REPÚBLICA DEL ECUADOR
Escuela Politécnica Nacional
"E SCIENTIA HOMINIS SALUS"

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.

- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.

- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

***Respeto hacia sí mismo y hacia los demás.***

# ESCUELA POLITÉCNICA NACIONAL

## FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

## DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE GESTIÓN BASADO EN POLÍTICAS, PARA EL WISP (WIRELESS INTERNET SERVICE PROVIDER) TELYDATA CÍA. LTDA.

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y REDES DE INFORMACIÓN**

**ALEJANDRO AUGUSTO ANDRADE MAFLA**
**alexenti_4m@hotmail.com**

**DIRECTOR: ING. CARLOS EGAS ACOSTA**
**cegas@ieee.org**

**Quito, Marzo 2013**

# DECLARACIÓN

Yo, Alejandro Augusto Andrade Mafla, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

_____

Alejandro Augusto Andrade Mafla

# CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Alejandro Augusto Andrade Mafla, bajo mi supervisión.

_____

Ing. Carlos Egas
DIRECTOR DEL PROYECTO

# AGRADECIMIENTO

Envío un infinito agradecimiento a Dios por ser el motor eterno de mi vida. Agradezco a toda mi familia, en especial a mis padres llenos de sabiduría: Jaime Andrade y Mariana Mafla, quienes han sido un ejemplo de amor, honestidad, responsabilidad y perseverancia.

Agradezco a la empresa Telydata Cía. Ltda., al Ing. Fernando Padilla, al Ing. Diego Padilla y su equipo de trabajo porque a través de ellos he conseguido dar mis primeros pasos en mi profesión, guiándome y permitiéndome explotar los conocimientos adquiridos a lo largo de la carrera universitaria.

Gracias a todos mis profesores en la Escuela Politécnica Nacional, ya que han aportado para la realización de este proyecto de titulación.

Alejandro Augusto.

# DEDICATORIA

A Dios porque este trabajo es uno de los resultados de su amor por mi…….

A mi familia y amigos a quienes llevo en mi corazón…

A Juliana Hernández por ser mi ilusión de todos los días…….

Alejandro Augusto

# ÍNDICE

**CAPÍTULO 1**

**CAPÍTULO 2**

## ÍNDICE DE FIGURAS

## ÍNDICE DE TABLAS

# RESUMEN

En el presente Proyecto se realiza el diseño e implementación de un entorno de gestión basado en políticas para brindar al departamento técnico del WISP Telydata Cía. Ltda., una forma de gestionar usuarios y controlar la capacidad de acceso al canal a los mismos, con el fin de mejorar la calidad y el servicio a los usuarios en la ciudad de Quito.

En el primer capítulo se estudian los fundamentos teóricos básicos de la gestión de redes, así como la terminología establecida en el RFC 3198, que sirve de apoyo para el diseño e implementación del sistema. Se estudia de los Modelos de Información PCIM (*Policy Core Information Model*) y PCELS (*Policy Core Extensions LDAP Schema*). También se estudian los conceptos básicos de Servicios Diferenciados y su campo de estudio en la plataforma Linux.

En el segundo capítulo se realiza un estudio de la situación actual del WISP, permitiendo conocer su infraestructura, necesidades específicas en cuanto a su organización y tráfico de la red, que servirán de base para el planteamiento general del sistema.

En el tercer capítulo se establece el diseño del sistema definiendo un modelo de información propio para Telydata, que permita almacenar datos de usuarios y aquellos datos relacionados al tráfico usando el esquema PCELS. Se define los SLS (especificaciones del nivel de servicios) para los clientes respecto a la capacidad de acceso al canal y consecuentemente se realiza el diseño de *scripts* que permitirán ejecutar un controlador para la capacidad de acceso implementado en Linux.

Además se diseña un *script* para realizar reinicios programados de las antenas del backbone inalámbrico, ya que es un requerimiento por parte del Departamento Técnico de la empresa.

Finalmente en el tercer capítulo, se seleccionan las herramientas de software libre que permitirán la implementación y administración de dicho modelo de información en un servicio de directorio LDAP. Además se seleccionará la distribución Linux que implementará el controlador de capacidad de acceso.

En el cuarto capítulo se realiza la instalación y configuración de lo planteado en el tercer capítulo dentro en la red de core del WISP Telydata Cía. Ltda. Además, se realizan las pruebas respectivas del sistema de gestión y un resumen de los costos referenciales del Proyecto.

En el quinto capítulo se presentan las conclusiones y recomendaciones que se obtuvieron del Proyecto.

Finalmente en los anexos se incluyen entre otras cosas, la jerarquía PCELS, tablas que ayudaron en cálculos relacionados al tráfico y procesamiento del Monitoreo Server, en el segundo y tercer capítulo respectivamente. Además se incluyen facturas que se obtuvieron de la adquisición de dispositivos para validar los costos referenciales expuestos.

# PRESENTACIÓN

Dentro del área de gestión de redes en una empresa se requiere que aquellas variables indispensables que hacen posible la operatividad de la Red sean almacenadas, actualizadas y se encuentren disponibles cuando el personal encargado de la infraestructura lo necesite, en casos de soporte técnico a los clientes finales, para encontrar soluciones frente a incidencias o cuando se requiera ejecutar políticas con el fin de cumplir con determinados objetivos del negocio.

En este contexto, la gestión de redes basada en políticas ofrece un conjunto de mecanismos, no solo para implementar lo anteriormente expuesto, sino para ayudar a mejorar la calidad en el servicio a los clientes finales. En el presente Proyecto de Titulación se adoptan algunos aspectos de este tipo de gestión para el WISP Telydata Cía. Ltda., desde dos perspectivas básicas:

- Aportando con un servicio de directorio basado en LDAP para el almacenamiento y actualización de las variables de red inalámbrica usada cotidianamente para brindar soporte técnico a los clientes finales.
- Aportando con la ejecución de las políticas generadas por el Departamento Comercial respecto a la capacidad de acceso y tráfico de datos asignado a cada cliente.

Esto contribuye como una solución frente al problema de falta de documentación e indisponibilidad de información para la gestión de usuarios de la cual el Departamento Técnico de la empresa actualmente adolece. Además, con ello se busca contribuir con una herramienta de software adicional para el soporte técnico en cuanto al almacenamiento específico de datos y control de la capacidad de acceso para los clientes finales.

# ACRÓNIMOS

1. RIT: *Red Inalámbrica de Telydata*
2. RFC: *Request For Comment*
3. CLI: *Interfaz de Línea de Comandos*
4. ISP: *Proveedor del Servicio de Internet*
5. WISP: *Proveedor del Servicio de Internet Inalámbrico*
6. SLS: *Especificación del Nivel de Servicio*
7. PBNM: *Gestión de Red Basada en Políticas*
8. DN: *Nombre Distinguido*
9. LDAP: *Protocolo Ligero de Acceso a Directorios*
10. WLAN: *Red de Área Local Inalámbrica*
11. TDMA: *Acceso Múltiple por División de Tiempo*
12. CPE: *Customer Premises Equipment*
13. AP: *Access Point*
14. IANA: *Internet Assigned Numbers Authority*
15. VLAN: *Virtual Local Area Network*
16. ATM: *Asynchronous Transfer Mode (ATM)*

# CAPÍTULO 1

# FUNDAMENTOS TEÓRICOS

*En este capítulo se describirán los conceptos del entorno de gestión basado en políticas, así como los protocolos y estándares recomendados por el RFC 3198, se incluirá un resumen detallado de los conceptos referentes de sus componentes principales y tecnología. Adicionalmente, se describe el concepto básico de calidad de servicio, como uno de los objetivos que busca este tipo de gestión de red.*

## 1.1 GESTIÓN DE REDES

Es un proceso encargado de la generación de actividades, métodos o políticas coordinadas para alcanzar mayor disposición, rendimiento y calidad en los servicios que ofrece una red de datos. La importancia que tiene el correcto control y gestión de tecnología es crucial en una organización para alcanzar sus objetivos.
Algunas de las expectativas que persigue la gestión de la red son:

- Medir, administrar y controlar los recursos, tales como la capacidad del canal, procesamiento y almacenamiento

- Mantener un estado de cambios, para que constantes actualizaciones permitan mantener la integridad y disponibilidad de la información indispensable.

- Gobernar correctamente la infraestructura de una empresa que depende fielmente de su disponibilidad. Para tal objetivo se requiere un adecuado mantenimiento, supervisión, evaluación de las redes de comunicaciones.

- Mantener información acerca de los componentes del sistema y sus correspondientes configuraciones.

### 1.1.1 ESTRUCTURA BÁSICA DE LA GESTIÓN DE REDES

Inicialmente la estructura se basó en el modelo cliente servidor en donde los datos administrados se centralizaban en el servidor. Posteriormente se adoptaron las terminologías agente y gestor los cuales representaban a los actores principales en un sistema de gestión. En la arquitectura básica aparecen tres elementos: NMS, Agente y objetos.

- NMS (*Network Management Station*)

Representa el elemento gestor, el cual es independiente de la red y realiza monitoreo a los dispositivos para recibir constantemente información. Una red que desea ser administrada debe tener al menos un NMS. Está formado básicamente por un NMA (*Network Management Application*), el cual es elemento que contiene la aplicación que permite obtener datos del Agente en un entorno estructurado y entendible para el administrador de red.

- Agente

Se relaciona a un programa o conjunto de programas que colecciona información de administración del sistema, nodo, entidad o elemento de red. Están compuestos de objetos que remiten mensajes según las directivas enviadas por el gestor.

- Objetos

Son elementos de bajo nivel que se representan como entidades administradas, sean estos ruteadores, servidores, computadoras simples.

La Figura 1.1 indica una estructura básica de gestión de red.

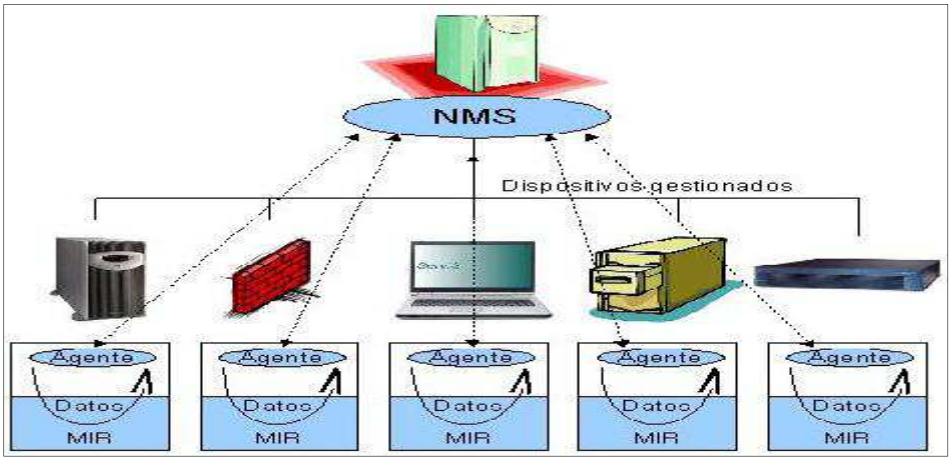



Figura 1. 1: Estructura de la gestión de Red [1]

## 1.2 GESTIÓN DE REDES BASADA EN POLÍTICAS (PBNM)

### 1.2.1 VISIÓN GENERAL DE PBNM

La gestión de redes basada en políticas (PBNM), entró por primera vez en las empresas desarrolladoras de tecnologías y equipos, construyendo productos de acuerdo a un conjunto de estándares conocidos. Los administradores que realizaron los primeros sistemas PBNM encontraron que no había un método simple y satisfactorio para solucionar problemas complejos de gestión de redes, a medida que las aplicaciones y productos fueron más sofisticados, con frecuencia entraban en conflicto en la red. Por ende, cada producto fue orientado a solucionar problemas solo en un contexto específico y ayudar a entregar un buen servicio al usuario final. Con el paso del tiempo se condujo a que este tipo de gestión se considerara sinónimo de aplicación a calidad de servicio. En consecuencia, las capacidades de PBNM no fueron desarrolladas en esencia, considerándose únicamente como un proyecto de investigación académica, sin una estandarización explicita para la gestión de redes en el mundo real y globalizado.

La premisa general de este tipo de gestión radica en la creación de políticas empresariales que permitan definir cómo manejar los recursos de una red en un determinado ambiente.

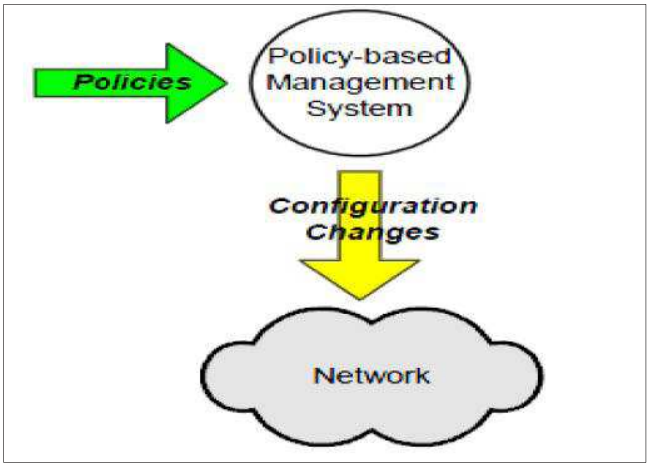



Figura 1. 2: Representación General de PBNM [2]

Tal como se muestra en la Figura 1.2, el establecimiento de políticas en PBNM implicaría cambios de configuración en la infraestructura de red para llevar a cabo lo que se desea lograr en una empresa. PBNM, en esencia se lo concibe como un camino para cumplir con directrices del negocio en una empresa y asegurar que se proporcionen los servicios que los usuarios requieren. Para ello, es posible usar dispositivos de marcas comerciales que implementen calidad de servicio o se puede usar un conjunto de técnicas y recomendaciones especificadas según el RFC 3198.
Las políticas en el sistema de gestión pueden ser de dos tipos:

- *Nivel Alto de Políticas* se define por el administrador, el cual se enfoca a determinadas reglas del negocio en una empresa.

- El *Nivel Bajo de Políticas* es cuando las políticas de alto nivel han sido traducidas a los distintos dispositivos de red, ya sean estos, enrutadores o servidores. En los equipos de red son un conjunto de sentencias o configuraciones de los equipos.

Como se pueden aplicar distintos tipos de políticas y en dispositivos diferentes, el IETF (*Internet Engineering Task Force*) ha estado trabajando en cómo almacenar, y compartir políticas e información para permitir la correcta administración de los recursos de red.

