassword,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel3,
javax.swing.GroupLayout.DEFAULT_SIZE, 19, Short.MAX_VALUE))
    .addGap(18, 18, 18)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignme
nt.BASELINE)
    .addComponent(jBCancelar)
    .addComponent(jBConectar,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addContainerGap()
);

javax.swing.GroupLayout mainPanelLayout = new
javax.swing.GroupLayout(mainPanel);
mainPanel.setLayout(mainPanelLayout);
mainPanelLayout.setHorizontalGroup(

```

```

mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(mainPanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addGap(157, 157, 157))
    );
mainPanelLayout.setVerticalGroup(

mainPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(mainPanelLayout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(19, Short.MAX_VALUE))
    );

statusPanel.setName("statusPanel"); // NOI18N

statusPanelSeparator.setName("statusPanelSeparator"); // NOI18N

statusMessageLabel.setName("statusMessageLabel"); // NOI18N

```

```

statusAnimationLabel.setHorizontalAlignment(javax.swing.SwingConstants.LEF
T);
    statusAnimationLabel.setName("statusAnimationLabel"); // NOI18N

    progressBar.setName("progressBar"); // NOI18N

    javax.swing.GroupLayout statusPanelLayout = new
javax.swing.GroupLayout(statusPanel);
    statusPanel.setLayout(statusPanelLayout);
    statusPanelLayout.setHorizontalGroup(

statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
        .addComponent(statusPanelSeparator,
javax.swing.GroupLayout.DEFAULT_SIZE, 265, Short.MAX_VALUE)
        .addGroup(statusPanelLayout.createSequentialGroup()
            .addComponent(statusMessageLabel)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 95,
Short.MAX_VALUE)
            .addComponent(progressBar,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(statusAnimationLabel)

```

```

        .addContainerGap()
    );
    statusPanelLayout.setVerticalGroup(

statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(statusPanelLayout.createSequentialGroup()
            .addComponent(statusPanelSeparator,
                javax.swing.GroupLayout.PREFERRED_SIZE, 2,
                javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

            .addGroup(statusPanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(statusMessageLabel)
                .addComponent(statusAnimationLabel)
                .addComponent(progressBar,
                    javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(3, 3, 3))
    );

    jProuterdatos.setName("jProuterdatos"); // NOI18N

    jScrollPane2.setName("jScrollPane2"); // NOI18N
    //Creamos una table para listar en pantalla

```



```

    }
) {
    Class[] types = new Class [] {
        java.lang.Integer.class, java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class, java.lang.String.class,
java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
jTdatosrouter.setName("jTdatosrouter"); // NOI18N
jScrollPane2.setViewportView(jTdatosrouter);

jTdatosrouter.getColumnModel().getColumn(0).setHeaderValue(resourceMap.get
String("jTdatosrouter.columnModel.title6")); // NOI18N

jTdatosrouter.getColumnModel().getColumn(1).setHeaderValue(resourceMap.get
String("jTdatosrouter.columnModel.title0")); // NOI18N

jTdatosrouter.getColumnModel().getColumn(2).setHeaderValue(resourceMap.get
String("jTdatosrouter.columnModel.title5")); // NOI18N

jTdatosrouter.getColumnModel().getColumn(3).setHeaderValue(resourceMap.get
String("jTdatosrouter.columnModel.title1")); // NOI18N

jTdatosrouter.getColumnModel().getColumn(4).setHeaderValue(resourceMap.get
String("jTdatosrouter.columnModel.title2")); // NOI18N

```

```

jTdatosrouter.getColumnModel().getColumn(5).setHeaderValue(resourceMap.get
String("jTdatosrouter.columnModel.title3")); // NOI18N

jTdatosrouter.getColumnModel().getColumn(6).setHeaderValue(resourceMap.get
String("jTdatosrouter.columnModel.title4")); // NOI18N

    label1.setFont(resourceMap.getFont("label1.font")); // NOI18N
    label1.setName("label1"); // NOI18N
    label1.setText(resourceMap.getString("label1.text")); // NOI18N

    javax.swing.GroupLayout jProuterdatosLayout = new
javax.swing.GroupLayout(jProuterdatos);
    jProuterdatos.setLayout(jProuterdatosLayout);
    jProuterdatosLayout.setHorizontalGroup(

jProuterdatosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LE
ADING)
        .addGroup(jProuterdatosLayout.createSequentialGroup()
            .addGroup(jProuterdatosLayout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.LEADING, false)
                .addGroup(jProuterdatosLayout.createSequentialGroup()
                    .addGap(41, 41, 41)

                .addGroup(jProuterdatosLayout.createParallelGroup(
                    javax.swing.GroupLayout.Alignment.LEADING, false)
                    .addGroup(jProuterdatosLayout.createSequentialGroup()
                        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jProuterdatosLayout.createSequentialGroup()
                            .addComponent(label1,
                                javax.swing.GroupLayout.PREFERRED_SIZE,
                                javax.swing.GroupLayout.DEFAULT_SIZE,
                                javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addGap(222, 222, 222))
        .addComponent(jScrollPane2,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 722,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(29, 29, 29))
    );
    jProuterdatosLayout.setVerticalGroup(

jProuterdatosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jProuterdatosLayout.createSequentialGroup()
            .addGap(16, 16, 16)
            .addComponent(label1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 381,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(109, 109, 109))
    );

    setComponent(mainPanel);
    setStatusBar(statusPanel);
} // </editor-fold>

private void jBConectarActionPerformed(java.awt.event.ActionEvent evt) {

```

```

// TODO add your handling code here:
//private TelnetClientJava clienteTelnet;
RouterManager routerManager=new RouterManager();
try {
    String respuesta=routerManager.conectar(jTextField1.getText(),
jTextField2.getText(), new String(this.jPassword.getPassword()));
    JOptionPane.showMessageDialog(null,respuesta);
    if(respuesta.indexOf("incorrecto")!=-1){
        System.exit(0);
    }
    PingBoxUsuarios nf=new PingBoxUsuarios(routerManager);
    nf.setVisible(true);
} catch (PingBoxException ex) {
    JOptionPane.showMessageDialog(null,
ex.getMessage(),"ERROR",JOptionPane.ERROR_MESSAGE);
}
}

private void jBCancelarActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
//this.clienteTelnet.disconnect();
}
//private TelnetClientJava clienteTelnet;
// Variables declaration - do not modify
private javax.swing.JButton jBCancelar;
private javax.swing.JButton jBConectar;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;

```

```

private javax.swing.JPanel jPanel2;
private javax.swing.JPasswordField jPassword;
private javax.swing.JPanel jProuterdatos;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JTable jTdatosrouter;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
private java.awt.Label label1;
private javax.swing.JPanel mainPanel;
private javax.swing.JProgressBar progressBar;
private javax.swing.JLabel statusAnimationLabel;
private javax.swing.JLabel statusMessageLabel;
private javax.swing.JPanel statusPanel;
// End of variables declaration

private final Timer messageTimer;
private final Timer busyIconTimer;
private final Icon idleIcon;
private final Icon[] busyIcons = new Icon[15];
private int busyIconIndex = 0;