### 1.2.2 FUNCIONALIDAD BÁSICA DE PBNM

La gestión de red basada en políticas proporciona un modo específico de asignar recursos de red, calidad de servicio o seguridad, considerando los requerimientos y políticas antes establecidas por una empresa. Las funciones que debe comprender una gestión basada en políticas son:

- Tomar decisiones en cuanto a un recurso de red, analizando su estado actual y comparándolo con el deseado (dicho recurso de red es establecido o acordado entre el cliente y proveedor, por ejemplo la capacidad de acceso), estudiando la forma de cómo conseguirlo. La entidad que desarrolla esta funcionalidad se denomina punto de decisión de políticas (***PDP***)

- La aplicación de tales decisiones establecidas en el PDP se consigue gracias a comandos, que aplicados sobre los distintos dispositivos de red obtiene un comportamiento deseado y política aplicada. La entidad que desarrolla esta funcionalidad se denomina punto de ejecución de políticas (***PEP)***

- Los datos de recursos que requieren tanto PDP como PEP para su aplicación son tomados desde un almacén de información debidamente estructurado y organizado. La entidad que desarrolla esta funcionalidad se denomina ***Repositorio de Información.***

Generalmente, todos los componentes de PBNM se implementan bajo la misma plataforma por razones de consistencia. La comunicación entre PDP y PEP puede usar diferentes protocolos, generalmente, entre los más importantes: telnet/CLI, SNMP[1], COPS[2], SOAP[3]. El RFC 3198 brinda referencias en cuanto a plataforma y protocolos que pueden servir de guía para la implementación de este tipo de gestión.

---

[1] SNMP(*Simple Network Management Protocol*): intercambio de información para administración de redes

[2] COPS (*Common Open Policy Service*): modelo para control de políticas en dispositivos de red

[3] SOAP (*Simple Object Access Protoco*l): protocolo que define de procesos de intercambio de datos XML (*eXtensible Markup Language*)

En la Figura 1.3 se ilustra la arquitectura de PBNM.

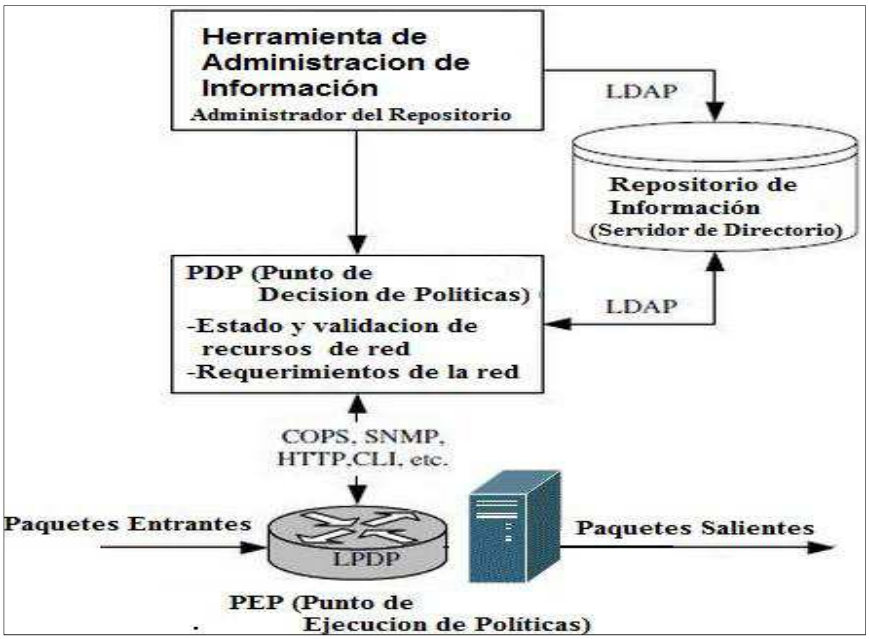



Figura 1. 3: Funcionalidad básica de PBNM [3]

## 1.3 RFC 3198 – TERMINOLOGÍA PARA LA GESTIÓN BASADA EN POLÍTICAS

El RFC 3198 describe una terminología compuesta de definiciones, áreas de uso, recomendaciones y estándares relacionados con la gestión de redes basada en políticas, así como del planteamiento de conceptos para su posible implementación. Algunas áreas de uso y protocolos que se recomiendan en este tipo de gestión y su implementación son: SLS (Especificación de Nivel de Servicios), Servicios Diferenciados (*DiffServ*), modelos para el almacenamiento de información y protocolos de gestión.

### 1.3.1 ESPECIFICACIÓN DEL NIVEL DE SERVICIOS (SLS)

Especifica el manejo de tráfico de clientes por parte del proveedor de servicios, este es previamente negociado entre un cliente, el proveedor y determinando las

características de los perfiles de cada uno de ellos. Promueve la definición de un ambiente *DiffServ* (Servicios Diferenciados) para su tratamiento en la red.

### 1.3.2 CALIDAD DE SERVICIO Y SERVICIOS DIFERENCIADOS

La calidad de servicio se refiere a la habilidad de entregar servicios de red de acuerdo a los parámetros especificados en un acuerdo de nivel de servicio (SLA). Hay dos tipos diferentes de enfoques de calidad de servicios: Servicios Integrados y Servicios Diferenciados. El campo de acción de los servicios diferenciados es en los nodos extremos de un dominio[4], en los cuales implementa reglas comunes para el tratamiento de paquetes que entran en la red. Por lo tanto, los servicios diferenciados permiten establecer criterios para la clasificar y controlar tráfico con el fin de lograr un determinado nivel de servicio en los clientes finales.

El campo de acción de los servicios integrados es en la implementación de una arquitectura que realice una reserva extremo a extremo de recursos en los elementos que conforman la red.

#### 1.3.2.1 Servicios Diferenciados en la plataforma Linux [4] [5] [6] [7] [8]

Las actuales versiones del kernel[5] de Linux reúnen un conjunto de importantes mecanismos enfocados a controlar tráfico. Existen tres elementos básicos: disciplinas de colas, clases, y filtros. Cada uno de ellos permite el acondicionamiento del tráfico en base a perfiles o requerimientos preestablecidos.

##### *1.3.2.1.1 Disciplinas de Colas:*

Es el algoritmo que gestiona el proceso de ordenamiento o encolado de paquetes que entran o salen de un dispositivo de red, dando forma a lo que se transmite para mejorar la capacidad del canal. Las disciplinas son categorizadas como: disciplinas con clase, o disciplinas sin clase.

---

[4] Una red o conjunto de redes que se encuentran bajo la misma administración, que permiten el transito del tráfico dirigido hacia clientes u otros proveedores de servicio.

[5] Kernel: programa que constituye el núcleo central del sistema operativo de una computadora, encargada de la administración de memoria, procesamiento y periféricos.

a) Disciplinas de colas sin clase: son aquellas que aceptan paquetes y se limitan a reordenarlos, retrasarlos o descartarlos y no contienen subdivisiones internas. Soportan los siguientes modos:

- *pfifo_fast*: es la disciplina más sencilla ya que opera en simple FIFO (*First Input First Output*), en donde los paquetes que entran primero a la interfaz de red, salen primero. Esta disciplina posee tres bandas con prioridades (0, 1, 2). Los paquetes que vengan con mayor prioridad van a la banda 0, los siguientes van a la banda 1 y el resto van a la banda 2. No puede ser cambiada por el usuario, pues ya tiene una configuración por defecto,
- *Token Bucket Filter* (TBF): La cual transmite solamente paquetes que llegan a una tasa que no excede a una preestablecida administrativamente. Sin embargo tienen la posibilidad de permitir cortas ráfagas de esta tasa.
- *Stochastic Fair Queuing* (SFQ), es una disciplina que intenta distribuir la capacidad del canal sobre un determinado interfaz de forma más justa o equitativa posible. Se basa en el algoritmo de encolamiento FQ (*Fair Queuing*) en el cual los paquetes son inicialmente clasificados en flujos, para luego ser asignados a una cola dedicada específicamente a un solo tipo de flujo. Luego cada cola es atendida por un algoritmo *Round Robin* (RR). El algoritmo RR da a cada sesión la oportunidad de enviar datos por turnos, permitiendo tener un comportamiento justo.

b) Disciplinas de colas con clase: Puede contener diferentes componentes de clases y es utilizada para dar tratamiento separado a diferentes tipos de tráfico, para saber que comportamiento dar a cada paquete se debe examinar a los filtros, los mismos que se llaman desde dentro de una disciplina de colas y no al revés.

Cada interfaz de red establece una forma en que los paquetes serán reenviados por una por el kernel de Linux, a esta se denomina disciplina de cola raíz de salida. Por defecto posee una disciplina de cola *pfifo_fast* sin

clase. Los controladores de estas disciplinas de colas consisten de dos partes, un número mayor y un número menor: <Mayor>:<menor >.

Según la Figura 1.4 se muestra la forma cómo se puede estructurar una clasificación y disciplinas de colas, por ejemplo un paquete puede clasificarse en una cadena como la siguiente:

1 ⟶ 1:1 ⟶ 1:12 ⟶ 12: ⟶ 12:2

Cada número separado por una flecha representa una clase, y cada clase solo habla de su clase paterna y de la disciplina de colas (qdisc), nunca directamente de una interfaz. Cada interfaz requiere de algún tipo de planificador, por defecto es pfifo.

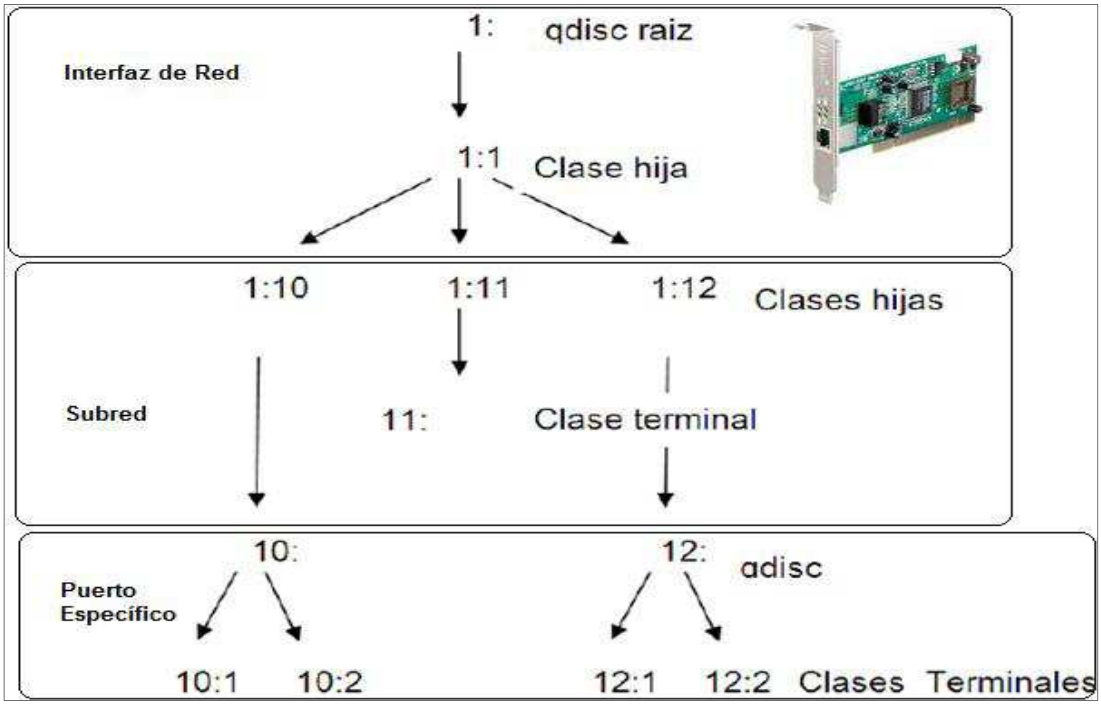



Figura 1. 4: Disciplina de colas con clase [4]

Por ejemplo, los paquetes que obedecen a la Figura 1.4 tendrían una estructura y orden de desencolado en base a una interfaz de red, subred y puerto específico. Las clases padres dictaminan el comportamiento de desencolado con el fin de poder

realizar un control de tráfico. Las disciplinas de colas frecuentemente usadas son: CBQ (*Class-Based Queuing*) y HTB (*Hierarchical Tocken Bucket*).

*1.3.2.1.2 Disciplina de Colas Class-Based Queuing (CBQ)*

Es una técnica para el control de tráfico a través de disciplinas de colas, en donde se establece un algoritmo de clasificación de paquetes entrantes, dentro de una jerarquía de clases y subclases. CBQ implementa básicamente WRR (*Weigthed Round Robin*)[6]. Cada flujo de datos que atraviesa por una interfaz de red se asigna a una clase o subclase, las cuales tendrán asignando un porcentaje de la capacidad del canal específico. En la Figura 1.5, se muestran tres tipos de colas generados en cada clase:

a) El tráfico a tiempo real, asignado el 25 % de la capacidad del canal
b) El tráfico interactivo asignado el 25 % de la capacidad del canal
c) El tráfico transferencia de archivos asignado el 50% de la capacidad del canal

Se tendrá la oportunidad de desencolar paquetes pertenecientes a la transferencia de archivos dos veces más que los paquetes pertenecientes al tráfico a tiempo real e interactivo. Los parámetros básicos para su configuración son: *bandwidth*, *rate*, *avpkt* al crear clases.

El parámetro *bandwidth* especifica la capacidad del canal disponible para el root *qdisc*, *rate* especifica la capacidad asignada para la clase y *avpkt* es el tamaño promedio estimado de paquetes para la interfaz. Dichos parámetros son usados para calcular el tiempo promedio de paquetes de transmisión, según la siguiente formula básica:

$$t = \frac{avpkt}{bandwidth}$$

[6] WRR: Algoritmo de planificación basado en el mecanismo Round Robin, en donde cierta cantidad de paquetes de la cola en cada turno de servicio.

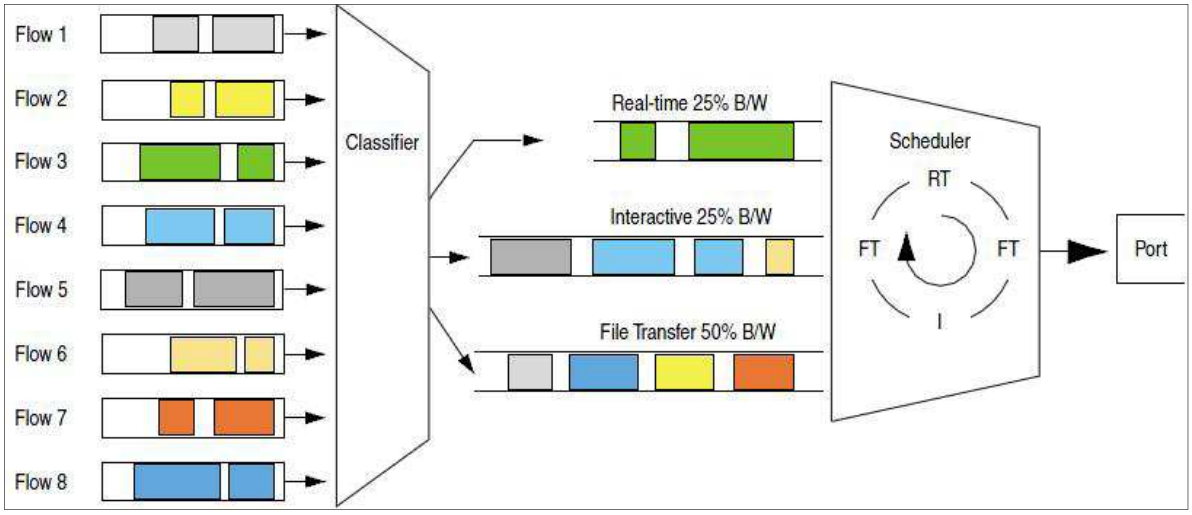



Figura 1. 5: Mecanismo de encolado en CBQ [5]

Por defecto, las clases de CBQ pueden pedir prestado capacidad del canal de sus clases padres si es que existe disponibilidad, sin embargo se puede usar el parámetro *bounded* para evitar que esto así sea. De igual manera, se puede usar el parámetro *isolated* para evitar que se tome prestado capacidad del canal de sus clases hermanas. Existe un parámetro denominado *Ingress qdisc* que puede ser usado para limitar la velocidad del trafico entrante, omitiendo aquellos paquetes que excedieron la velocidad deseada.