private JDialog aboutBox;
}

```

Gráfico III.11 Ingreso textbox y asignación de parámetros

3.3.3.2.6 Código Fuente Conexión RouterOS

3.3.3.2.6.1 Librería Commons-Net

La librería para **java** de **Apache Commons Net**, implementa de lado del cliente muchos protocolos básicos IP (telnet, ftp, tftp, smtp ...).

Para el IDE de desarrollo NetBeans, se incluye esta librería en el proyecto pingbox.

Procedimiento:

Descarga de la librería

http://commons.apache.org/net/download_net.cgi

<http://apache.rediris.es//commons/net/binaries/commons-net-2.2-bin.zip>

Cargar librería global:

Descomprimo **commons-net-2.2.jar** en la carpeta de un proyecto de NetBeans o, mejor, en una carpeta de NetBeans para librerías añadidas

Abrimos el proyecto de NetBeans

Menú Herramientas > Bibliotecas > Nueva Biblioteca > (definimos el nombre de la biblioteca) ApacheCommonsNet > aceptar.

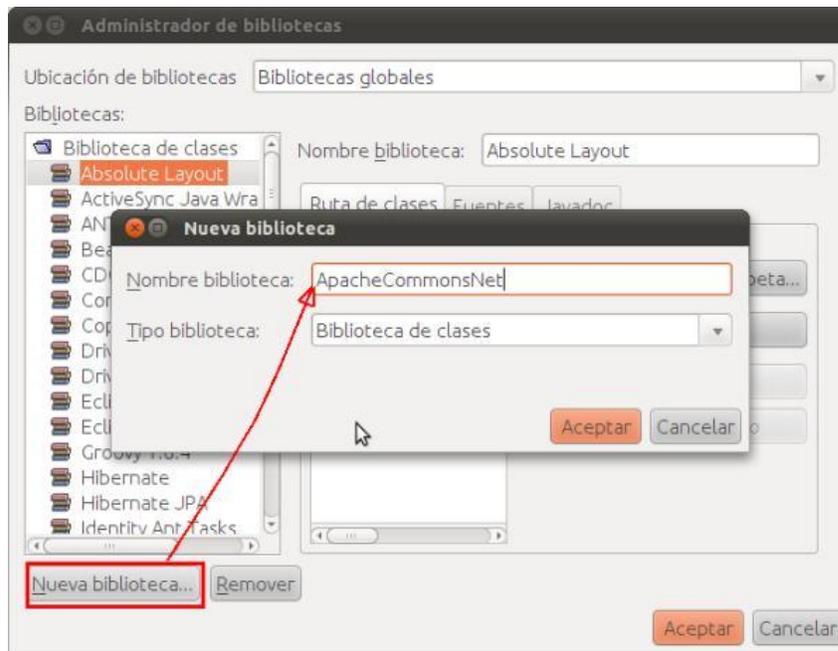


Gráfico III.12 Carga del proyecto

Agregar Archivo Jar/Carpeta > indicamos el path (camino) a commons-net-2.2.jar

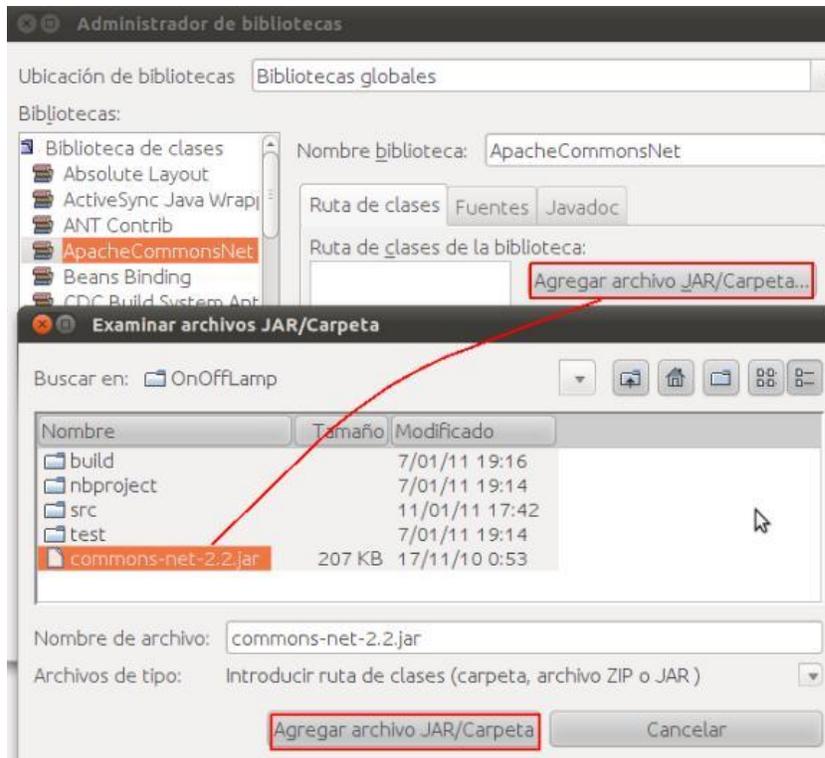


Gráfico III.13 Incorporación del archivo

Agregar librería al proyecto:

Menú Archivo > Proyecto Properties (propiedades) (Nombre de tu proyecto) > Bibliotecas > Añadir biblioteca > Elegir la biblioteca global cargada anteriormente (ApacheCommonsNet) > añadir biblioteca > aceptar

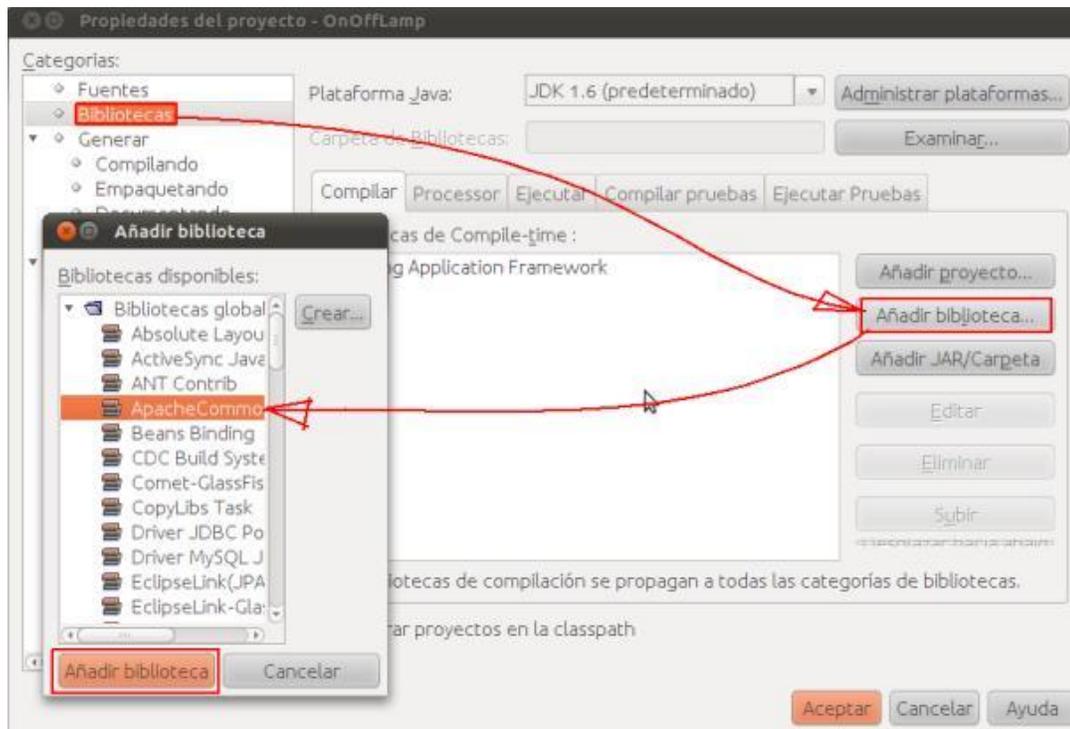


Gráfico III.14 Añadidura de librerías.

Ya se puede acceder a las librerías org.apache.commons.*

Por ejemplo:

```
import org.apache.commons.net.telnet.telnetclient
```

3.3.3.2.6.2 Acceso de logueo a router mediante puerto telnet

Para acceder al sistema RouterOS necesariamente tendremos que interpretar la ejecución de comandos por medio del puerto “Telnet” por lo que se crea un acceso agregando las siguientes librerías y ejecutando algunas líneas de comandos directos,

la clase principal el archivo se llama “RouterManager” el cual analizamos en este apartado.

3.3.3.2.6.3 Librerías fundamentales utilizadas en este proyecto

El paquete que el API estándar de Java nos proporciona para gestionar las operaciones de I/O tanto del sistema como desde/a ficheros. Como sabemos, Java es un lenguaje orientado a red, cuyo potencial radica en todo aquello que tenga que ver con la ejecución de aplicaciones a través de la red y, por tanto, este paquete se ajusta perfectamente a esta condición permitiendo la gestión de ficheros desde sitios remotos. Evidentemente, Java va mucho más allá del típico `println()`, cuyo uso más normal y natural es el de hacer los "debugging" de la aplicación que estemos creando.

java.io

Para la entrada y salida a través de flujos de datos, y ficheros del sistema. Dentro de esta librería se encuentran algunas funciones utilizadas en este proyecto.

- **java.io.BufferedReader**
Permite leer en modo texto (como con `System.in`)
- **java.io.IOException**
Esta clase es la clase general de excepciones producida por operaciones que han fallado o han sido interrumpidas.
- **java.io.InputStream**
Esta clase abstracta es la superclase de todas las clases que representan un flujo de entrada de bytes.
- **java.io.InputStreamReader**
La clase le permite convertir una cadena normal en un lector.
- **java.io.PrintStream**
Esta clase le permite escribir los datos con formato en un `OutputStream` subyacente. Por ejemplo, la escritura de `writing int`, `long` y otros formatos de texto, en lugar de sus valores de byte.

java.net

Librería que apoya la interface con telnet.

- **java.net.SocketException**

Utilizada para realizar la conexión con aplicaciones en red, o aplicaciones con interfaces de red.

java.util

Clases como diccionarios, tablas de hash, stack, técnicas de codificación, y decodificación, hora, fecha, etc.

- **java.util.ArrayList**

Esta clase permite contener y ordenar objetos, incluso, puede almacenar objetos duplicados. Su tamaño es dinámico, es decir, esta lista crecerá a medida que se inserten en ella más elementos.

- **java.util.List**

Una colección ordenada (también conocido como una secuencia). El usuario de este interfaz tiene un control preciso sobre el lugar donde en la lista de cada elemento se inserta. El usuario puede acceder a los elementos por su índice de número entero (posición en la lista), y la búsqueda de elementos de la lista.

- **java.util.logging.Level**

La clase de nivel define un conjunto estándar de los niveles de registro que se puede utilizar para controlar el registro de salida.

- **java.util.logging.Logger**

Un objeto Logger se utiliza para registrar los mensajes de un determinado sistema o componente de aplicación.

- **org.apache.commons.net.telnet.TelnetClient**

Biblioteca implementa el lado del cliente de muchos protocolos básicos de Internet. El propósito de la biblioteca es facilitar el acceso de protocolo fundamental, no abstracciones de alto nivel. Por lo tanto, algunos de

los diseños orientados a objetos violan los principios de diseño. Esta librería se aplica en la utilización de cliente “Telnet”.

pingbox.Usuario

Clase importada, es la parte grafica de la aplicación del proyecto.

```
/*
 * Autenticación de RouterOS
 * a través del puerto 23 de telnet
 */
package pingbox;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.PrintStream;
import java.net.SocketException;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import org.apache.commons.net.telnet.TelnetClient;
import pingbox.Usuario;

/**
 * @author Chancusig Omar && Martinez Luis
 */
```

Gráfico III. 15 Importación de clase

Ahora creamos la clase “RouterManager”, la cual utilizaremos para loguearnos para aplicar el control de balanceo de carga en el sistema RouterOS de Mikrotik.