#### *1.3.2.1.3 Disciplina de Colas Hierarchical Tocken Bucket (HTB)*

Adopta un enfoque jerárquico que ajusta una cantidad de la capacidad del canal para diferentes propósitos dándole a cada uno de ellos una cantidad fija de la capacidad del canal garantizado y con la posibilidad de tomar prestada capacidad de otras clases. HTB usa el concepto de *tokens*[7] y *buckets*[8] usado en TBF (*Token Bucket Filter*) con un sistema basado en clases para permitir un control granular de tráfico. Algunos parámetros definidos son:

- *rate*: mínima velocidad de transmisión

[7] Token: piezas virtuales de información que representa un conjunto de bytes
[8] Bucket: determinada capacidad de almacenamiento en donde se encuentran los tokens

- *ceil:* máxima velocidad deseada para transmitir el tráfico, valor que puede ser alcanzado cuando existe préstamo de la capacidad del canal.

- *burst:* define el tamaño del *bucket* en bytes para un periodo dado. Es decir la máxima cantidad en bytes para que existan *tokens* disponibles. Generalmente es un parámetro fijo y dependiente del hardware.

- *cburst:* representa al valor máximo de velocidad en que el bucket acumula y desencola bytes. Valor dictaminado por hardware.

- *r2q:* es establecido por el usuario para ayudar a HTB a determinar un óptimo quantum para una clase particular. Su valor referencial es 10 [6]

- *quantum [7]:* representa la máxima cantidad en bytes que se entregarán desde una clase hija. Este parámetro se usa en HTB para controlar el particionamiento y préstamo del tráfico entre clases hijas, en base a la velocidad mínima de transmisión. Cuanto mayor sea el valor respecto a MTU, una clase hija prestará más *tokens* a su clase vecina. Este parámetro necesita ser calculado por el administrador en base a la siguiente relación:

$$quantum = \frac{rate}{r2q}$$

- *mtu:* tamaño máximo del paquete para una tasa dada. Su valor por defecto es de 1500 bytes

- *prio:* prioridad que se le da a una clase terminal, el valor más bajo será el más prioritario.

La estructura y conformado de clases se indica en la Figura 1.6.

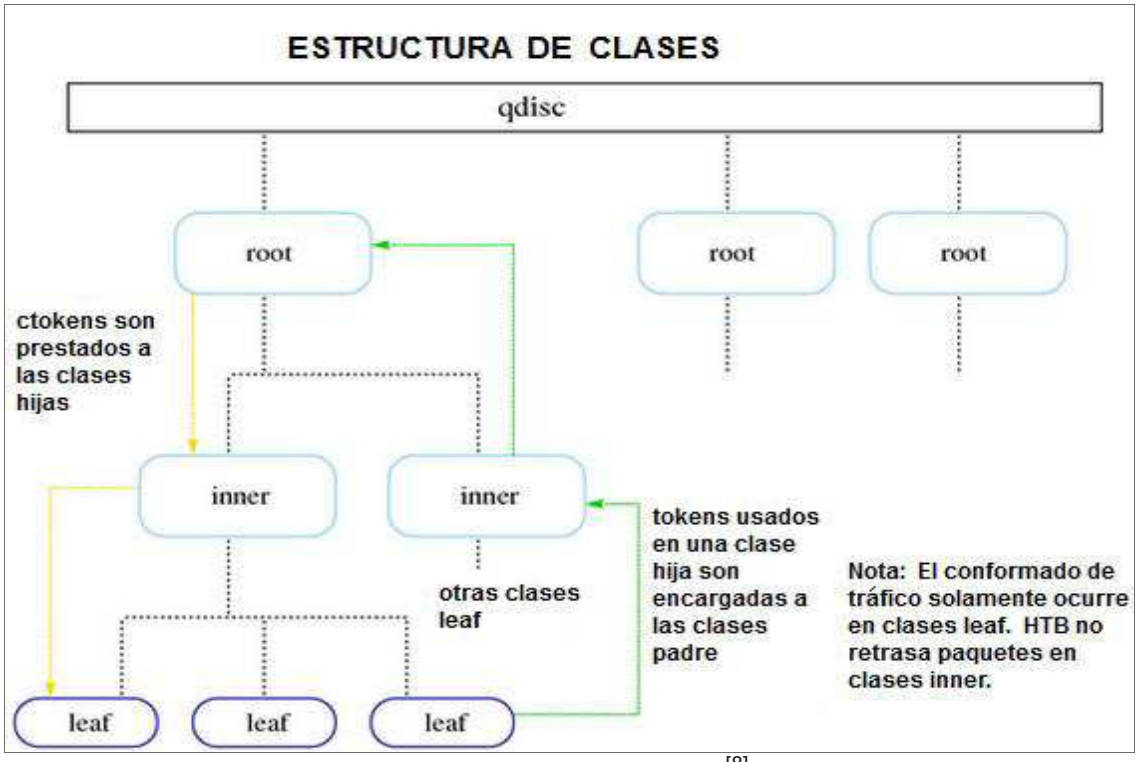



Figura 1. 6: Clases HTB [8]

En la Figura 1.6, se aprecia tres tipos de clases que se describen a continuación:

- Las clases *root* constituyen la parte superior de la jerarquía y todo el tráfico pasa a través de ellas.
- Las clases *inner* son aquellas que tienen clases padre e hijas.
- Las clases *leaf* son clases terminales o también denominadas bandas que tienen clases padre pero no clases hijas. El préstamo de *tokens* fluye hacia las clases leaf y son devueltos cuando la clase padre lo requiere.

*1.3.2.1.4 Cuadro comparativo entre CBQ y HTB*

En la Tabla 1.1 se realiza un cuadro comparativo en términos generales de control de tráfico y sencillez de uso.

| | CBQ | HTB |
|---|---|---|
| **Velocidad de datos** | Limita la velocidad de transmisión | Limita la velocidad de transmisión |
| **Prioridades** | Clasifica el tráfico de diferentes aplicaciones dependiendo de la configuración | Clasifica el tráfico de las diferentes aplicaciones dependiendo de la configuración |
| **Sencillez** | Difícil de configurar | Relativamente fácil de configurar |
| **Creación de clases Hijas** | Se puede crear tantas clases hijas como sean necesarias | Se puede crear tantas clases hijas como sean necesarias |
| **Repartición de la capacidad del canal** | Se puede repartir la capacidad del canal entre subredes pero su configuración es más compleja y no muy precisa | Permite repartir capacidad del canal entre direcciones IP o subredes. |

Tabla 1. 1: Comparación entre HTB y CBQ [4]

*1.3.2.1.5 Control de Tráfico*

El sistema operativo Linux puede procesar datos en la red, encolando paquetes en cierto orden para dar prioridad a ciertos flujos, retardando o únicamente reenviando paquetes. Los paquetes que llegan a una interfaz de entrada son verificados para el cumplimiento de reglas, si los paquetes cumplen con dichas políticas son directamente enviados a la red, o pasan a capas superiores según la pila del protocolo para procesarlas. Para el caso en que el sistema operativo Linux realice un control de tráfico se sigue el procesamiento indicado en la Figura 1.7.

Para el caso en que solo se use como ruteador, el proceso de los paquetes será el indicado en la Figura 1.8.

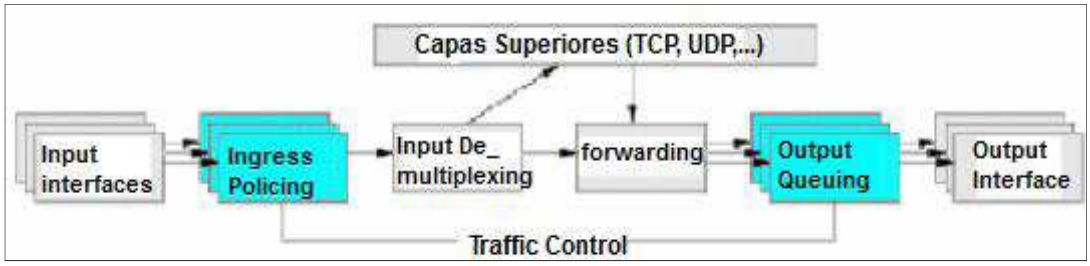



Figura 1. 7: Servidor Linux como gestor de Tráfico [4]

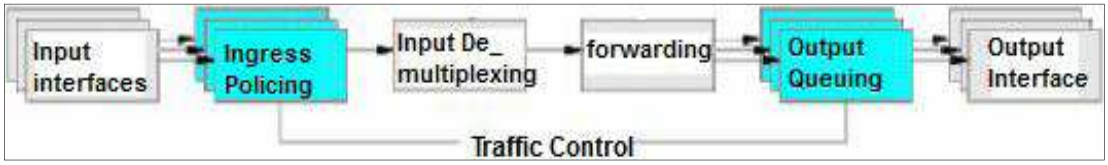



Figura 1. 8: Servidor Linux como ruteador [4]

En los bloques celestes de las Figuras 1.7 y 1.8 se ejerce control sobre los paquetes. Las políticas de entrada son el primer punto de control, y el segundo punto es el encolamiento de salida, en donde cada paquete es encolado; descartado, retrasado o priorizado de acuerdo a la regla implementada.

*1.3.2.1.6 Paquete iproute2*

Incluido en el sistema operativo Linux, especialmente a partir de las versiones de kernel 2.4.x.+, brinda funcionalidades entre otros de: control de tráfico, tunelización, multicasting y balanceo de carga. Para ello cuenta con dos herramientas principales:

- Herramienta tc, es la que permite desarrollar en sí la gestión de tráfico en una red.

- Herramienta ip, que administra las entradas a las tablas de enrutamiento.

*1.3.2.1.7 Herramienta tc*

Es una herramienta que interactúa con el kernel para la creación, borrado, o modificación directa de estructuras de control de tráfico, su sintaxis es:

tc [FORMAT] OBJECT {COMMAND | help}

En donde:

- [FORMAT] define cualquiera de las opciones {-s(tatistics) | -d(etails) | -r(aw) | p[retty] }
- OBJECT se refiere a si se va a manejar disciplinas de colas (*qdisc*)[9], clases (class), o filtros (filter)
- COMMAND define el conjunto de comandos que están disponibles para *qdisc*, class y filter

Es necesario entender los parámetros particulares de *qdisc*, *class* o *filter,* ya que pueden diferir uno de otro en cada objeto, y a su vez, pueden complementarse para realizar una adecuada configuración del control de tráfico.

A. Uso de tc para disciplinas de colas

El manejador *qdisc* permite encolar o desencolar paquetes de acuerdo a varios criterios, y sobre todo del algoritmo utilizado. La nomenclatura de tc para *qdisc* es:

tc qdisc [ add | del | replace | change | get ] dev STRING [ handle QHANDLE ] [ root | ingress | parent CLASSID] [ estimator INTERVAL TIME_CONSTANT] [[QDISC_KIND] [HELP [OPTIONS] ] ]

Como se puede verificar, para *qdisc* se tienen una serie de parámetros, a continuación se describe cada uno:

- *add*: añade una *qdisc* al dispositivo
- *del*: borra una *qdisc* al dispositivo
- *replace*: reemplaza una *qdisc* por otra en un dispositivo
- *change*: cambia una *qdisc*
- *get*: obtiene la *qdisc*
- *dev*: hace referencia a un dispositivo al que se relaciona la *qdisc*

[9] Disciplina de colas o *qdisc* implica un simple ordenamiento de paquetes a través de una cola.

- *handle*: es el manejador asignado a la *qdisc*, su forma es: <mayor: menor>
- *root*: le indica al dispositivo la *qdisc* está en la raíz y además es propietaria de toda la capacidad del canal que soporta el dispositivo. Puede ser de tipo egress[10]. También puede haber de tipo ingress[11].
- *parent*: determina la *qdisc* del padre (parte superior del árbol HTB).
- *estimator*: utilizado para verificar si todos los requerimientos en la *qdisc* han sido satisfechos.

*B. Uso de tc para manejo de clases*

Esta herramienta permite dividir diferentes tipos de tráfico que atraviesan un dispositivo como una tarjeta de red; utilizando clases y asignando una clase a cada tipo de tráfico y además se puede incorporar una velocidad de transmisión, esto con el fin de asegurar que siempre tenga una parte de la capacidad del canal disponible. También se lo emplea con el fin de que ninguna aplicación monopolice la capacidad del canal. Su sintaxis es la siguiente:

tc class [ add | del | change | get ] dev STRING [ classid CLASSID ] [ root | parent CLASSSID ] [CLASS_KIND] [ help | OPTIONS ]

Donde:

- *add*: añade una clase a un dispositivo
- *del*: elimina una clase de un dispositivo
- *dev*: hace referencia al dispositivo con el cual trabajará la clase
- *classid*: representa al manejador de la clase, es asignado por el usuario y su forma es <mayor:menor>
- *root*: representa la clase raíz en el enlace compartido.
- *parent*: representa el manejador de la clase padre.

[10] egress: hace referencia a la disciplina de encolamiento que maneja el tráfico de salida de la interfaz.

[11] ingress: hace referencia a la disciplina de encolamiento que maneja el tráfico de entrada de la interfaz.

Para las especificaciones de velocidad se tiene la siguiente nomenclatura:

mbps = 1024 kbps = [1024 x 1024] bps (bytes/segundo)
kbps = 1024 bps = 1024 (bytes /segundo)
kbit = 1024 (bits/segundo)
mbit = 1024 kbits (Kilobits/segundo)
mb = 1024 kb [1024 x 1024] b (byte)

Para la definición del tiempo:

ms, msec o msecs se refiere a milisegundos
us, usec o usecs define microsegundos

### C. *Uso de tc para manejo de filtros*

De igual manera con la herramienta tc se pueden crear filtros que permitan clasificar los paquetes que posean ciertas propiedades. Para lograr esto, las disciplinas de colas utilizan los filtros para designar los paquetes qué ingresan a una clase particular, estos pueden ser mantenidos por clases o por disciplinas de colas, dependiendo del diseño.

tc filter [ add | del | change | get ] dev STRING [ pref PRIO ] [protocol PROTO ] estimator INTERVAL TIME_CONSTANT ] [ root | classid CLASSID ] [ handle FILTERID] [FILTER_TYPE] [help | Options]

Donde:

- *add*: añade un filtro al dispositivo
- *del*: borra un filtro del dispositivo
- *dev*: se refiere al dispositivo sobre el cual se está ejecutando
- *pref*: define la prioridad asignada al filtro.
- *protocol*: identifica el protocolo con el cual el filtro trabaja
- *root*: indica que el filtro está en la raíz de la jerarquía del enlace compartido.