```
public class RouterManager
{

    private TelnetClient telnet;
    private InputStream in;
    private PrintStream out;
    private BufferedReader reader;

    public RouterManager()
    {
        telnet = new TelnetClient();
    }

    public String conectar(String servidor, String usuario, String password)
    throws PingBoxException
    {
        try {

            //Se abre la conexión al telnet por el puerto 23

            telnet.connect(servidor, 23);

            //Ahora se necesita una forma de leer las respuestas que
            //envía el telnet, para esto se obtiene un InputStream
            //del objeto telnet

            in = telnet.getInputStream();
```

```

//Ahora se necesita una forma de enviarle los comandos al
    //telnet para esto se utiliza un OutputStream desde
//el objeto telnet

out = new PrintStream(telnet.getOutputStream());

    //Ahora se incluye el InputStream dentro de un
    // BufferedReader para que la lectura de las respuestas del
    // “Telnet” sean mucho más sencillas y mejor gestionadas

reader = new BufferedReader(new InputStreamReader(in));

    //Ahora se lee de la consola a través del método
    //readUntil el cual lee de la consola hasta que el último
    //caracter (un char) sea -1 y que el prompt sea igual
    //al patrón que le enviamos como argumento, en este caso
    //es hasta que el prompt despliegue el patron login:

leerLoginPassword("Login: ");

    //cuando el readUntil de login finaliza, se procede a
    // ingresar el user a través del método write, el cual
    // escribe en la consola

enviarComando(usuario);

    //esperamos hasta que el prompt muestre la palabra password:
    //La palabra tiene que ser exacta a la que sale en el prompt

```

```

String respuesta=leerLoginPassword("Password: ").toString();
if(respuesta.equalsIgnoreCase("login o password incorrecto"))
    {
        return respuesta;
    }

    //Ahora ingresamos el password

enviarComando(password);

    //Esperamos hasta que salga el prompt de nuestro servidor

respuesta=leerLoginPassword("[admin@MikroTik] >").toString();

    //Si el usuario o password es incorrecto aplicamos esta
    //regla

if(respuesta.equalsIgnoreCase("login o password incorrecto"))
    {
        return respuesta;
    }

    //Si la conexión es exitosa nos devolverá este mensaje

return new String("Conexión exitosa");

    }
    catch (Exception ex)

```

```

        {
            throw new PingBoxException("Error al conectarse al router");
        }
    }

public void enviarComando(String value) {
    try {
        out.println(value);
        out.flush();
        System.out.println(value);
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public StringBuffer leerLoginPassword(String pattern) {
    StringBuffer sb = new StringBuffer();
    try {
        char lastChar = pattern.charAt(pattern.length() - 1);
        int check = in.read();
        char ch = (char) check;
        while (check != -1) {
            System.out.print(ch);
            sb.append(ch);
            if(sb.toString().indexOf("incorrect username or password")!=-1){
                return new StringBuffer("login o password incorrecto");
            }
            if (ch == lastChar) {

```

```

        if (sb.toString().endsWith(pattern)) {

            return sb;
        }
    }
    check = in.read();
    ch = (char) check;
}
} catch (Exception e) {
e.printStackTrace();
}
return sb;
}

```

Gráfico III.16 Creación de clase

Una vez ingresado al RouterOS mediante “Telnet”, procedemos a extraer la información interna, los datos a importar son los de las Colas Simples (queues simple)

```

//Dentro de esta clase se ingresa hasta el directorio de
//Mikrotik y directamente imprimiendo los datos “queue
//simple print”

public StringBuffer leerUsuarios()
{
    enviarComando("/queue simple print");
    StringBuffer sbUsuarios = leerPrint(new StringBuffer());
return sbUsuarios;
}

```

```

//Esta clase se almacenara en un buffer la información
//obtenida del directorio de colas simples de mikrotik

public StringBuffer leerPrint(StringBuffer sb) {
    //StringBuffer sb = new StringBuffer();
    try {
        char lastChar = '>';
        int contadorLastChar = 0;
        int check = in.read();
        char ch = (char) check;
        while (check != -1) {

            //a pesar de terminar la lectura nunca toma el valor -1

            System.out.print(ch);
            sb.append(ch);

            if (sb.indexOf("dump") > -1) {
                System.out.println("LLEGA dump");
                enviarComando("\n");

                //envio enter para que me siga devolviendo informacion

                sb.delete(sb.length() - 4, sb.length());

                //borro la palabra dump

                leerPrint(sb);
            }
        }
    }
}

```

```

return sb;
    }
if (ch == lastChar) {
    contadorLastChar++;
    if (contadorLastChar == 3) {
        return sb;
    }
}
check = in.read();
ch = (char) check;
}
} catch (Exception e) {
    e.printStackTrace();
}
return sb;
}

```

Gráfico III.17 Extracción de la información interna

3.3.3.2.6.4 Clasificación y separación de datos

Ahora con los datos totales de las colas simples clasificaremos cada uno de ellos extraídos de las colas simples, este se encuentra en un solo archivo por lo que haremos una búsqueda del nombre, y del numero para clasificar cada uno de los parámetros que se obtuvo de los datos de las Colas Simples (queue simple), se utilizara un array para clasificar los mismos.

```

//Creamos una clase llamada "List"
//donde comenzaremos a separar los usuarios

public List<Usuario> obtenerUsuarios() {

```

```

        return separarUsuarios();
    }

    public List<Usuario> separarUsuarios() {

        Usuario usuario = null;
        List<Usuario> usuarios = new ArrayList<Usuario>();
        StringBuffer sbU = leerUsuarios();
        String stringUsuarios = new String(sbU);
        String arregloPartes[] = stringUsuarios.split("=");
        for (int i = 0; i < arregloPartes.length; i++) {
            if (contiene(arregloPartes[i], "name")) {
                int numero = encontrarNumero(arregloPartes[i].getBytes());
                System.out.println("NUMERO: " + numero);
                usuario = new Usuario();
                usuario.setNumero(numero);
            } else {
                String parametro = recuperarParametro(arregloPartes[i].substring(3));
                System.out.println("PARAMETRO:" + parametro);
                asignarParametro(parametro, arregloPartes[i - 1], usuario, usuarios);
            }
        }

        return usuarios;
    }

    //Asignaremos los parámetros que vamos a extraer
    //de las colas simples para luego almacenarlas
    //en un array

```

```

private void asignarParametro(String parametro, String cadenaAnterior,
    Usuario usuario, List<Usuario> usuarios) {
if (contiene(cadenaAnterior, "name")) {
    usuario.setName(parametro);
} else if (contiene(cadenaAnterior, "target-addresses")) {
    usuario.setTargetAdresses(parametro);
} else if (contiene(cadenaAnterior, "dst-address")) {
    usuario.setDstAdress(parametro);
} else if (contiene(cadenaAnterior, "interface")) {
    usuario.setInterfaceU(parametro);
} else if (contiene(cadenaAnterior, "parent")) {
    usuario.setParent(parametro);
} else if (contiene(cadenaAnterior, "direction")) {
    usuario.setDirection(parametro);
} else if (contiene(cadenaAnterior, "priority")) {
    usuario.setPriority(parametro);
} else if (contiene(cadenaAnterior, "queue")) {
    usuario.setQueue(parametro);
} else if (contiene(cadenaAnterior, "burst-limit")) {
    usuario.setBurstLimit(parametro);
} else if (contiene(cadenaAnterior, "burst-threshold")) {
    usuario.setBurstThreshold(parametro);
} else if (contiene(cadenaAnterior, "max-limit")) {
    usuario.setMaxLimit(parametro);
} else if (contiene(cadenaAnterior, "limit-at")) {
    usuario.setLimitAt(parametro);
} else if (contiene(cadenaAnterior, "burst-time")) {
usuario.setLimitAt(parametro);

```

```

        usuarios.add(usuario);
    }
}

//Creamos la clase para recuperar la información que se va a clasificar

public String recuperarParametro(String cadena) {

    //para el comando "name" de las colas simples

    if (cadena.indexOf("\"") > -1) {
        cadena = cadena.substring(1,cadena.length());
    }
    byte[] cadenaBytes = cadena.getBytes();
    int posicionFinal = 0;
    //para el name

    if (cadena.indexOf("\"") > -1) {
        cadena = cadena.substring(0,cadena.length());
        posicionFinal=cadena.indexOf("\"");
    }
    else{
        for (int i = 0; i < cadenaBytes.length; i++) {
            if (cadenaBytes[i] == 32) {
                posicionFinal = i;
                break;
            }
        }
    }
}

```

```

return cadena.substring(0, posicionFinal);
}

//Aquí buscaremos por números cada uno de los valores asignados
//a las colas simples

public int encontrarNumero(byte[] cadena) {
    String numeroStr = "";
    int primerEspacio = posicionPrimerEspacio(cadena);
    int posicionFinalNumero = posicionDigitoFinal(cadena, primerEspacio);
    if (cadena[posicionFinalNumero - 1] != 32) {
        numeroStr = "" + (char) cadena[posicionFinalNumero - 1] + (char)
cadena[posicionFinalNumero];
    } else {
        numeroStr = "" + (char) cadena[posicionFinalNumero];
    }
    return Integer.parseInt(numeroStr);
}

//retorna la posición del primer espacio en blanco analizando
//de atrás hacia adelante
//si no encuentra espacio, retorna -1

public int posicionPrimerEspacio(byte[] cadena) {
for (int i = cadena.length - 1; i > 0; i--) {
    if (cadena[i] == 32) {
        return i;
    }
}
}

```

```

    return -1;
}

//encuentra el primer digito buscando de
//atrás hacia adelante, desde el "primerEspacio"

public int posicionDigitoFinal(byte[] cadena, int posicionPrimerEspacio) {
for (int i = posicionPrimerEspacio; i > 0; i--) {
    if (cadena[i] != 32) {
        return i;
    }
}
return -1;
}

public boolean contiene(String cadena, String clave) {
int posicion = cadena.indexOf(clave);
if (posicion == -1) {
    return false;
} else {
return true;
}
}
}

```

Gráfico III.18 Clasificación de datos.

Ahora con todos la información del dato seleccionado procederemos a realizar la actualización del ancho de banda requerido aplicando un balance en la limitación, esto nos generara una calidad de servicio, además agregaremos un parámetro de calidad de servicio al momento de actualizar la información de las colas simples.

Lo principal es extraer este tipo de información, realizar el cambio, eliminar dicha información y reingresarla con los datos actualizados a las colas simples.