- *handle*: representa el manejador con el cual el filtro es identificado, su formato varía según el clasificador, el tipo puede ser u32[12], fw[13], route[14], entre otros.
- *classid*: define el manejador de la clase a la cual el filtro es aplicado.

A continuación se explica un ejemplo de cómo aplicar tc (control de tráfico).

- *Paso 1: Definición de la disciplina de cola raíz*

***tc qdisc add dev eth0 root handle 1 htb default 0 r2q 10***

En primer lugar se define la estructura de cola que se va a utilizar para manejar toda la tarjeta de red representada por eth0, generalmente se especifica una disciplina de colas con clase para posteriormente poder tener clases sobre los diferentes tipos de tráfico con su respectiva *qdisc*. Se tiene como manejador de esta cola al 1 (*handle* 1) En este caso se utiliza HTB.

- *Paso 2: Definición de la disciplina de colas para una clase terminal*

***tc qdisc add dev eth0 parent 1:4 handle 40 pfifo***

En la línea se añade una disciplina de cola al dispositivo eth0, definiendo el manejador del padre representado como 1:4 y el manejador de la clase terminal como 40. La clase terminal tiene una prioridad según el algoritmo pfifo.

- *Paso3: Definición de la estructura de clases*

***tc class add dev eth0 parent 1:4 classid 1:40 htb rate 500 kbit ceil 550 kbit burst 24 k quantum 6400***

[12] u32: es un filtro que permite discernir paquetes utilizando criterios como: dirección IP origen y destino, protocolo, puertos origen y destino o valores de campo TOS de la cabecera IP.
[13] fw: filtro que establece marcas a los paquetes con la herramienta iptables, generalmente las marcas son valores numéricos de cualquier valor.
[14] route: filtrado que se basa en los registros existentes dentro de las tablas de enrutamiento del sistema.

Se define una clase padre para las demás clases que se crearán a partir del manejador 1:4 en el dispositivo eth0, a esta clase padre generalmente se le asigna una cantidad de la capacidad del canal disponible para posteriormente administrarla. Se designa a la clase hija como 1:41 con una disciplina de colas del tipo HTB a una velocidad de 500 Kbps con un tope de 550 Kbps, dicha clase hija será la que tenga algún tipo de tráfico específico como el de un puerto o protocolo. El valor referencial del burst en Linux es de 24 k, el cual está relacionado a la cantidad de información del *bucket*. Asumiendo el parámetro r2q igual a 10, el quantum se calcula según la siguiente ecuación:

$$quantum = \frac{rate}{r2q} = \frac{500\ (kbit)}{10} * \frac{1024\ bits}{1kbit} * \frac{1\ byte}{8\ bits} = 6400\ bytes$$

Por lo tanto el valor del quantum es 6400 bytes, el cual cumple con la recomendación de ser mayor o igual al MTU[15] (Maximum Transmission Unit).

- *Paso 4: Filtrado del Tráfico*

***tc filter add dev eth0 parent 1:0 protocol ip prio 200 handle 4 fw classid 1:40***

Se añade un filtro a la eth0, para que los paquetes del protocolo IP con una prioridad de 200 se dirijan hacia la clase terminal 1:40.

*1.3.2.1.8 Herramienta iptables*

Es una herramienta a nivel de usuario que permite configurar reglas de filtrado, definición de firewalls o NAT en el kernel. Posee diferentes modos de operación como por ejemplo: hacer *flush*[16] de todas las reglas, configurar una acción por defecto para una cadena, agregar, insertar, reemplazar, borrar reglas, mostrar reglas, encerar estadísticas.

[15] Esto es debido a que cuanto mayor sea el valor de quantum respecto a MTU, una clase hija tendrá la posibilidad de prestar más tokens a una clase vecina para el conformado de tráfico.
[16] flush: elimina todas las reglas de una cadena, comenzando desde la que se haya especificado.

Los paquetes pueden ser marcados, etiquetándolos con un número específico para poder filtrarlos ya sea por dirección origen, dirección destino, puerto origen, puerto destino, o por identificación del protocolo.

Otro aspecto a tomar en cuenta al establecer reglas de filtrado es la acción a tomar cuando los paquetes cumplan con determinados criterios, a estas acciones se les denomina objetivos, los objetivos básicos son:

- ACCEPT: permite el paquete
- DROP: no permite el paquete
- RETURN: Se pueden presentar dos situaciones, la primera si la cadena es una subcadena de otra, entonces se regresa a la cadena principal, y la segunda es que si la cadena tiene *return*, entonces se aplicará la política por defecto[17]
- QUEUE: Pasa el paquete al espacio de usuario donde algún otro programa analizará el paquete para realizar alguna acción específica.

Nomenclatura básica de iptables:

*iptables [-t table] command [chain] [parameters] [-j target]*

*1.3.2.1.9 Tablas*

Se definen las tablas:

- *Filter*: es la tabla por defecto para reglas de filtrado
- *Nat*: define la tabla para reglas NAT (Network Address Translation)
- *Mangle*: define la tabla para la alteración de paquetes

[17] Por lo general existen dos políticas por defecto: aceptar todo y luego bloquear algunos puertos por seguridad o rechazar todo y aceptar solo lo necesario.

En la Figura 1.9 se indica un mapa de la utilización básica de cada una de las tres tablas.

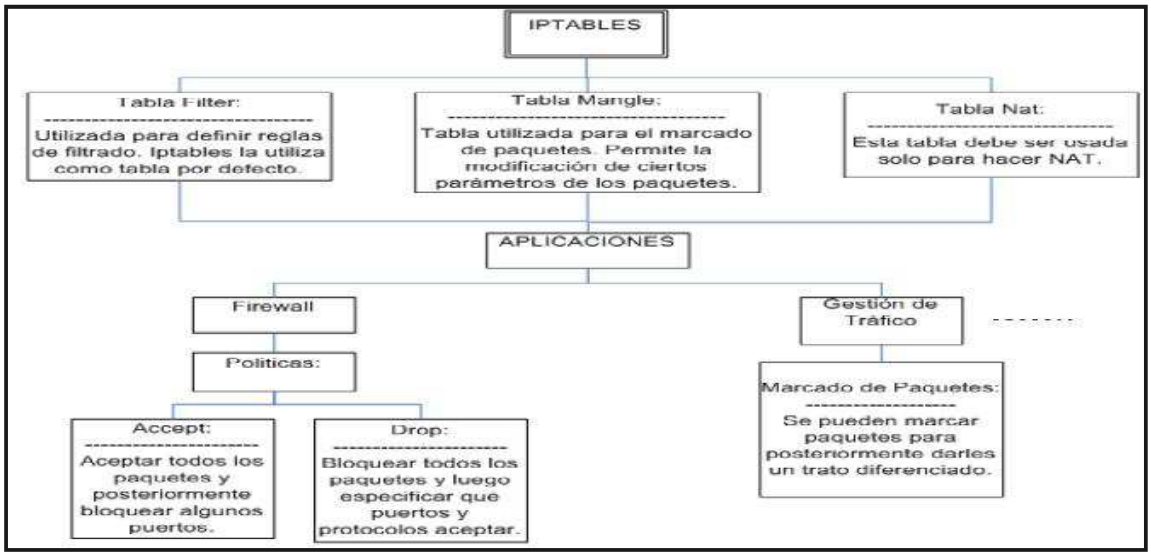



Figura 1. 9: Aplicaciones de la herramienta iptables [4]

*1.3.2.1.10 Cadenas*

Es una serie de reglas que son verificadas para un cierto tipo de paquete. Las cadenas por defecto son:

- *INPUT*: para paquetes de entrada enviados al host.
- *OUTPUT*: para paquetes de salida enviados desde este host.
- *FORWARD*: para paquetes enviados a través del host hacia otra máquina.
- *PREROUTING*: Manejo de paquetes antes de ser enrutados, para DNAT.[18]
- *POSROUTING*: Manejo de paquetes luego de ser enrutados, para SNAT[19]

Un usuario puede agregar sus propias cadenas, y cada una de ellas es verificada en orden. Cuando una regla no hace coincidencia, chequea la siguiente, y cuando si lo hace ejecuta la acción.

[18] DNAT (Destination Network Address Translation): cuando se requiere alterar la dirección de destino
[19] SNAT(*Source Network Address Translation*): cuando se requiere alterar o enmascarar la dirección de origen

Cuando un paquete entra a un dispositivo proxy o firewall, llega al hardware y alcanza el núcleo del sistema operativo por su driver correspondiente. Después el paquete sigue el flujograma presentado en la Figura 1.10.

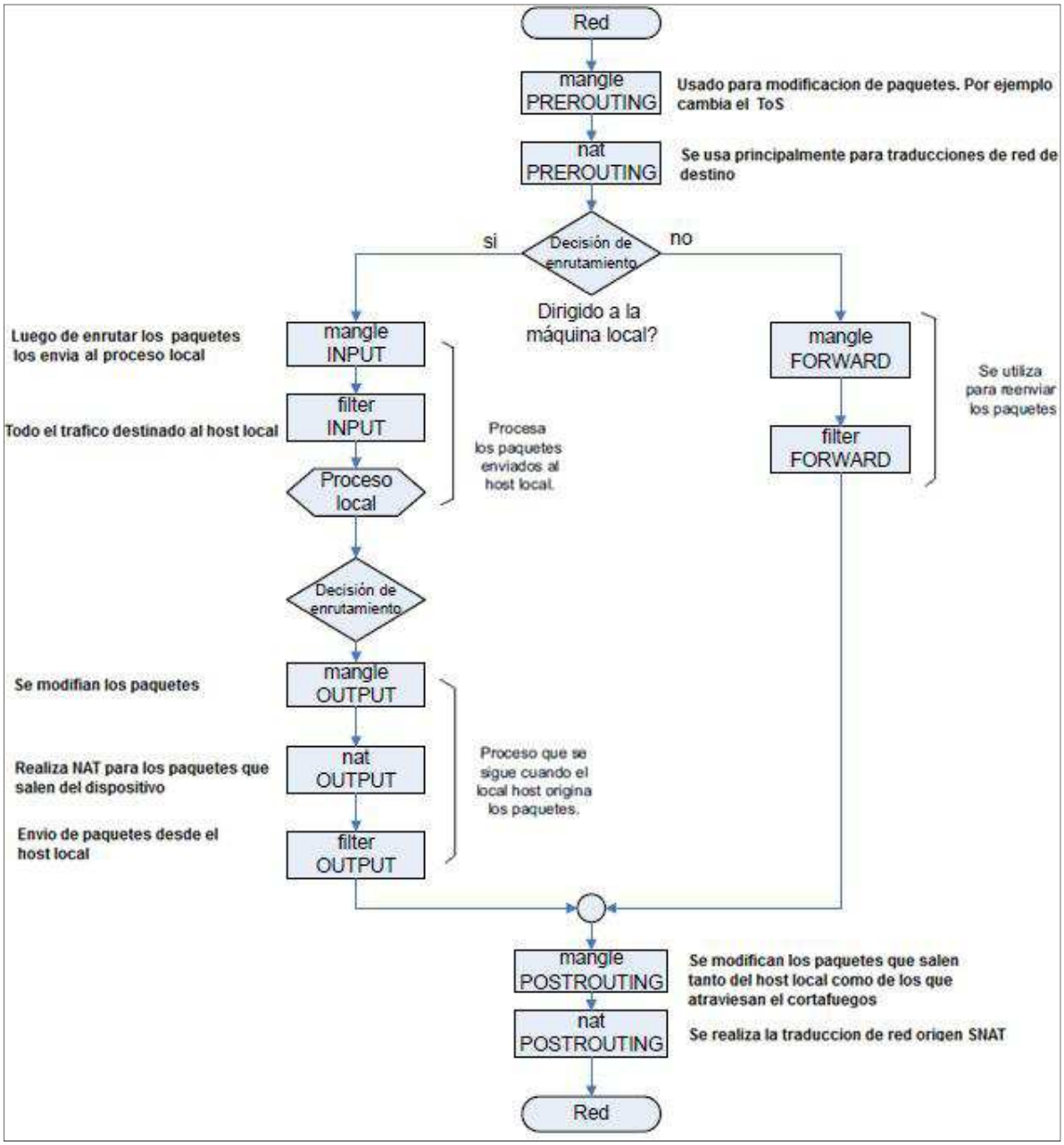



Figura 1. 10: Procesos al atravesar tablas y cadenas en Linux [4]

*1.3.2.1.13 Comandos*

Los comandos dicen al programa que operación realizar, entre los principales se tienen:

- -A, --*append*: añade la regla al final de la cadena.
- -D, --*delete*: borra una regla de la cadena. Esto se puede hacer de dos maneras. Bien introduciendo la regla a comparar, o bien especificando el número de la regla que se desea eliminar. Las reglas están numeradas progresivamente desde la primera, empezando con el número 1.
- -R, --*replace*: funciona de forma similar que el comando delete, pero en vez de eliminar completamente la entrada, la sustituye por una nueva.
- -I, --*insert*: inserta una regla en la posición de la cadena que se especifique. La posición se indica después de especificar la cadena.
- -L, --*list*: comando que ofrece una lista de todas las cadenas de una tabla especificada.
- -F: --*flush*: comando que elimina todas las reglas de una cadena, comenzando desde la que se ha especificado. Es equivalente a borrar cada regla una a una rápidamente. Si se la emplea sin opciones, esta borrara todas las reglas de todas las cadenas en la tabla especificada.
- -N, --*new-chain*: hace que el núcleo cree una nueva cadena con un nombre especificado, tomando en cuenta que no puede haber ninguna cadena ni objetivo con el mismo nombre.
- -X, --*delete-chain*: Este comando borra de la tabla una cadena dada, para ello se debería borrar o cambiar todas las reglas que tengan algún vínculo con dicha cadena antes de borrarla. Si se usa el comando sin opciones, todas las cadenas creadas por el usuario serán eliminadas y solo permanecerán aquellas que se instalan con iptables.
- -P, --*policy*: Este comando hace que el núcleo establezca la política u objetivo por defecto en una cadena. Todos los paquetes que no coincidan

con ninguna regla emplearan esa política de la cadena. Los objetivos permitidos son: DROP y ACCEPT.

*1.3.2.1.14 Objetivos / Targets*

Cuando la comparación de la regla que encuentra un paquete coincide con las condiciones que impone, alcanza el objetivo/salto donde se indica a la regla que debe hacer con el paquete. Existen dos objetivos básicos: ACCEPT y DROP; pero también se puede efectuar un salto, el cual ejecute otra cadena dentro de la misma tabla. Para crear una cadena se utiliza el comando –N. A continuación se indica un ejemplo:

```
iptables –N controlador.down

iptables –A  INPUT –p tcp –j  controlador.down
```

En primera línea se crea una cadena denominada controlador.down para realizar un salto siempre que se lo necesite. En la segunda línea indica que todos los paquetes TCP saltarán desde la cadena INPUT hasta la cadena controlador.down y serán filtrados por ella.

*A. Objetivo Accept*

Si la comparación de un paquete es satisfecha y se indica ACCEPT (aceptar) como objetivo, el paquete es aceptado y se salta del resto de verificaciones de la regla actual y cualquier otra verificación de la misma tabla. Para emplear este propósito hay que indicar –j.

*B. Objetivo Drop*

Si el paquete en la regla coincide, será desechado y no regresa nada de vuelta al servidor que envió el paquete. Se debe tener en cuenta que esta acción puede llegar

a tener en determinadas ocasiones un efecto no deseado, ya que puede dejar sockets[20] muertas en cualquier host.

En este entorno si se quisiera dar una solución más acertada, se usaría el objetivo REJECT, en donde el paquete es descartado y se envía un paquete de error al host que envió el paquete.

*C. Objetivo Mark*

Permite establecer marcas a los paquetes con valores concretos. Este objetivo solo es válido en la tabla *mangle* y no funcionará fuera de ella. Se debe tomar en cuenta que el valor de la marca no se establece en el paquete, sino que es un valor asociado a él dentro del núcleo, por lo tanto no se puede esperar que estableciendo una marca en un paquete, ese valor permanezca en otro host. Los valores de las marcas pueden usarse conjuntamente con las capacidades de enrutado avanzado de Linux, de manera que se envíen diferentes paquetes por diferentes rutas y se les pueda indicar el uso de los diferentes tipos de cola *qdisc*.
Para establecer la marca se tiene la opción *--set-mark* tomando un valor entero para un flujo específico de paquetes, o en todos los paquetes de un host específico. Posteriormente, se efectúa un enrutado avanzado sobre ese flujo o host para incrementar o disminuir la capacidad de la red.

A continuación se presenta un ejemplo:

- iptables –t mangle -A controlador.eth1-172.16.33.1 -m mark –mark 0 –p udp –j MARK --set-mark 10

En este ejemplo se añade una regla a la cadena controlador.eth1-172.16.33.1 de la tabla *mangle* en la cual se hace un chequeo por protocolo UDP y se lo marca con el

[20] socket: Está definido por el par de direcciones IP y remota, protocolo de transporte y un par de números de puerto local y remoto

número 10. La línea de código anteriormente expuesta puede considerarse como una especie de plantilla para el manejo de iptables en la marcación de paquetes.

### 1.3.3 DEFINICIÓN DE ESQUEMAS

El RFC 3198 define a los esquemas como: *"Un conjunto de reglas que determinan que información puede ser almacenada en una base de datos o servicio de directorio"*, o también como "*una colección de modelos de datos que son ligados al mismo tipo de repositorio*". En el contexto de sistemas de información, un esquema representa una recopilación de atributos y clases de objetos definidos para el control y almacén de datos.

### 1.3.4 MODELO DE INFORMACIÓN

El RFC 3198 define a un Modelo de Información como *"Una representación y abstracción de entidades del ambiente gestionado, sus propiedades, atributos, operaciones y la forma de cómo se relacionan entre sí. Es independiente de un repositorio específico, software, protocolo o plataforma".* Un modelo de información puntualiza una estructura de almacenamiento de acuerdo a una finalidad o entorno específico, esto significa la elaboración de mecanismos para representación y organización de datos. Para el entorno de una red de comunicaciones basada en términos de políticas existen dos modelos importantes: PCIM y PCELS.

#### 1.3.4.1 *Policy Core Information Model* (*PCIM*)

Según el RFC 3060, uno de los objetivos más importantes de PCIM y sus esquemas de extensión es crear un puente entre las políticas de alto nivel que define el administrador humano, y el lenguaje de aplicación real que deberían ser ejecutados en los nodos de red para llegar a los objetivos del negocio planeados.

Uno de los aspectos importantes de PCIM es la compatibilidad total con protocolos de acceso de directorios como LDAP. Su adaptación es casi total para todas las clases de LDAP, logrando compatibilizar la plataforma y optimizar tiempos de

respuesta. A través del modelado de PCIM se puede definir un conjunto de clases y relaciones como un medio para representar políticas, permitiendo controlar los elementos de una red de datos.

Por lo tanto, se puede modelar: el estado de una entidad de red, variables de la entidad de red a las cuales se requieran cambiarlas a un nuevo estado cuando se requiera, y también pueden ayudar a controlar aplicaciones.


Figura 1. 11: Ejemplo de Aplicación de Políticas [9]

En la Figura 1.11, se muestra un ejemplo de política representada en lenguaje de alto n donde la condición es que: si el departamento de una empresa es administrativo entonces asigne prioridad a aquel tráfico relacionado a videoconferencia. La variable tráfico representa a un recurso que se requiere gestionar eficientemente, tal que se cumplan con los objetivos planteados por los administradores de red. En PICIM, si se quiere representar dicha política se tendría que trabajar sobre la representación de clases presentada en la Figura 1.12, en la cual, las clases PolicyRule, PolicyCondition y PolicyAction definen la Regla, Condición y Acción respectivamente.

La Figura 1.12 indica una visión muy global de la representación de políticas, ya que se dispone de un conjunto más amplio y complejo de clases y subclases para una descripción más minuciosa de un sistema de gestión basado en políticas. A partir de las representaciones de PCIM se define el modelo PCIMe (*Policy Core Information Model extensions*) que incluye mapeos para varias implementaciones concretas, especialmente para directorios que funcionen como protocolos de acceso como es el caso de LDAP.

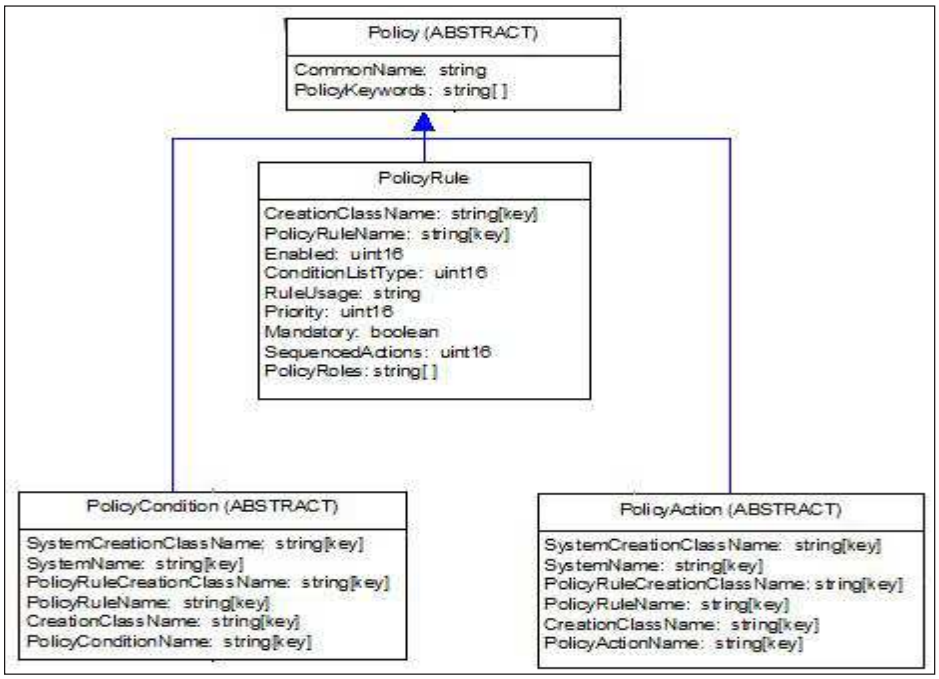



Figura 1. 12: Representación de políticas en PCIM [10]

### 1.3.5 LDAP (*Lightweight Directory Access Protocol*) [13]

Es un protocolo basado en X.500[21] para el acceso a información almacenada en un directorio. Está considerado como un modelo de red cliente servidor, en donde la información se almacena en objetos independientes, y relacionados jerárquicamente. Posee las siguientes características:

- Proporciona una respuesta rápida a operaciones de búsqueda o consulta.
- Funcionamiento sobre TCP/IP y SSL, puerto 389
- Interoperabilidad con varias aplicaciones adoptando interfaces de conexión a LDAP e integrándose fácilmente.
- Trabaja en entornos distribuidos: permite la replicación automática de datos y el uso de bases de datos distribuidas.
- Orientado a Objetos: el directorio representa a elementos y objetos. Los objetos son creados como entradas que contienen una colección de atributos.

[21] X.500:Es un conjunto de estándares de la ITU-T sobre servicios de directorio

- Es ampliamente adaptable con plataformas y aplicaciones con licencia tanto comercial como de código abierto.

LDAP está compuesto de varios modelos: información, nombrado, funcional y seguridad. A continuación se describe el modelo de información.

### 1.3.5.1 Modelo de información de LDAP [11]

Está representada por clases de objetos que están compuestos por un conjunto de atributos. Presenta tres conceptos importantes: entrada, esquema y atributo.

*A)* *Entrada*: está compuesta por el conjunto de atributos que puede tener múltiples valores con organización sin ambigüedades.

*B)* *Esquema*: Toda la definición de clases, atributos y tipos de datos se hace en el esquema, este define además de las clases y atributos permitidos, su sintaxis, que ayuda a mantener la consistencia del conjunto de datos. Facilita la interoperabilidad a través de los OID.[22]

*C)* Atributo: es un contenedor que almacena un solo tipo de información. Los atributos pueden ser requeridos u opcionales. Los atributos requeridos conocidos como MUST son aquellos que deben estar presentes en las entradas cuando se definan objetos. Los atributos opcionales conocidos como MAY pueden no incluirse en la definición de la clase de objetos. Un atributo tiene la definición de sintaxis que le corresponde y se representan mediante el par:

*tipo de atributo : Valor*

Un servidor LDAP puede implementar paralelamente varios esquemas simultáneos, respondiendo a varios pedidos de diferente información y con diferentes estructuras.

[22] Identificadores de Objetos: secuencia de números que se asignan jerárquicamente que identifican objetos en la red

La estructura de información es jerarquizada y se representa a través de un árbol de directorio denominado como DIT (*Directory Information Tree*), el mismo que puede ser usado a través de entradas. Las entradas poseen una combinación de atributos e identifican de manera única a un objeto en el repositorio de datos, a este se denomina como *dn* (*distinguished name*). Los elementos pertenecientes a una entrada son separados por comas e identifican el camino desde la raíz hasta los objetos. En la Figura 1.13 se muestra el DIT con sus componentes principales

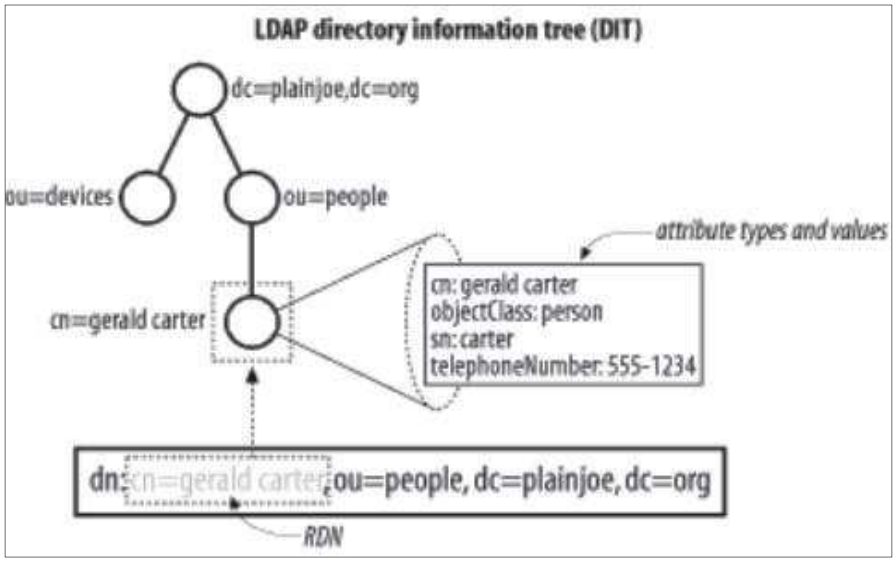



Figura 1. 13: Estructura de la información LDAP [11]

Por lo tanto:

- Un DN está formado por: Base DN + Un atributo único (también denominado Relative DN o RDN). Por ejemplo en la Figura 1.13 se tiene:

  Base DN: “ou=people,dc=plainjoe,dc=org”
  RDN: “cn=gerald carter”

- El Base DN es la combinación de atributos base que identifica a un objeto en el directorio LDAP.

En la Figura 1.14 se indica como ejemplo la definición de la clase organizationalUnit y sus atributos:

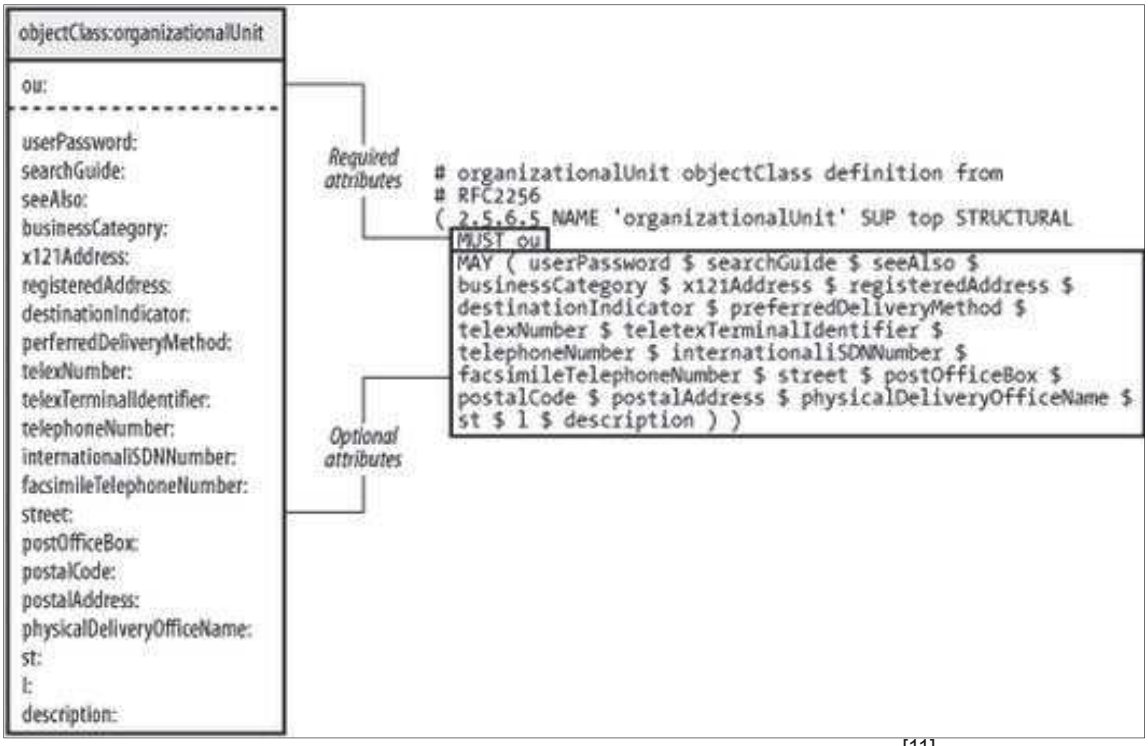



Figura 1. 14: Clase de ejemplo organizationalUnit [11]

En LDAP se puede crear un conjunto de clases de objetos propios con atributos que se vayan a necesita. En el esquema LDAP existen tres tipos diferentes de clases:

- *Abstracta*: definen características base para ser heredadas por otras clases (no por entradas). Todas las clases derivan de la superclase abstracta denominada “top”.