```
public void eliminarRegistro(Integer registro) {
    enviarComando("que sim remove "+registro);
    leerEliminar();
}

public StringBuffer leerEliminar() {
    StringBuffer sb = new StringBuffer();
    try {
        char lastChar = '>';
        int contadorLastChar = 0;
        int check = in.read();
        char ch = (char) check;
        while (check != -1) { //a pesar de terminar la lectura nunca toma el valor -1
            System.out.print(ch);

            sb.append(ch);
            if (ch == lastChar) {
                contadorLastChar++;
                System.out.println("CONTADOR LAST CHAR:" +
                contadorLastChar);
                if (contadorLastChar == 4) {
                    return sb;
                }
            }
            check = in.read();
            ch = (char) check;
        }
    }
}
```

```

    } catch (Exception e) {
e.printStackTrace();
}
return sb;
}

```

Gráfico III.19 Cambio y reingreso de datos

Ingresaremos la información enviando directamente el comando de agregar hacia la consola interna “Telnet”, para de esta manera terminar la actualización de los datos

```

public void agregarRegistro(Usuario usuario) {
int numSignos=6;
    String cadena="que simple add name=\""+usuario.getName()
        +"\\" max-limit="+usuario.getMaxLimit()+
        " interface="+usuario.getInterfaceU();
    if(usuario.getTargetAdresses()!=null){
cadena=cadena+" target-addresses="+usuario.getTargetAdresses();
        numSignos=4;
    }
    enviarComando(cadena);
leerAgregar(numSignos);
}

public StringBuffer leerAgregar(int numSignos) {
    StringBuffer sb = new StringBuffer();
    try {
        char lastChar = '>';
        int contadorLastChar = 0;
        int check = in.read();
char ch = (char) check;

```

```

        while (check != -1) { //a pesar de terminar la lectura nunca toma el valor -1
System.out.print(ch);

        sb.append(ch);
        if (ch == lastChar) {
            contadorLastChar++;
            System.out.println("AGREGAR CONTADOR LAST CHAR:" +
contadorLastChar);
            // if (contadorLastChar == numSignos) {
            if (contadorLastChar == 2) {
                return sb;
            }
        }
        check = in.read();
        ch = (char) check;
    }
} catch (Exception e) {
e.printStackTrace();
}

return sb;

}
}

```

Gráfico III.20 Actualización de datos

3.3.3.2.7 Interface de modificación de usuarios

Analizaremos la estructura de programación de cada uno de las herramientas en este formulario e donde se encuentran los datos de las limitaciones de ancho de banda

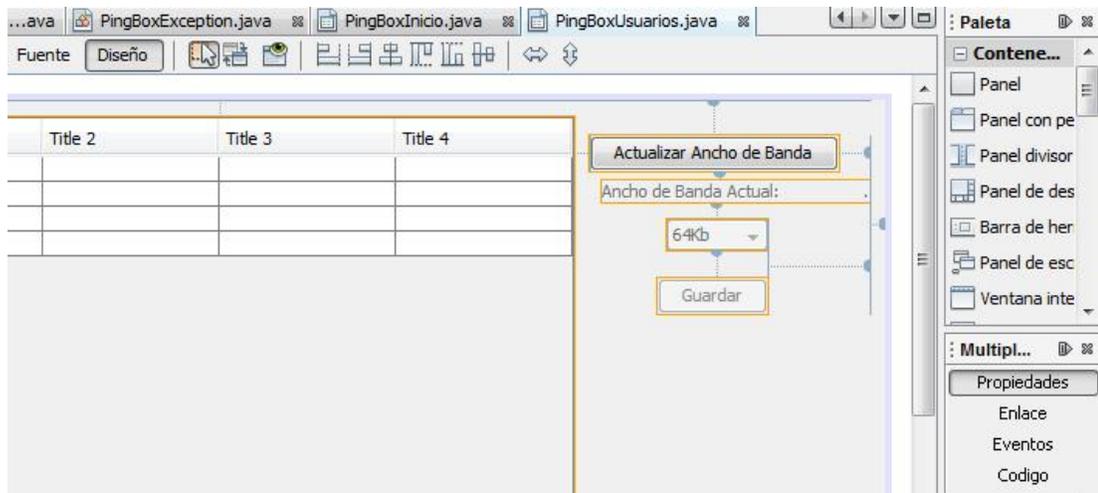


Gráfico III.21 Datos de las limitaciones

```

/*
 * PingBoxUsuarios.java
 *
 * Created on 05/01/2005, 11:35:59 AM
 */
package pingbox;

import java.util.List;
import java.util.Vector;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import javax.swing.table.TableModel;

public class PingBoxUsuarios extends javax.swing.JFrame {

    private RouterManager routerManager;
    List<Usuario> usuarios;

```

```

/** Creates new form PingBoxUsuarios */
public PingBoxUsuarios(RouterManager rm) {
routerManager = rm;
    usuarios = routerManager.obtenerUsuarios();

    initComponents();
    pintarTabla();
}

public void pintarTabla() {
    int i = 1;
Vector vectorNombres = new Vector();
    vectorNombres.add("#");
    vectorNombres.add("Usuario");
    vectorNombres.add("Direccion IP");
    vectorNombres.add("Ancho de Banda");

Vector vectorDatos = new Vector();
Vector fila;
for (Usuario usuario : usuarios) {
    fila = new Vector();
    fila.add(usuario.getNumero());
    fila.add(usuario.getName());
    fila.add(usuario.getTargetAdresses());
    fila.add(usuario.getMaxLimit());
    vectorDatos.add(fila);
}
}

```

```

        TableModel tableModel = new DefaultTableModel(vectorDatos,
vectorNombres);
jTable1.setModel(tableModel);
    }

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jScrollPane1 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jButton1 = new javax.swing.JButton();
    lblAnchoActual = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    btnGuardar = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    org.jdesktop.application.ResourceMap resourceMap =
org.jdesktop.application.Application.getInstance(pingbox.PingboxApp.class).getC
ontext().getResourceMap(PingBoxUsuarios.class);
    setTitle(resourceMap.getString("Form.title")); // NOI18N
    setBounds(new java.awt.Rectangle(320, 200, 450, 350));
    setName("Form"); // NOI18N

```

```

jScrollPane1.setName("jScrollPane1"); // NOI18N

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "Title 1", "Title 2", "Title 3", "Title 4"
    }
));
jTable1.setName("jTable1"); // NOI18N
jScrollPane1.setViewportViewView(jTable1);

jTable1.getColumnModel().getColumn(0).setHeaderValue(resourceMap.getString(
("jTable1.columnModel.title0"))); // NOI18N

jTable1.getColumnModel().getColumn(1).setHeaderValue(resourceMap.getString(
("jTable1.columnModel.title1"))); // NOI18N

jTable1.getColumnModel().getColumn(2).setHeaderValue(resourceMap.getString(
("jTable1.columnModel.title2"))); // NOI18N

jTable1.getColumnModel().getColumn(3).setHeaderValue(resourceMap.getString(
("jTable1.columnModel.title3"))); // NOI18N

```

```

jButton1.setText(resourceMap.getString("jButton1.text")); // NOI18N
jButton1.setName("jButton1"); // NOI18N
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});

lblAnchoActual.setText(resourceMap.getString("lblAnchoActual.text")); //
NOI18N
lblAnchoActual.setEnabled(false);
lblAnchoActual.setName("lblAnchoActual"); // NOI18N

jComboBox1.setModel(new javax.swing.DefaultComboBoxModel(new
String[] { "64Kb", "128Kb", "256Kb", "512Kb", "1M", "2M", "Ilimitado" }));
jComboBox1.setEnabled(false);
jComboBox1.setName("jComboBox1"); // NOI18N

btnGuardar.setText(resourceMap.getString("btnGuardar.text")); // NOI18N
btnGuardar.setEnabled(false);
btnGuardar.setName("btnGuardar"); // NOI18N
btnGuardar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnGuardarActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());

```

```

        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 458, Short.MAX_VALUE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 10,
Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
        .addComponent(lblAnchoActual,
javax.swing.GroupLayout.Alignment.TRAILING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
        .addComponent(jButton1)
        .addGap(20, 20, 20))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRA
ILING)
        .addComponent(btnGuardar)

```

```

        .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(66, 66, 66))
        .addContainerGap()
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)