- *Estructural*: Definen las características de entradas del DIT. Representan objetos del mundo real y generalmente definen atributos obligatorios.

- *Auxiliar*: definen características adicionales (atributos adicionales) que se pueden incluir a las entradas del DIT.

La sintaxis de representación de todas las clases y atributos se pueden encontrar en los correspondientes ficheros *schema*.

#### 1.3.5.1 Representación de la gestión basada en políticas usando tecnología LDAP

Los modelos de información que usan LDAP contienen una representación no solo de políticas, sino también de un conjunto amplio para la representación de variables en una red de datos, esto es según las recomendaciones establecidas:

- RFC 3703: *Policy Core LDAP Schema (PCLS)*
- RFC 4104: *Policy Core Extension LDAP Schema (PCELS)*

En ellas se definen un conjunto de clases y atributos mapeados desde el modelo de información PCIM hasta la plataforma LDAP

#### 1.3.4.2 PCLS (*Policy Core LDAP Schema)*

PCLS se deriva del modelo de información genérico orientado a objetos dado en PCIMe y contiene una representación básica del esquema LDAP: sus clases y nombrado de atributos. Define más de 70 clases diferentes, entre estructurales y auxiliares. PCLS define un mapeo a las clases del modelo de información en el servicio de directorio LDAP versión 2 o versión 3; como protocolo de acceso. El mecanismo de mapeo involucrado desde PCIM a PCLS es de dos tipos:

- Para las clases estructurales en el modelo de información, el mapeo es básicamente uno a uno. Las clases del modelo de información se referencian a clases establecidas en LDAP.

- Para la relación de clases en el modelo de información, las relaciones de clases en PCIM y sus propiedades son traducidas en tres maneras: (a) como clases auxiliares; (b) con representación de atributos y referencias por nombre

distinguido (DN); y (c) para relaciones jerárquicas establecidas en el árbol de información de directorio (DIT).

### 1.3.4.3 PCELS (*Policy Core Extensions LDAP Schema*) [12]

La asignación de las extensiones del modelo PCIM a un esquema LDAP es un proceso que ha desarrollado IETF con las recomendaciones RFC 3060 y RFC 3703, en los cuales finalmente se describen las clases de objetos y tipos de atributos en LDAP y son prefijadas con el nombre *pcels*. Se presenta la jerarquía de clases combinada para LDAP en el **Anexo 1**.

La representación en PCELS es muy útil ya que el almacenamiento en un directorio puede adaptarse a las necesidades del administrador de infraestructura. Por ejemplo si se refiere a los dispositivos de red, tales como enrutadores, host o conmutadores, los cuales tienen una dirección MAC asociada, se los puede representar con la clase *pcelsMACAddrValueAuxClass,* la cual se encuentra esquematizada en PCELS de acuerdo a la Figura 1.15.

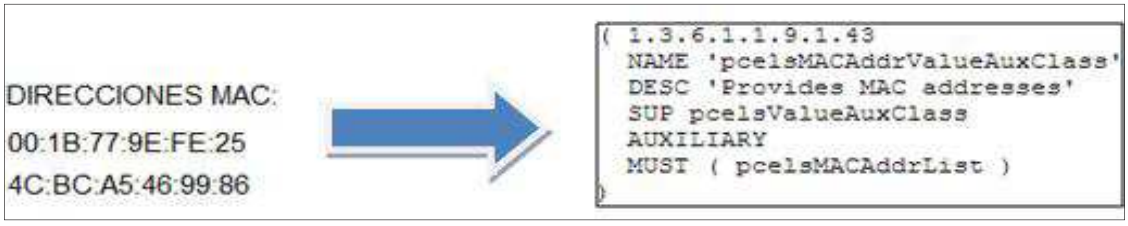



Figura 1. 15: Ejemplo para la representación de variables de red a un esquema PCELS [12]

Como se puede verificar en la definición de la clase de la Figura 1.12, se tienen tres partes importantes, la primera es que se define un OID[23] propio con número 1.3.6.1.1.9.1.43, la segunda es que define una clase superior, que en este caso es pcelsValueAuxClass, y la tercera es que define un atributo obligatorio denominado

[23] OID (Object Identifier)

pcelsMACAddrList. Dicho atributo representa un conjunto no ordenado de direcciones MAC o rangos de direcciones MAC.
En un servidor LDAP se incluye la representación PCELS copiando textualmente la nomenclatura del RFC 4104 a un fichero tipo *schema* usando un editor de texto.

La nomenclatura de los esquemas y estructura jerárquica de los esquemas pertenecientes a PCIM y PCELS se especifica en las respectivas recomendaciones RFC.

### 1.3.6 *SIMPLE NETWORK MANAGEMENT PROTOCOL* (SNMP) [14] [15]

Es un protocolo que comprende un conjunto de especificaciones para la gestión y administración de una red de datos, empleando los servicios ofrecidos por la arquitectura TCP/IP. Varios comités técnicos de Internet lo han desarrollado desde el año 1990 como una herramienta para gestionar dispositivos en cualquier red.

#### 1.3.6.1 Aspectos Generales

- Es un protocolo flexible que puede ser extensible a varios tipos de redes.

- SNMP puede gestionar servicios propietarios por medio de un componente *proxy*, utiliza la capa de transporte TCP/IP mediante el envío de datagramas UDP. Los puertos comúnmente utilizados son 161 y 162.

- El desarrollo del protocolo ha permitido establecer las versiones SNMPv1, SNMPv2, SNMPv3 que han sido ampliamente recomendadas en Internet.

- Trabaja a nivel de capa aplicación y está descrita principalmente en el RFC 1157, la cual brinda especificaciones que contiene entre otras cosas, la arquitectura y todo lo relacionado a definiciones generales de objetos.

Además, se tienen esencialmente tres estándares relacionados a SNMP descritos en varios RFC y publicados por el IAB (Internet Activities Board)[24], propuesta del mismo IETF. Estos son:

- RFC 1155(SMI[25]): Estructura e identificación de la información de gestión para Internet basada en TCP/IP. (Mayo 1990).

- RFC 1156(MIB): Para gestión de red en Internet basada en TCP/IP: MIB- II. (Marzo 1991).

### 1.3.6.2 Arquitectura del Protocolo

Se compone de los siguientes elementos básicos:

- Estación de gestión (Gestor en el modelo OSI): dispositivo en el cual se almacena la información de gestión proveniente de los agente.
- Agente: Es un dispositivo capaz de comunicar y ejecutar un procesos para recuperar determinados valores de una MIB
- MIB (*Management Information Base*): contiene la información estructurada de los dispositivos gestionados en la red de comunicaciones.

SNMP usa un servicio no orientado a conexión a través del protocolo UDP (User Datagram Protocol), enviando mensajes con formatos ya establecidos llamados PDU (Protocol Data Unit)[26] específicos e información de seguridad. Los puertos usados son: 161 mensajes de envío (para mensajes de petición respuesta) y 162 para mensajes de notificación mediante Traps[27]. En la Figura 1.16 se representa la arquitectura del protocolo.

---

[24] Es un comité sin ánimo de lucro que gestiona y diseña sistemas para la evolución de Internet
[25] El SMI define las reglas para describir los objetos gestionados y cómo los protocolos sometidos a la gestión pueden acceder a ellos
[26] PDU: Es una unidad de datos que entrega una entidad de red a otra con información de control y estado
[27] Traps: es un mensaje que permite la notificación de eventos significativos desde los agentes hacia los gestores de manera asíncrona

SNMP v1 (versión 1): realiza operaciones simples de petición respuesta, utiliza los cinco mensajes básicos anteriormente descritos. Está descrito en el RFC 1155, 11571, 1212 del IETF.

Utiliza comunidades y puede proporcionar diferentes categorías de acceso estableciendo el control a través de acciones de solamente lectura (RO), solamente escritura (RW) y vistas, además puede establecer una colección de información que se requiera para un determinado ambiente. Se definen cinco tipos de operaciones permisibles con sus objetos para representar formatos de mensajes de la siguiente manera:

- *GetRequest, GetNext Request*: se utilizan para leer información de un elemento de Red.
- *SetRequest*: Se utilizan para realizar cambios en los objetos establecidos.
- *GetResponse*: sirve para confirmar o responder a los tipos de mensajes anteriormente descritos.
- *Traps*: Se utilizan cuando se genera un evento de los elementos gestionados. Este tipo de mensajes no necesitan respuesta o confirmación.

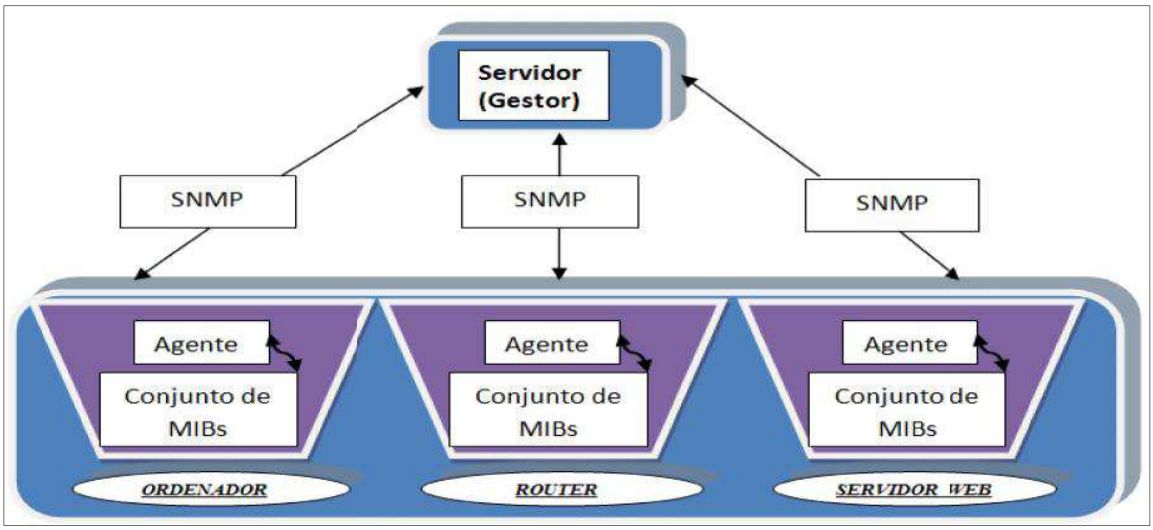



Figura 1. 16: Arquitectura del Modelo de gestión de Red Internet [14]

En la Figura 1.17, se indican los formatos de los mensajes SNMP.

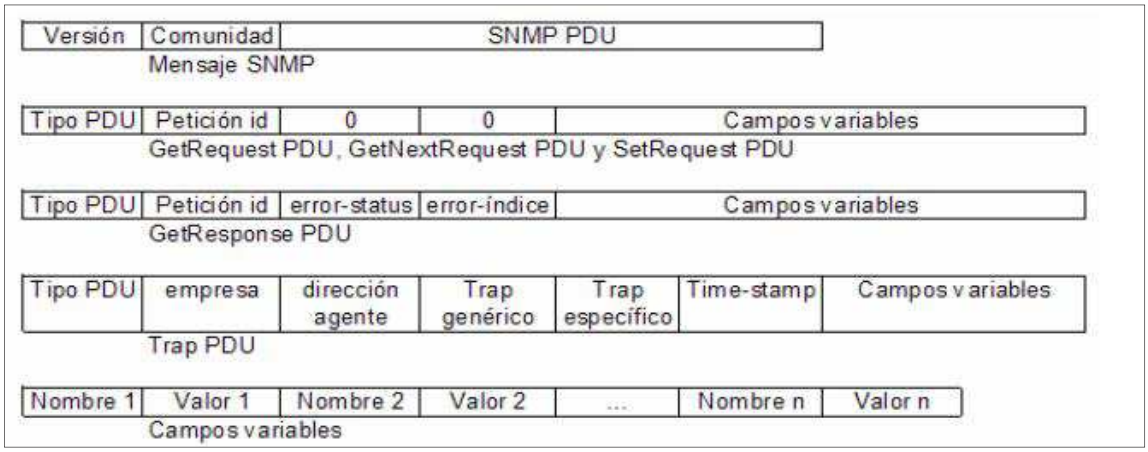



Figura 1. 17: Formatos de mensajes SNMP [15]

SNMP v2 (versión 2): surgió con el fin de perfeccionar su funcionalidad y seguridad. Las operaciones sobre el protocolo varían respecto a la versión 1, estableciendo dos nuevos PDU.

- *GetBulkRequest*: utilizado cuando se solicita la transferencia de una gran cantidad de datos. Es parecido al comando *GetNextRequest* solo que no solicita objeto por objeto sino que puede solicitar una gran cantidad de una vez.

- InformRequest: Es enviado desde una estación SNMP para notificar al gestor que un evento ha ocurrido, o que se presenta una condición especifica

SNMP v3 (versión 3): Añade mejoras de seguridad y administración respecto a snmpv2. En cuanto a la seguridad define un modelo de seguridad basado en usuarios (USM) RFC 2274, definiendo aspectos de autenticación con algoritmos MD5 o SHA1, privacidad, encriptando el contenido utilizando el algoritmo DES[28].

[28] DES (*Data Encription Standard*): es un algoritmo de cifrado de datos, el cual toma un texto claro de longitud fija y lo transforma en un texto cifrado de igual longitud utilizando una clave criptográfica.

En lo que respecta a la administración, se define el modelo de control de acceso basado en vistas. (VACM) el cual gestiona usuarios basado en vistas sobre la MIB actual, permitiendo restringir acceso a ciertas partes de la MIB.

### 1.3.7 Bases de Información de Gestión (MIB) [15]

La Base de Información de Gestión es una colección de objetos que representan de forma abstracta a los elementos gestionados y sus componentes. Se organizan en grupos según sea su temática. Una estación gestora usa una MIB para modificar los valores de objetos, garantizando con ello interoperabilidad. Su estructura es jerárquica como se indica en la Figura 1.18.

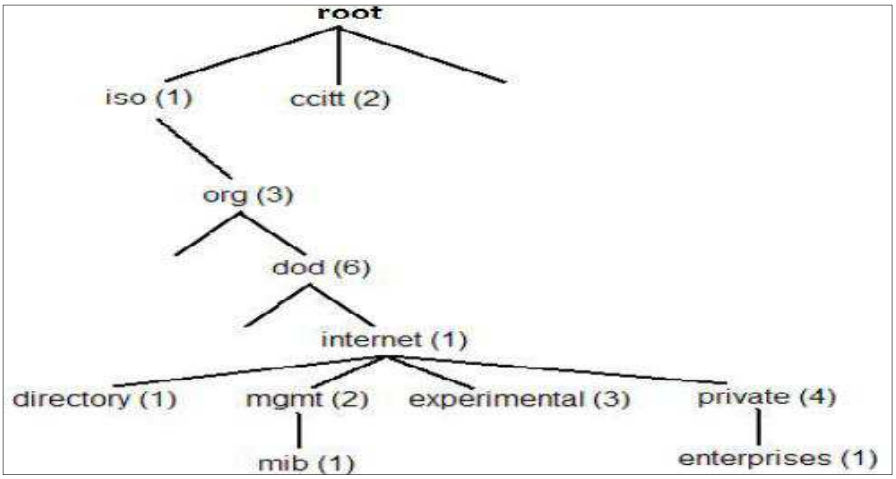



Figura 1. 18: Estructura General de la MIB [15]

Existen demarcados cuatro tipos diferentes de MIB:

- MIB Estandarizadas. MIB-I (RFC 1156) y MIB-II (RFC 1213). Base de información que se identifican a partir del OID[29]: iso(1).org(3).dod(6).internet(1).mgmt(2).

- MIB Experimentales. Base de información se identifica a partir del OID: iso(1).org(3).dod(6).internet(1).experimental(3).

[29] OID(Object Identifiers): Son una secuencia de enteros positivos separados por un punto que identifican objetos en un árbol

- MIB Privadas. Base de información que se identifican a partir del OID: iso(1).org(3).dod(6).internet(1).private(4).

En la Figura 1.19 se indica el marco de trabajo para la administración de red SNMPv1.

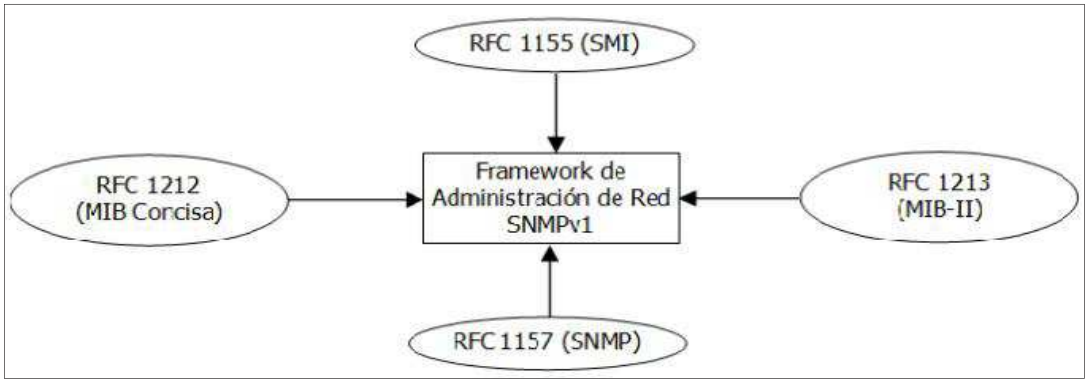



Figura 1. 19: Marco de trabajo para SNMP v1

# CAPÍTULO 2

# ANÁLISIS DE LA SITUACIÓN ACTUAL DE LA RED DE TELYDATA.

*En éste capítulo se analiza el estado actual de la red de datos de Telydata Cía. Ltda. Se describe su infraestructura física y lógica. Luego se hace énfasis en el estado de la red inalámbrica respecto al backbone de la red y su requerimiento organizacional para su operación. Además se realiza un análisis del tráfico de la red, capacidad del canal, descripción de usuarios.*

## 2.1 DESCRIPCIÓN GENERAL DE TELYDATA TELECOMUNICACIONES Y DATOS

Telydata Cía. Ltda., es una empresa ecuatoriana que se dedica a brindar servicios de Telecomunicaciones, transmisión de datos, y aprovisionamiento de Internet a nivel nacional. Está considerado como un proveedor de servicios de Internet e involucrado dentro del grupo de Servicios de Valor Agregado (SVA) establecido por la SENATEL desde el 23 de Junio del 2002. Durante el tiempo transcurrido, ha logrado comercializar servicios de Internet principalmente a través tres medios de transmisión: ADSL[30], WiFi y Fibra Óptica. Actualmente cuenta con más de 700 usuarios entre clientes internos y finales distribuidos en la ciudad de Quito, Ambato, Latacunga, Ibarra, Otavalo, Guallabamba y Cayambe. Además, se ha convertido en un apoyo importante en el área del soporte técnico para otras empresas corporativas como lo son Ecuawagen, DWconsulware, Uribe Shwarzcoft, entre otras. Sus oficinas principales se encuentran ubicadas en las avenidas Amazonas N69-169 y Gaspar de Villarroel. Edificio Reinoso.

[30] ADSL (Asymmetric Digital Subscriber Line*):* Tecnología de última milla utilizada para el acceso a Internet a velocidades diferentes para carga y descarga de datos.

*Visión* [17]*:* Telydata está enfocada a ser líder en el mercado de las telecomunicaciones a nivel nacional, afianzando la empresa como ejemplo de desarrollo humano y profesional, además, brindando a los clientes el mejor servicio en base a las soluciones tecnológicas disponibles, adecuadas para cumplir con sus más altas exigencias.

### 2.1.1 APROVISIONAMIENTO DEL SERVICIO DE INTERNET INALÁMBRICO

Desde hace dos años, la empresa se encuentra en un plan de desarrollo y mejoramiento del servicio de Internet inalámbrico en la ciudad de Quito, tanto en el sector norte como sur, especialmente enfocándose en las áreas rurales, donde existe dificultad para el acceso a Internet. Hasta el momento las diferentes empresas de aprovisionamiento de Internet existentes en el mercado no han tenido cobertura.

#### 2.1.1.1 Sectores de Cobertura y operación de la Red Inalámbrica

Para el sector norte de la ciudad se tienen las zonas de perímetro rural que se encuentran ubicadas a los alrededores de la parroquia de El Condado, cubriendo los sectores:

1. 23 de Junio
2. Jaime Roldós
3. Pisulí
4. La Planada

El punto de acceso inalámbrico principal se encuentra ubicado en la Planada, con las coordenadas: *Latitud*: S 00° 05´ 45.2”, *Longitud*: W 78° 30´59.4".

Para el sector sur de la ciudad se tienen las siguientes zonas de perímetro rural y urbano ubicadas a los alrededores del Camal Metropolitano para los siguientes sectores:

1. Guamaní
2. La Ecuatoriana
3. Hacienda Ibarra
4. Camal Metropolitano

El punto de acceso inalámbrico se encuentra ubicado en el Camal Metropolitano, con las coordenadas: *Latitud:* S 00°20´05.8"; *Longitud:* W 78°34´22.5".

## 2.2. SITUACIÓN ACTUAL DE LA RED DE DATOS DE TELYDATA CÍA. LTDA.

El backbone del ISP está constituido por un conjunto de equipos de comunicaciones marca CISCO, tanto enrutadores como switches. Lo que respecta al resto de equipos activos como servidores, firewalls, proxy son implementados bajo la plataforma GNU/LINUX. Actualmente se cuenta con enlaces de acceso vía fibra óptica por salidas internacionales provistas por CNT-EP (Corporación Nacional de Telecomunicaciones - Empresa Pública) y Megadatos. En la Figura 2.1, se muestra la topología principal de la red de core (principal) actual de la empresa.

### 2.2.1 BREVE DESCRIPCIÓN DE LOS EQUIPOS DE ACCESO

Se cuenta con un switch principal que realiza la troncalización y un switch secundario que realiza la distribución de la red. Además se tienen tres ruteadores asignados a diferentes redes y funciones específicas.

#### 2.2.1.1 Switch Principal

Es un switch Cisco Catalyst 2960 que soporta interfaces FastEthernet, GigabitEthernet, con módulos de expansión GBIC[31]; el cual es ser utilizado como dispositivo de troncalización de puertos e interconexión de datos con las empresas de última milla CNT y MEGADATOS.

[31] GBIC: Gigabit Interface Converter: Es un módulo transmisor-receptor que convierte señales eléctricas en ópticas.

El equipo establece una configuración lógica de capa enlace con CNT a través de VLAN, cuyos valores asignados para Telydata son 2102, 304 y 310, con los cuales se gestiona también el acceso a Internet de los usuarios. También soporta la tecnología PoE[32].

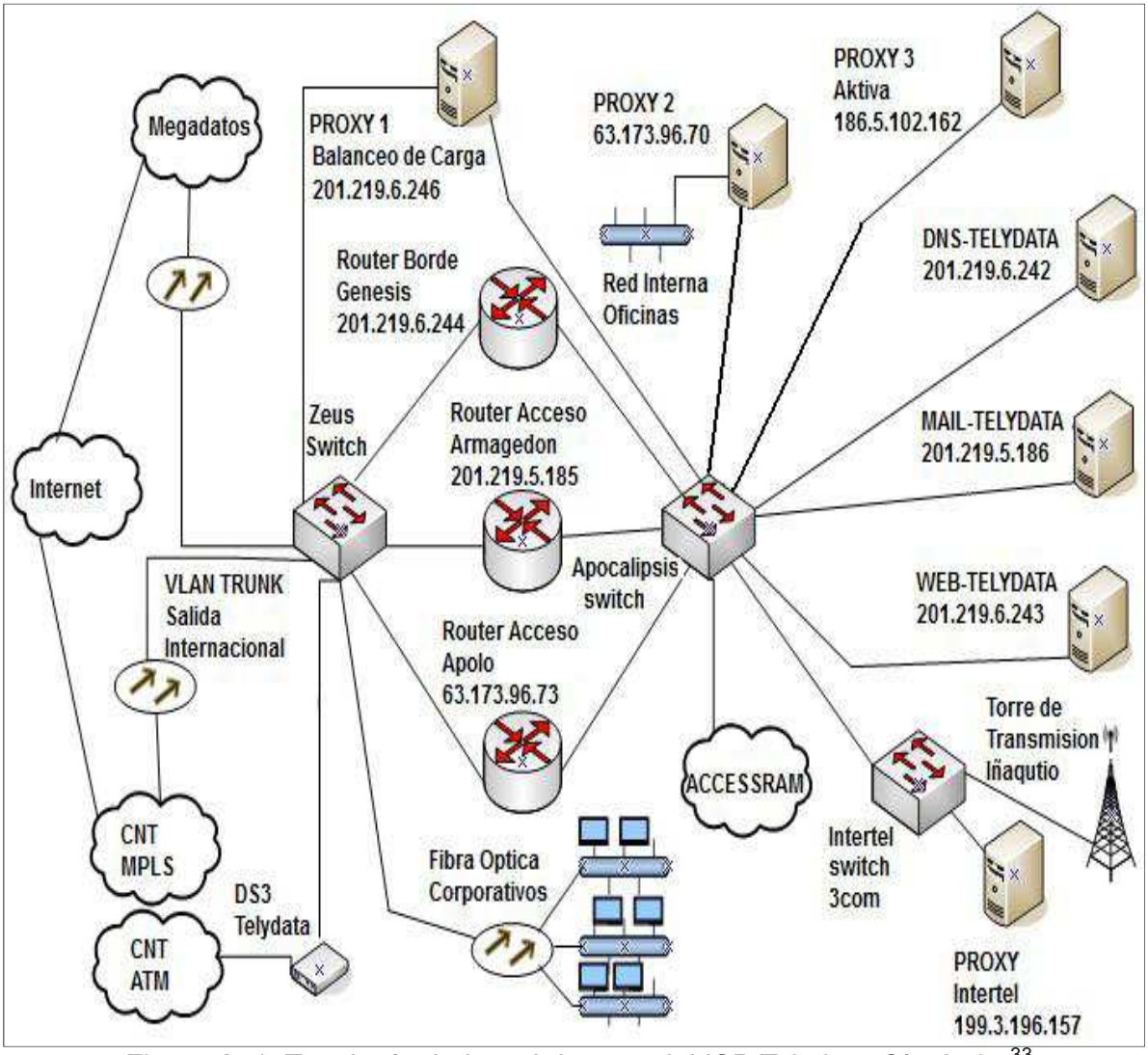



Figura 2. 1: Topología de la red de core del ISP Telydata Cía. Ltda.[33]

[32] PoE (Power over Ethernet): es una tecnología que incorpora alimentación eléctrica a una infraestructura estándar

[33] Fuente: Ing. Diego Padilla, Gerente Técnico de Telydata Telecomunicaciones y Datos Cía. Ltda.

### 2.2.1.2 Router Principal A (Génesis)

Es un enrutador Cisco 3745 de borde, mediante el cual se interconecta a los clientes del ISP con la infraestructura de CNT usando tecnología ADSL.
También realiza su interconexión con el carrier Megadatos de cobertura nacional. A través de este equipo se obtiene la interconexión del 75% de clientes entre residenciales y corporativos. Además, este equipo permite compartir la infraestructura con otros ISP, como lo son Intertel, Aktiva y Telynet. Para la salida internacional CNT-Telydata, la capacidad máxima que soporta en la red de Telydata es de 53 Mbps.

### 2.2.1.3. Router Principal B (Armagedon)

Es un enrutador Cisco 3640, mediante el cual se mantiene la interconexión con el proveedor Megadatos. A través de éste enrutador se maneja una capacidad de 18 Mbps, cumpliendo con el 10% de la demanda entre usuarios residenciales y corporativos.

### 2.2.1.4 Router C (Apolo)

Es un enrutador Cisco 3745 que opera tanto con enlaces de última milla ADSL como enlaces inalámbricos. También comparte tráfico con el ISP Intertel. Este dispositivo mantiene el enrutamiento principal para los enlaces inalámbricos. Se maneja una capacidad máxima de 22 Mbps, cumpliendo con el 15 % de la demanda de usuarios.

### 2.2.1.6 Acceso de última milla inalámbrico

El acceso está implementado a través de la tecnología propietaria de *Ubiquiti Networks*, la cual opera bajo el estándar IEEE 802.11n y el sistema MIMO [34] para la transmisión digital de datos. Se tienen instaladas principalmente cinco torres de

[34] MIMO (*Multiple Input Multiple Output*): Tecnología usada en WiFi para comunicaciones digitales, en la que permite utilizar varios canales a la vez para enviar y recibir datos en base a la incorporación de arreglos de antenas.

transmisión en toda la ciudad de Quito, en los sectores de: Iñaquito, La Planada, Roldós, Loma de Puengasí y Camal Metropolitano. Tanto el ruteador Cisco 3745(Apolo) como la radio base ubicada en Iñaquito forman el nodo principal del WISP que abastece el servicio a todos los usuarios. (Ver Figura 2.2.)