        .addGroup(layout.createSequentialGroup()
            .addGap(23, 23, 23)
            .addComponent(jButton1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(lblAnchoActual)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(btnGuardar))
        .addGroup(layout.createSequentialGroup()

```

```

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 504, Short.MAX_VALUE)))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    pack();
} // </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    int fila = jTable1.getSelectedRow();
    if (fila >= 0) {
        String anchoActual = (String) jTable1.getValueAt(fila, 3);

        lblAnchoActual.setText("Ancho de banda Actual:"+anchoActual);
        lblAnchoActual.setEnabled(true);
        jComboBox1.setEnabled(true);
        btnGuardar.setEnabled(true);
    } else {
        JOptionPane.showMessageDialog(null, "Debe seleccionar el usuario que
desea modificar","ERROR",JOptionPane.ERROR_MESSAGE);
    }
}

private void btnGuardarActionPerformed(java.awt.event.ActionEvent evt) {
    int fila = jTable1.getSelectedRow();

```

```

        if (fila >= 0) {
Integer valorSeleccionado = (Integer) jTable1.getValueAt(fila, 0);
        System.out.println("Numero registro" + valorSeleccionado);
        routerManager.eliminarRegistro(valorSeleccionado);
        Usuario usuarioSeleccionado = buscar(valorSeleccionado);
        int indiceSeleccionado = jComboBox1.getSelectedIndex();
        String ancho = anchoBanda(indiceSeleccionado);
        usuarioSeleccionado.setMaxLimit(ancho);
        routerManager.agregarRegistro(usuarioSeleccionado);
        JOptionPane.showMessageDialog(null, "Usuario modificado
exitosamente");
        System.exit(0);
        usuarios = routerManager.obtenerUsuarios();
        pintarTabla();
lblAnchoActual.setEnabled(false);
        jComboBox1.setEnabled(false);
        btnGuardar.setEnabled(false);

        }
// TODO add your handling code here:
}

public String anchoBanda(int indice) {
    if (indice == 0) {
        return "64000/64000";
    } else if (indice == 1) {
        return "128000/128000";
    } else if (indice == 2) {
        return "256000/256000";
    }
}

```

```

    } else if (indice == 3) {
        return "512000/512000";
    } else if (indice == 4) {
        return "1000000/1000000";
    } else if (indice == 5) {
        return "2000000/2000000";
    } else if (indice == 6) {
        return "0/0";
    }
    return "0/0";
}

public Usuario buscar(int numero) {
    for (Usuario usuario : usuarios) {
        if (usuario.getNumero() == numero) {
return usuario;
        }
    }
    return null;
}

// Variables declaration - do not modify
private javax.swing.JButton btnGuardar;
private javax.swing.JButton jButton1;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
private javax.swing.JLabel lblAnchoActual;
// End of variables declaration
}

```

Gráfico III.22 Asignaciones

3.3.4 Fase de instalación

3.3.4.1 Instalación y configuración de sistema Ping Box v1.1

3.3.4.1.1 Requerimientos de hardware y software

3.3.4.1.1.1 Requisitos y recomendaciones de hardware

Los siguientes requisitos de hardware proporcionan un punto de partida mínimo para la selección de hardware compatible con su instalación de Ping Box.

	Requerido	Recomendado
Procesador(es)	Intel Pentium III	Procesadores de doble núcleo o superior
Velocidad del procesador	800 MHz o más	2 GHz o superior
RAM	128 MB	1 GB o superior
Espacio en disco duro para aplicación y plataforma Java.	700 Mbps	1 GB o superior
Tarjeta de interfaz de red	100 Mbps	1 Gbps o superior
CD-ROM o DVD-ROM	Requerido si se instala desde CD-ROM. No requerido para instalaciones realizadas desde una descarga.	Requerido si se instala desde CD-ROM. No requerido para instalaciones realizadas desde una descarga.
Resolución de pantalla de vídeo	1280 x 1024 o superior	1024 x 768
Tarjeta de sonido	No se requiere	No se requiere
Módem y línea telefónica	No se requiere	No se requiere

Módem GSM	No se requiere	No se requiere
-----------	----------------	----------------

Tabla III.1 Requisitos y recomendaciones de hardware

3.3.4.1.1.2 Requisitos y recomendaciones de software

Los siguientes requisitos de software proporcionan un punto de partida mínimo para la selección de software compatible con su instalación de Ping Box.

	Requerido	Recomendado
Sistema Operativo	Windows XP Profesional, Ubuntu 9.10, Fedora 14	Windows 7 x86 32 bits, Ubuntu 11.10, Fedora 16
Máquina virtual Java	Java(TM)6	Java(TM)6

Tabla III.2 Requisitos y recomendaciones de software

3.3.4.1.2 Proceso de Instalación

3.3.4.1.2.1 Arranque de CD-ROM

Primeramente debe asegurarse que el disco de instalación y el lector de discos CD-ROM se encuentren en buen estado, esto es necesario para que se ejecute el "instalador" de instalación Ping Box y posteriormente no se tenga problemas en la instalación en su disco duro.

NOTA: Los paquetes adicionales están dentro del disco de instalación que son requerimientos específicos al momento de ejecutar la aplicación, sin embargo para la instalación de la máquina virtual java es necesario tener una conexión a internet.

Instalación Requerimientos: Los requerimientos son esenciales para el funcionamiento de la aplicación Pingbox, cada uno de estos deben estar ejecutados antes inicializar la aplicación.

3.3.4.1.2.2 Instalación Java (JM) 6

Primeramente debemos saber si tenemos ya instalada java, para esto abrimos una consola de comandos "cmd", damos clic en INICIO, TODOS LOS PROGRAMAS, ACCESORIOS, luego ejecutamos "Símbolo del Sistema".

Ya en la consola de Windows tecleamos "java -version", nos devolverá información sobre la versión de java instalado.



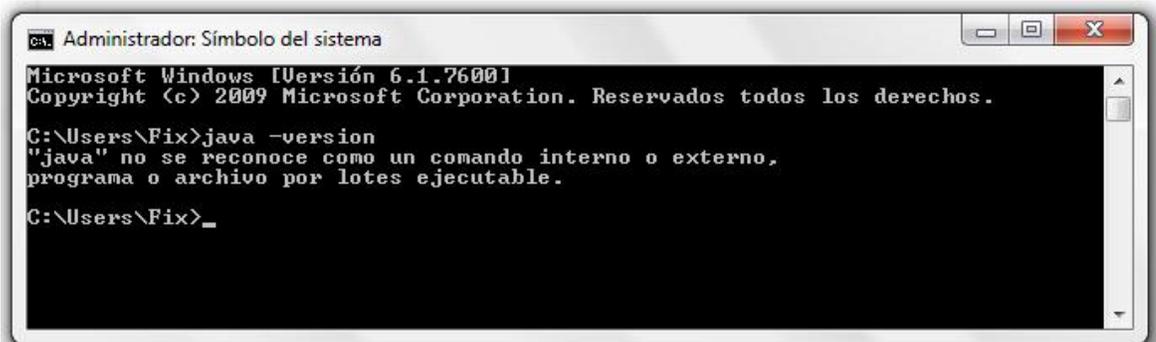
```
ca. Administrador: Símbolo del sistema
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Fix>java -version
java version "1.7.0_01"
Java(TM) SE Runtime Environment (build 1.7.0_01-b08)
Java HotSpot(TM) Client UM (build 21.1-b02, mixed mode, sharing)

C:\Users\Fix>
```

Gráfico III.23 Versión java en Windows

Si no tenemos ninguna versión de java instalado nos devolverá un mensaje de error, y procedemos a instalarlo.



```
ca. Administrador: Símbolo del sistema
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Fix>java -version
"java" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\Fix>_
```

Gráfico III.24 Impresión de Java no instalado

Dependiendo del sistema operativo de su equipo, seleccione un archivo de instalación que está dentro de la carpeta de cd de instalación.

Accedemos hasta el cd de instalación, seleccionamos la carpeta “Java (JM) 6”, según la distribución procederemos con la instalación.

Para distribuciones Microsoft Windows

Seleccionamos el archivo “jre-6u30-windows-i586-s” y lo ejecutamos.

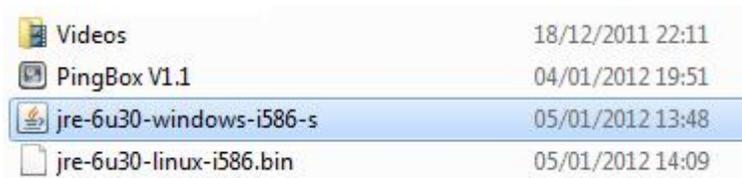


Gráfico III.25 Selección y ejecución de archivo jre

Dependiendo de la velocidad de la CPU tardara un poco en ejecutarse, se mostrara la siguiente ventana;



Gráfico III.26 Aceptación de la licencia

Damos un click en Instalar>, el proceso de instalación iniciara,



Gráfico III.27 Instalación de Java

Al finalizar la instalación nos mostrara una ventana avisando que la instalación termino.



Gráfico III.27 Finalización de la Instalación

Listo ya tenemos instalada nuestra máquina virtual de Java.

3.3.4.1.2.3 Instalación PINGBOX

Luego de haber verificado nuestros requerimiento pondremos en marcha la instalación de la aplicación pingbox, accedemos hasta nuestro disco de instalación y seleccionamos el archivo “PingBox V1.1” y lo ejecutamos

Nombre	Fecha de modifica...
 PingBox V1.1	04/01/2012 19:51
 jre-6u30-windows-i586-s	05/01/2012 13:48
 jre-6u30-linux-i586.bin	05/01/2012 14:09

Gráfico III.28 Selección y ejecución PingBox v1.1

La primera ventana es la de bienvenida a la instalación de la aplicación, y cuáles son los beneficios de utilizar la misma,

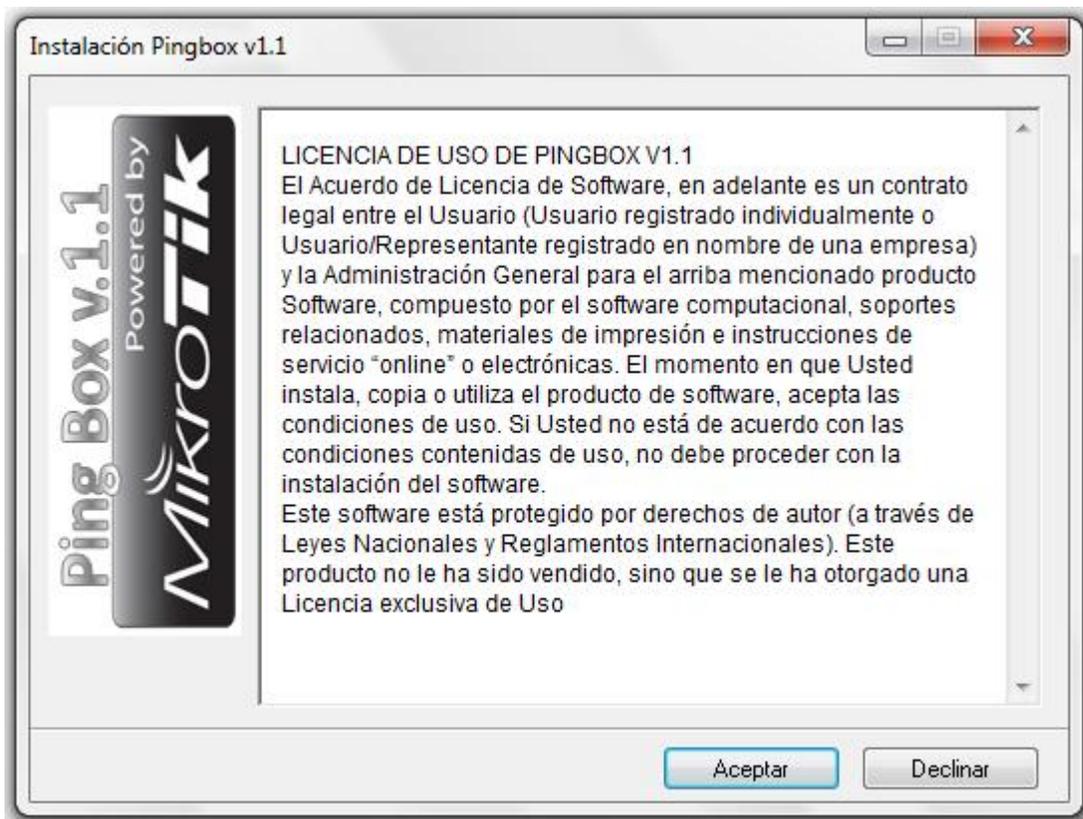


Gráfico III.29 Acuerdo de licencia

Damos clic en “Aceptar”, en la siguiente ventana podemos destinar una dirección diferente de instalación, lo más recomendable es mantener la misma dirección por defecto, luego damos clic en instalar.

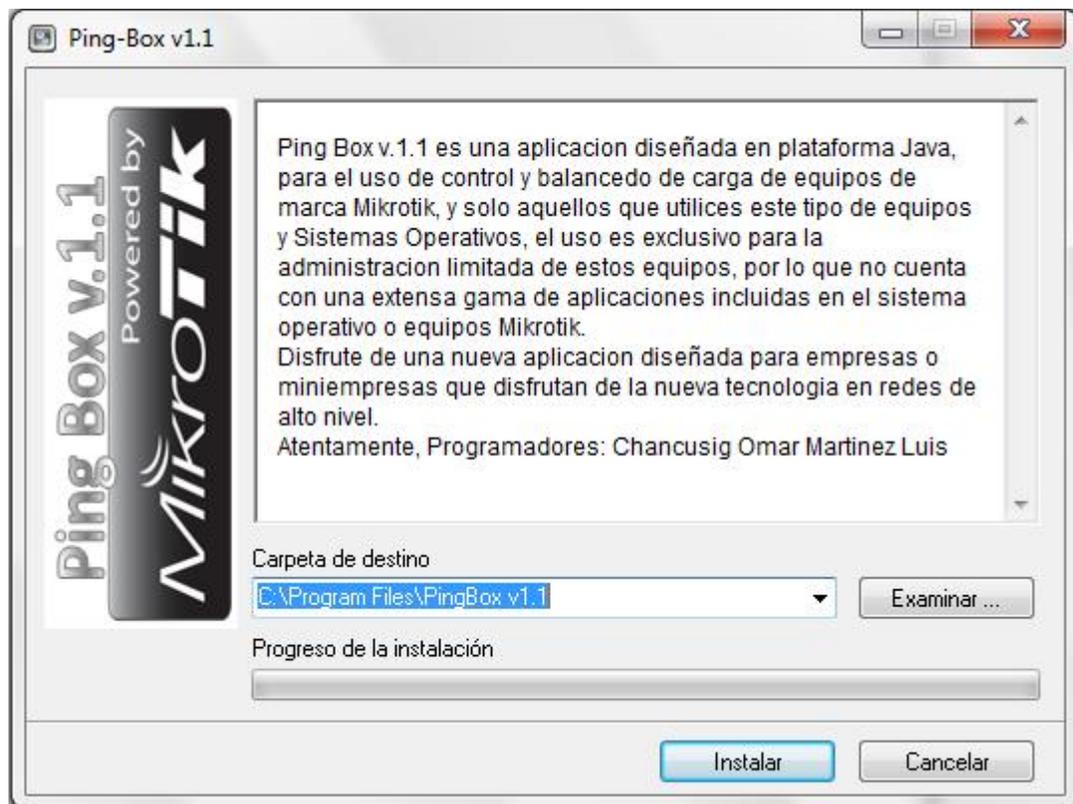


Gráfico III.30 Instalación del sistema

El proceso de instalación comenzara e instantáneamente luego de terminar la ventana automáticamente se cerrara, se creara un acceso directo en el escritorio y en Todos Los programas.

3.3.4.1.2.3.1 Manual de uso de aplicación PingBox-MK v1.0

3.3.4.1.2.3.1.1 Utilización del sistema PingBox-MK v1.0

El sistema está diseñado de manera que el usuario de RouterOS o administrador de Mikrotik pueda utilizar el sistema sin mayor problema.

3.3.4.1.2.3.1.2 Acceso de usuario

Para acceder al sistema RouterOS se necesita la dirección IP que sea alcanzable dentro de la red y perteneciente al cualquiera de las interfaces de Mikrotik, a continuación se detalla las características de la ventana de acceso mediante login.



Gráfico III.31 Acceso al sistema

- Ip Address, Dirección IP alcanzable desde equipo remoto hasta RouterOS Mikrotik, se puede utilizar la dirección IP del equipo asociado; para esto el equipo debe estar configurado dentro de la misma red IP o rutas alcanzables hacia este equipo.
- Usuario, Nombre de usuario dentro del sistema RouterOS, el usuario debe tener permisos especiales para realizar modificaciones en el sistema.
- Password, Contraseña de acceso a RouterOS, va en relación del usuario.
- Conectar, Luego de completar los campos este botón nos dará acceso hasta el sistema RouterOS e importara los datos de las colas simples.
- Cancelar, Cancela el acceso antes de ingresar al sistema RouterOS (cierra ventana Ping Box V.1.0)

3.3.4.1.2.3.1.3 Error de acceso

El sistema esta diseña para detectar errores de acceso como:

- Error de dirección IP, si la dirección IP de destino de cualquier interface de RouterOS no es alcanzable, el sistema arrojara un mensaje de error, o bien si la dirección IP no pertenece a un sistema Mikrotik.
- Error de Usuario, dentro del sistema RouterOS están almacenados todos los usuarios con permisos, si el sistema Ping Box no encuentra ningún usuario con el ingresado no se tendrá acceso al sistema RouterOS de Mikrotik
- Error de Password, naturalmente si la contraseña es errónea o bien se digitó de manera equivocada impedirá el acceso al sistema RouterOS enviándonos en pantalla un mensaje de error, automáticamente al pulsar en el botón “Aceptar” retornara a la ventana de logueo de Ping Box.



Gráfico III.32 Error de acceso

3.3.4.1.2.3.1.4 Conexión exitosa

Dentro del sistema Ping Box si los datos están correctos se tendrá acceso al sistema, imprimiendo un mensaje de conexión exitosa.

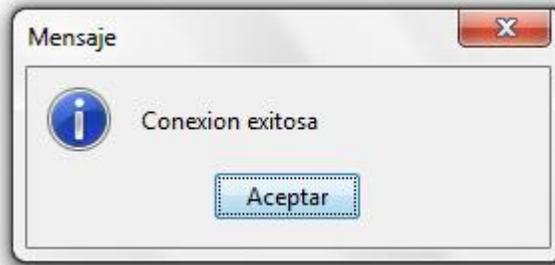


Gráfico III.32 Conexión exitosa

3.3.4.1.2.3.1.5 Control de ancho de banda y calidad de servicio

Esta es la ventana principal de Ping Box V.1.0 aquí se puede visualizar datos principales de las colas simples del sistema RouterOS, los datos aparecerán en una tabla distribuida en número o Flags, Nombre de la cola, Dirección IP asignada para la limitación, control de ancho de banda.

3.3.4.1.2.3.1.6 Limitación de colas simple (Queue Simple)

Los datos principales de cada usuario están almacenados dentro del sistema RouterOS, solo se visualizara datos principales como, nombre de usuario, direcciones IP, limitación de ancho de banda de subida y bajada, este último se lo concateno para balancear de mejor manera el tráfico de consumo.

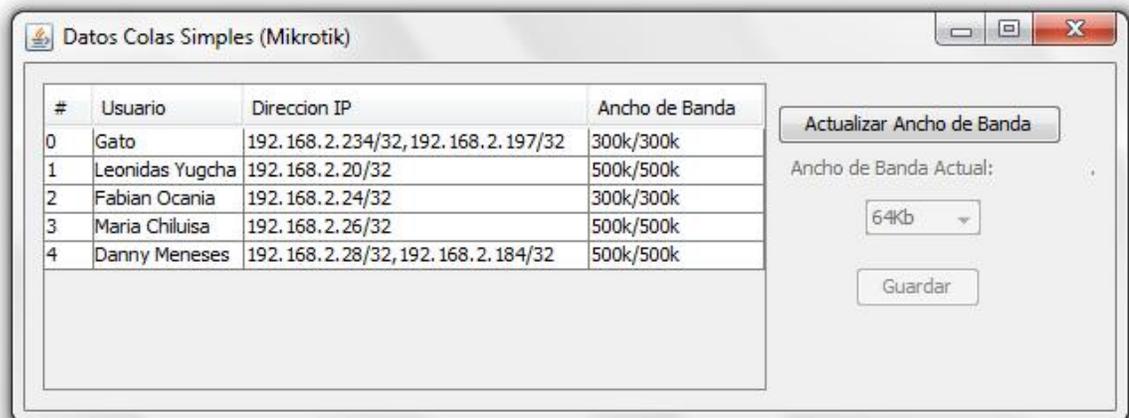


Gráfico III.33 Datos de usuarios

Ahora seleccionaremos al usuario destinado a realizar cambios o aplicar el balanceo de carga de manera que pueda quedar equilibrado.

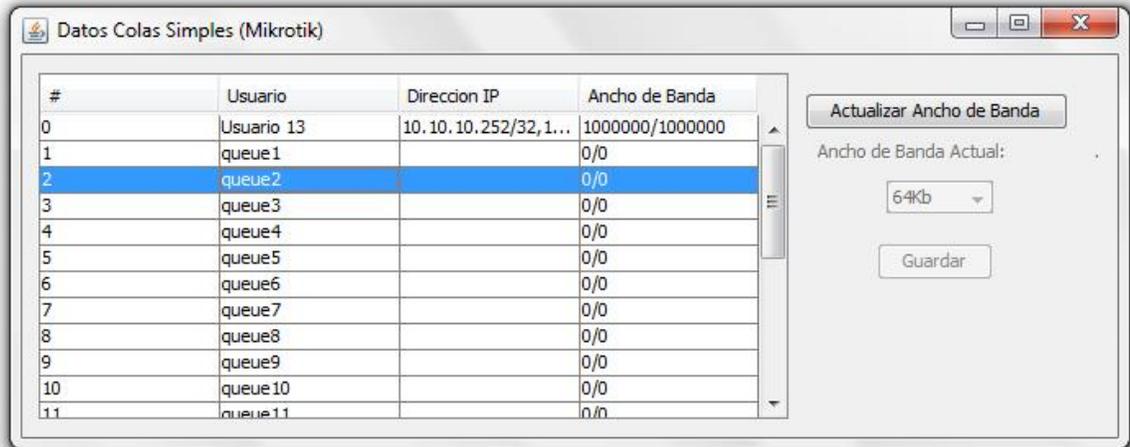


Gráfico III.34 Aplicación de cambios

El usuario de nombre “queue2” se lo ha seleccionado, posteriormente ejecutaremos un clic en el botón “Actualizar Ancho de Banda”, para que se habiliten el resto de opciones pendientes:

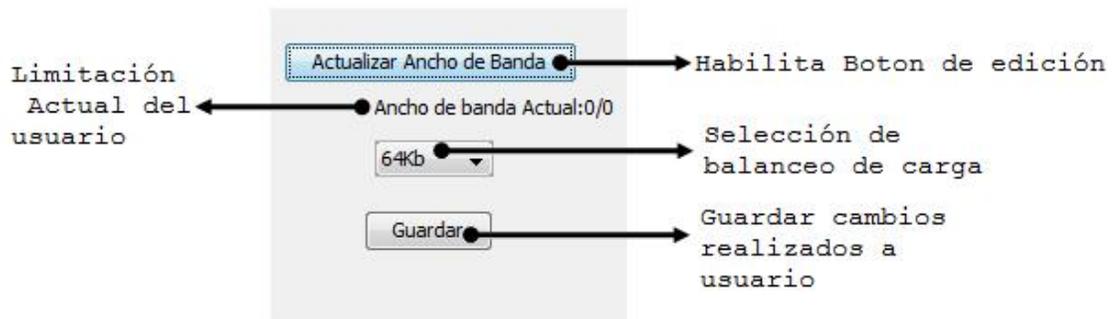


Gráfico III.35 Actualización de ancho de banda

3.3.3.1.1.2.3.1.7 Selección de balanceo de carga

En este proyecto se está utilizando el estándar de limitación de Routers Mikrotik por lo que no se asignara manualmente, se escogerá cada uno de los valores establecidos

por el mismo sistema, 64k, 128k, 256k, 512k, 1M, 2M este valor será seleccionado de acuerdo al requerimiento del usuario o administrador quien esté autorizado a cambiar o asignar una nueva limitación.

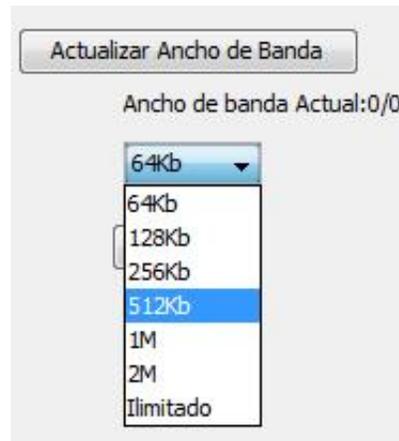


Gráfico III.36 Limitación de ancho de banda

Luego de seleccionar la limitación ejecutaremos un clic en el botón “Guardar”, inmediatamente el usuario tendrá una nueva limitación de ancho de banda, el programa automáticamente se cerrara por seguridad, ya que está diseñado para ejecutar una sola acción. Nos aparecerá un mensaje de “Usuario modificado exitosamente”, aceptaremos y ya nuestro usuario tendrá una nueva limitación de ancho de banda aplicando calidad de servicio.

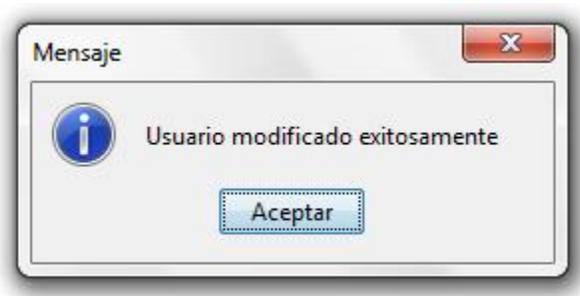


Gráfico III.37 mensaje de modificación exitosa

3.3.4.1.2.4 Comprobación del cambio de Limitación

Para medir el ancho de banda utilizaremos “Speed Test”, con esta herramienta puede medir inmediatamente y de forma gratuita la velocidad de su línea ADSL y de

conexión a Internet. Este es un script que puede calcular el ancho de banda o la velocidad de su conexión a Internet. Con este fin, comprobaremos si la eficacia del software “Ping Box” es la adecuada además este sitio web cuenta con un servidor independiente para garantizar que la medición sea lo más exacta posible. Pueden acceder al siguiente link: <http://www.speedtest.net/>

Su navegador debe de estar completamente actualizado para que la herramienta de medición pueda ejecutarse sin ningún inconveniente.



Gráfico III.38 Ingreso a Speed Test

Damos un clic en “COMENZAR PRUEBA”, y esperamos a que la herramienta nos devuelva datos de nuestra conexión.

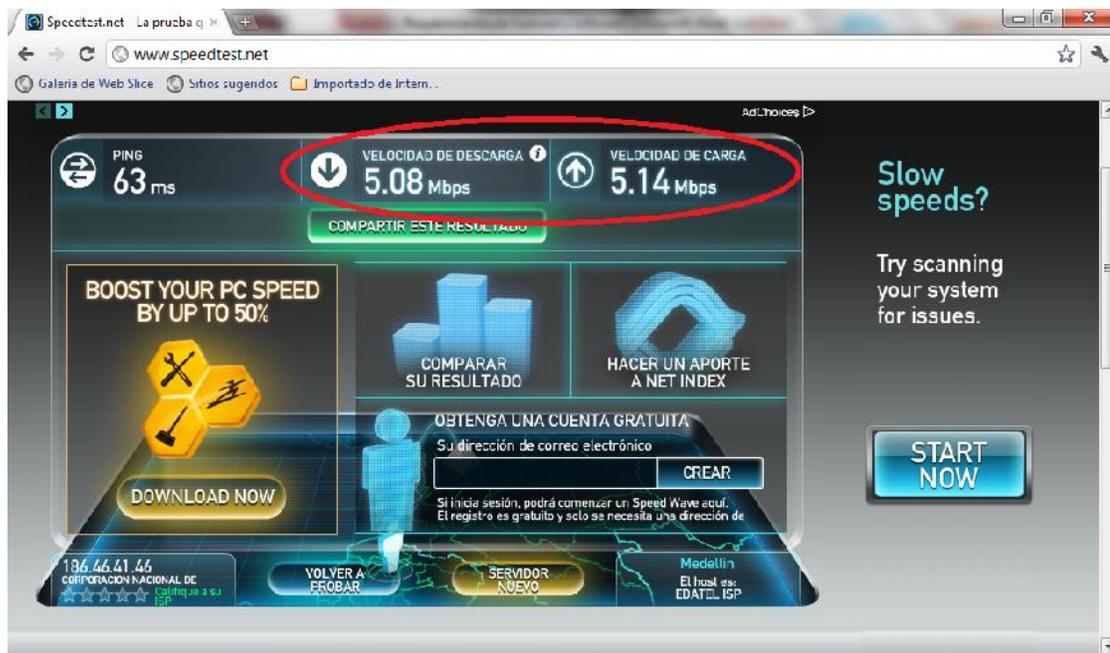


Gráfico III.39 Fase de comprobación de velocidad

Al terminar la medición nos dará resultados sobre la limitación de nuestra conexión a internet. Observamos que nuestro balanceo es equilibrado y se encuentra en 512k de subida y 512k de bajada.

3.3.5 Fase de prueba

Esta fase se realizó con los usuarios de la red Wlan de los Sistemas de comunicación Latacunga, en la cual se obtiene todos los resultados deseados para esta prueba.

3.3.6 Fase de funcionamiento

La fase de funcionamiento se encuentra habilitada en la actualidad en la red Wlan de los Sistema de Comunicación Latacunga, para cada uno de los usuarios, a quienes ya se podrá manipular el ancho de banda deseado.

CONCLUSIONES

- Al aplicar balanceo de carga estamos limitando equilibradamente el consumo de internet a través de las colas simples que pertenecen al sistema RouterOS de Mikrotik, por lo tanto el balanceo es persistente si aplicamos calidad de servicio al momento de limitar el ancho de subida como de bajada para mantener un estándar de consumo.
- La aplicación “Ping Box” es de fácil uso y diseñado para aquellos administradores de red puedan utilizar la aplicación sin ningún conocimiento de programación o manejo de comandos en algún “Terminal server”.
- La utilización de Ping box asegura un cambio adecuado, cuando el RouterOS se desconecte momentáneamente este intentara reconectarse para no afectar cambios en el sistema RouterOS, además de ser seguro en la modificación de ancho de banda.

RECOMENDACIONES

Emitida las conclusiones se manifiesta las siguientes sugerencias:

- Es aconsejable dar acceso a los dispositivos Mikrotik aquellos administradores que tengan permisos de uso del software para poder garantizar una seguridad al momento de operar el software “Ping box”.
- La aplicación es de fácil manipulación pero aun siendo esta así, es recomendable quien administre el presente sistema sea netamente personal técnico de la empresa, para que su uso sea óptimo.
- El ancho de banda aparte de ser una herramienta importante dentro de la distribución del internet es la parte fundamental para optimizar todos los recursos de la misma, por lo cual esta debe ser considerada y analizada antes de administrar la red.

BIBLIOGRAFÍA

BIBLIOGRAFÍA CONSULTADA

- Ceballos Javier.- Java 2. 3ra.- Edit. RA-MA, 2008
- Engst Adam.- Introducción a las Redes Inalámbricas. 1ra.- Edis. Anaya, 2007
- Flickenger Rob.- Redes inalámbricas en los países en desarrollo. 1ra.- Edit. Hacker Friendly LLC, 2008
- García Jesús, Raya José, Raya Victor.- Alta velocidad y calidad de servicio en redes IP. 1ra.- Edit. RA-MA, 2005
- Gastón Hillar.-Redes Inalambricas WiFi Edic. 1ra.- Edit. Hasa, 2005
- Huidobro José.- Redes y Servicios de Telecomunicaciones. 4ta.- Edit Thomson, 2008

BIBLIOGRAFÍA CITADA

(<http://www.maestrosdelweb.com/editorial/redeswlan/>)

(<http://multingles.net/docs/Manual%20-Z20Redes%20WiFi%20inalambricas.pdf>)

(<http://es.wikipedia.org/wiki/WAN>)

(http://www.elprisma.com/apuntes/ingenieria_de_sistemas/redesinalambricas/default2.asp)

(<http://www.mediafire.com/?mymt3jfyet>)

(<http://technet.microsoft.com/es-es/library/cc784767%28WS.10%29.aspx>)

(<http://elsitiodetelecomunicaciones.iespana.es/routers.htm>)

(<http://www.idg.es/computerworld/%C2%BFQue-es-el-balanceo-de-carga?/secciones/articulo-111063>)

(<http://qos.iespana.es/>)

(<http://qos.iespana.es/capitulo2.htm>)

(http://es.wikipedia.org/wiki/Calidad_de_servicio)

(<http://www.maestrosdelweb.com/editorial/redeswlan/>).- 15/11/2010.- H: 15h35

(<http://multingles.net/docs/Manual%20-Z20Redes%20WiFi%20inalambricas.pdf>).-
15/11/2010.- H: 16h50

(<http://multingles.net/docs/Manual%20-%20Redes%20WiFi%20inalambricas.pdf>)

(<http://es.wikipedia.org/wiki/WAN>).- 15/11/2010.- H: 17h15

(http://www.forpas.us.es/aula/hardware/dia4_redes.pdf).- 15/11/2010.- H: 17h40

(http://www.elprisma.com/apuntes/ingenieria_de_sistemas/redesinalambricas/default2.asp).- 15/11/2010.- H: 18h45

(<http://es.kioskea.net/contents/wireless/wlintro.php3>).- 23/11/2010.- H: 22h00

(<http://blyx.com/public/wireless/redesInalambricas.pdf>).- 25/11/2010.- H: 21h30

(http://www.redsinfronteras.org/pdf/redes_wireless.pdf).- 25/11/2010.- H: 22h00

(<http://www.masadelante.com/faqs/wireless>) .- 25/11/2010.- H: 23h00

(<http://www.mediafire.com/?mymt3jfyet>) .- 30/11/2010.- H: 19h50

(<http://www.adslnet.es/index.php/2006/04/18/tarjetas-minipci-wireless/>).-
30/11/2010.- H: 21h00

(<http://technet.microsoft.com/es-es/library/cc784767%28WS.10%29.aspx>) .-
02/12/2010.- H: 14h10

(<http://elsitiodetelecomunicaciones.iespana.es/routers.htm>).- 03/12/2010.- H: 18h15

(<http://www.idg.es/computerworld/%C2%BFQue-es-el-balanceo-de-carga?secciones/articulo-111063>) .- 6/12/2010.- H: 20h00

(<http://qos.iespana.es/capitulo2.htm>) .- 6/12/2010.- H: 21h25

(http://es.wikipedia.org/wiki/Calidad_de_servicio) .- 6/12/2010.- H: 22h00

(<http://www.routerboard.com/>).- 07/12/2010.- H: 21h00

(<http://www.pdfce.com/mikrotik-v20-router-software-technical-reference-manual.html>) .- 10/01/2011.- H: 20h50

(<http://es.wikipedia.org/wiki/Telnet>) .- 13/01/2011.- H: 17h50

(http://netbeans.org/index_es.html) .- 13/01/2011.- H: 20h45

(<http://es.wikipedia.org/wiki/NetBeans>).- 15/01/2011.- H: 20h00

(<http://technet.microsoft.com/es-es/library/cc784767%28WS.10%29.aspx>)24/02/2011.- H: 21h00

ANEXOS

**Encuesta Aplicada a los Usuarios de Internet de los Sistemas de Comunicación
Latacunga.**

1.- ¿La conexión de su Router inalámbrico es permanente?

Si

No

A veces

2.- ¿Está conforme con la calidad del equipo inalámbrico instalado en su domicilio?

Si

No

3.- ¿Es periódico el mantenimiento de su Router inalámbrico?

Si

No

A veces

4.- ¿Se debe aumentar el Ancho de Banda en los servidores por parte de Radio
Latacunga?

Si

No

5.- ¿Un servicio cableado mejoraría sus expectativas, al servicio inalámbrico que
recibe?

Si

No

6.- ¿Es satisfactorio el Internet en su Equipo?

Si

No

A veces

7.- ¿Recibe en su totalidad el ancho de banda deseado?

Si

No

8.- ¿Tiene problemas de descarga en el Internet?

Si

No

A veces

9.- ¿El servicio de Internet le permite distribuir en varios equipos?

Si

No

10.- ¿La distribución adecuada de Ancho de Banda mejorara el servicio de Internet que usted recibe?

Si

No

Entrevista realizada al Gerente de los Sistemas de Comunicación Latacunga

1 ¿Cuál es su apreciación por parte de usted, frente a la implementación del Sistema de control de manejo de banda para la empresa?

2 ¿Cómo mejorara la implementación del Sistema de Control, Balanceo de Carga y Ancho de Banda por Routers Mikrotik con Calidad de Servicio (QoS), para la red wlan (wireless local area network) en los Sistemas de Comunicación Latacunga el servicio de internet a los diferentes usuarios de la presente empresa?

Anexo 2 Entrevista