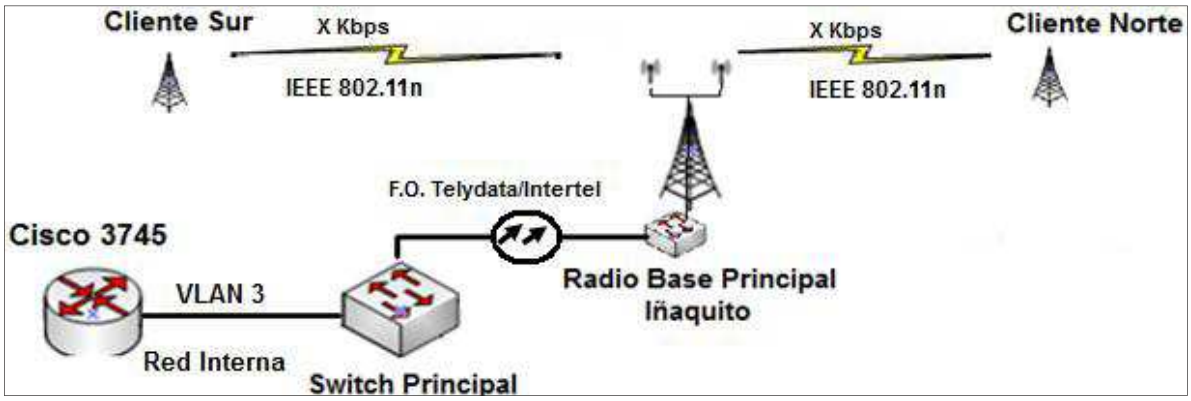



Figura 2. 2: Red de Acceso de Clientes Inalámbricos

## 2.3 PLATAFORMA INALÁMBRICA UTILIZADA

Se utiliza la plataforma Ubiquiti Networks, que cumple con el estándar 802.11n, en las frecuencias cuyo rango está comprendido en 5.2 GHz- 5.9 GHz, y en 2412-2462 MHz. Dicha plataforma es muy usada en nuestro medio, especialmente por varios WISP, ya que logra mejorar problemas por distancias en los enlaces inalámbricos y su capacidad de transmisión en una red WLAN. Además, es una plataforma que ofrece buen rendimiento y mejoras respecto a latencia y escalabilidad. Su estructura está básicamente basada por una tecnología propietaria denominada *Airmax.*

### 2.3.1 DESCRIPCIÓN GENERAL DE LA TECNOLOGÍA INALÁMBRICA AIRMAX

Está desarrollada básicamente por dos componentes importantes:

- Protocolo propietario TDMA[35] el cual se usa para sincronización de los diferentes CPE (dispositivos en el lado de los clientes).

[35]TDMA (Time Division Multiple Access): Es una técnica que permite la transmisión de señales digitales en donde se ocupa un canal (normalmente de gran capacidad) de transmisión a partir de diferentes fuentes durante una fracción del tiempo total.

- Tecnología MIMO se basa en la utilización de varias antenas para transmitir múltiples flujos de datos en un mismo periodo de tiempo. La fortaleza de *Airmax* es el uso de MIMO 2X2 que se refiere a dos antenas en el lado del emisor conectándose con dos antenas en el lado del receptor.

### 2.3.2 DESCRIPCIÓN BREVE DEL FUNCIONAMIENTO

Al transmitir señales digitales usando MIMO 2X2 se realiza la emisión de datos a través de dos antenas quienes generan dos flujos diferentes de señales codificadas individualmente, lo que se conoce como flujo espacial en paralelo. Esta propiedad permite conseguir un transporte mayor de datos en un canal determinado. En el extremo emisor se codifican y multiplexan las señales para su envío, mientras que en el lado del receptor, se decodifica y demultiplexa cada flujo de señales como se esquematiza en la Figura 2.3.

Al mismo tiempo entra en operación un sistema TDMA, el cual divide el tiempo de transmisión en múltiples ranuras de tiempo para cada radio CPE (estación receptora) que está enlazado a un AP principal.

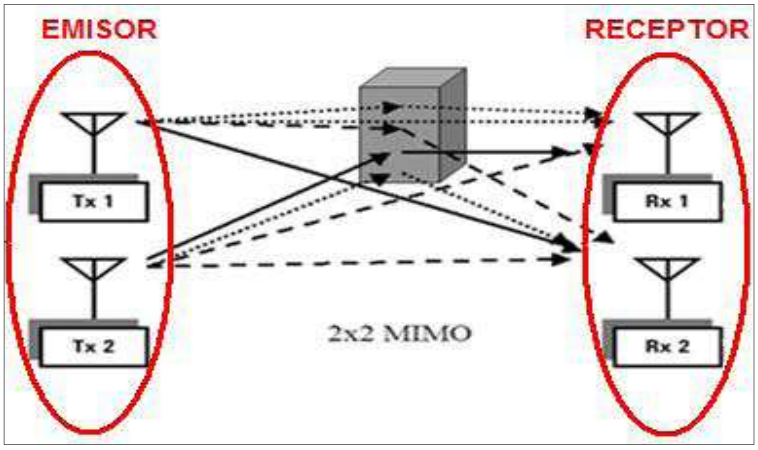



Figura 2. 3: Tecnología MIMO [18]

Las ranuras de tiempo están dinámicamente asignadas por el AP principal a cada radio CPE, para permitir la transmisión y recepción de paquetes desde o hacia el AP. Si un radio CPE está enviando o recibiendo una gran cantidad de datos, debe hacerlo dentro de sus espacios de tiempo asignados. A cada radio CPE se le asigna también una prioridad; si la red llega a estar congestionada y todo el tiempo llega a ser usada, el AP dará más ranuras de tiempo para obtener una prioridad una más alta.

El sistema *Airmax* hace que una pequeña cantidad de tiempo esté disponible para que un CPE pueda iniciar el uso del canal sin necesidad de esperar grandes periodos de tiempo, y elimina en lo posible la necesidad de forzar la interrupción de transmisiones existentes.

Según varias entidades que han utilizado este sistema [19] aseguran que se puede tener hasta 30 clientes por AP, manteniendo un buen rendimiento con cada uno de ellos.

### 2.3.3 LISTA DE DISPOSITIVOS AIRMAX SERIES M [20]

Los principales productos *Airmax* del fabricante *Ubiquiti Networks* utilizados por Telydata son aquellos que tienen la nomenclatura M incorporada:

- *Rocket M5*: Adecuados para estaciones base y enlaces formados a gran distancia, tanto para enlaces punto a punto como punto multipunto. Operan sobre la frecuencia de los 5.8 GHz, estos dispositivos son utilizados para la interconexión con el backbone.

- *Nanostation M*: los que poseen serie M2 funcionan para las frecuencias de operación en la banda de de 2.4 GHz. Los productos que poseen serie M5 funcionan para las frecuencias de operación en la banda de 5.8 GHz. Válido para enlaces punto a punto y punto multipunto. Utilizado generalmente a medianas distancias en el lado del CPE.

- *NanoBridge M5*: para frecuencias de operación sobre los 5 GHz, utilizado generalmente como radios CPE a grandes distancias.

- *AirGrid M*: válido para frecuencias de operación tanto en 2.4 GHz como en 5.8 GHz, esta no posee la característica 2X2 MIMO. Usada en el lado CPE.

### 2.3.4 PRINCIPALES PARÁMETROS TOMADOS EN CUENTA EN EL ENLACE INALÁMBRICO [21]

Los principales parámetros tomados en cuenta para garantizar el buen estado en un enlace son:

- *SSID de Estación Base*: Se refiere al nombre de la red inalámbrica, dado por el punto de acceso principal.

- *Modo Inalámbrico*: Muestra el modo de funcionamiento inalámbrico del dispositivo. AirOS v5[36] soporta los modos de operación: *Punto de Acceso* y *Estación.*

- *MAC del AP (AP MAC)*: muestra la dirección MAC del punto de acceso donde el dispositivo está asociado mientras opera en modo estación.

- *Intensidad de la Señal (Signal Strenght):* Medida en dBm, muestra el nivel de la señal inalámbrica recibido en el lado del receptor o cliente, mientras opera en modo estación. Depende de varios factores, entre ellos de la alineación entre ellos. Según el fabricante un nivel de señal superior a -85 dbm es recomendado para un enlace estable [22]. Para el caso puntual de Telydata se ha experimentado que un enlace es estable cuando su nivel de señal se encuentra en el rango de -40 dBm y -70 dBm[37], basándose en la idea de que

[36] AirOS v5: sistema operativo propietario de *Ubiquiti Networks* para la administración de los dispositivos inalámbricos

[37] Información provista por el Sr. José Vásquez, Departamento Técnico de Telydata.

mientras más alta sea la potencia de la señal sobre el ruido, mejor calidad tendrá el enlace.

- *Ruido Base (Noise Floor):* Muestra el nivel actual de ruido, medido en dBm.

- *Pausa ACK (ACK Timeout):* Muestra el actual valor de *timeout* (tiempo de espera) para los cuadros ACK[38]. El valor de ACK puede ser especificado manualmente o ajustable automáticamente. Esto especifica cuanto debe esperar el dispositivo AirOS por un acuse de recibo por parte del otro dispositivo confirmando la correcta recepción del paquete de datos antes de que el paquete sea considerado erróneo y deba ser reenviado. Es muy importante para los parámetros de rendimiento en enlaces inalámbricos en el exterior.

- *Tasa de Recepción y Envío (Tx Rate and Rx Rate):* Muestra la máxima tasa de transmisión mientras el equipo opera en modo estación. Dicha tasa de transmisión generalmente depende del tipo de modulación, velocidades de transmisión e intensidad de señal que se esté utilizando.

- *Canal/Frecuencia (Frequency):* Hace referencia a la frecuencia que opera el punto de acceso al que se encuentra conectado un dispositivo (cliente) para el envío y recepción de datos. El rango de frecuencias depende de las regulaciones locales.

- *Ancho del Canal (Channel Width):* Es la ancho del canal de radio usado por el dispositivo inalámbrico. Los rangos disponibles son 5 MHz, 10MHz, 20 MHz y 40 MHz. En modo estación la opción por defecto es 20/40 MHz.

- *CCQ de Transmisión (Transmit CCQ):* Este es un índice de cómo se evalúa la calidad de la conexión del cliente inalámbrico. Se toma en cuenta el conteo de

[38] ACK (Acknowledgement): es una señal emitida por un proceso o equipo de comunicación para emitir una respuesta a otro proceso o equipo de comunicación en un protocolo dado.

errores de transmisión, latencia y rendimiento. Evalúa la tasa de paquetes correctamente transmitidos en relación con los que deben ser transmitidos y tiene en cuenta la actual tasa en relación con la mayor tasa especificada.

- *Máxima Tasa de Transmisión (Max. Rate):* Especifica el tipo de modulación usada para la comunicación digital y la máxima velocidad a la que pueden ser enviados los datos.

### 2.3.5 DESCRIPCIÓN DEL BACKBONE INALÁMBRICO

Telydata Cía. Ltda., tiene su nodo de comunicaciones inalámbricas principal ubicado sobre la terraza del edificio Centro de Oficinas Iñaquito I, situado en las calles Jorge Drom y Arizaga, sector Iñaquito. Sobre este se encuentra una torre de transmisión de 4 metros. A partir de dicho nodo se distribuye el backbone inalámbrico hacia el norte y el sur de la ciudad. Actualmente, no se tiene información debidamente esquematizada, documentada, ni actualizada, del backbone inalámbrico tanto para el norte como sur de la ciudad.

## 2.4 REQUERIMIENTOS DEL DEPARTAMENTO TÉCNICO PARA LA OPERATIVIDAD DE LA RED

Para establecer información más detallada respecto al estado actual y problemas que afectan en la operatividad de la red, se realiza una entrevista al Ing. Diego Padilla, gerente técnico de la compañía. A continuación se resume lo expuesto:

- “Mientras más eficiente sea el departamento técnico al momento de solucionar una incidencia en la red, la disponibilidad y servicio a los clientes será mejor. Esto tiene que ver con el grado de conocimiento que se tenga de nuestra propia infraestructura, datos y comportamiento actual”

- “Al momento, la red no se encuentra debidamente documentada, ya que en los últimos tres meses ha estado en un proceso de crecimiento y

mejoramiento constante, apoyado por la gran demanda que tiene el mercado del servicio de Internet inalámbrico en las áreas rurales. La base de información de usuarios que se tiene actualmente depende únicamente del sistema CRM (Customer Relationship Management). El problema de este sistema es que muchas veces está desactualizado y no tiene la información necesaria. Inicialmente el CRM se creó para almacenar datos técnicos generales de usuarios con enlaces ADSL, no inalámbricos. Al momento de realizar soporte técnico, no se cuenta con los datos de los equipos y parámetros de los enlaces pertenecientes a cada usuario. Esto hace que el técnico de Telydata demore con la solución de un problema en específico."

- "Adicionalmente se tienen dos problemas que influyen directamente en el servicio y operatividad de la red, son los siguientes:

  a) No se ha realizado ningún tipo de gestión sobre la capacidad del canal de la RIT que permita asegurar que los usuarios puedan navegar a una velocidad máxima contratada. Ese tipo de control no se lo ha implementado aún en alguno de los equipos del core de la red.

  b) No se cuenta con un reinicio programado de antenas. Esto es recomendado por el fabricante, para mejorar el rendimiento de las mismas. También es recomendable desde el punto de vista técnico-experimental, ya que si no se lo realiza podría ocasionar indisponibilidad del servicio."

### 2.4.1 FACTORES TÉCNICOS QUE INTERVIENEN EN EL SOPORTE TÉCNICO

Con el fin de conocer cuál es la información técnica que se requiere almacenar en una base de información, se toma en cuenta los aspectos técnico-experimentales y aquellos recomendados por el fabricante para el desempeño correcto del enlace inalámbrico, y por ende, del servicio entregado a los clientes finales. Para garantizar

una buena calidad en la transmisión de datos, se necesita conocer constantemente de las estaciones de los clientes, los siguientes puntos básicos:

- *Nivel de la señal: (Signal Strenght):* en los dispositivos pertenecientes al backbone inalámbrico se tienen valores entre -40 dBm y 55 dBm[39] . En los equipos pertenecientes a los usuarios se tienen valores entre -59 dBm y -75dBm[39]. (Dichos valores son logrados por el Departamento Técnico de la empresa al trabajar con la implementación de la Plataforma).

- *Calidad de Conexión (CCQ):* en los dispositivos pertenecientes al backbone inalámbrico se tienen valores de 95 % o superiores. En los equipos pertenecientes al usuario se tienen valores de 90 % o superiores[40].(Dichos valores son logrados por el Departamento Técnico de la empresa al trabajar con la implementación de la Plataforma).

- *Frecuencia* (*Frequency*): debe estar seleccionada en un canal que no presente interferencias por enlaces vecinos. Generalmente se encuentran en valores desde 5200 MHz hasta 5805 MHz[41], depende de la antena con la que se trabaje.

- *MAC Address (WLAN MAC):* es la dirección MAC de una antena. Con dicha dirección, se puede verificar si una antena está enganchada al AP principal.

Al momento de realizar la instalación y configuración de las antenas en cada cliente, se necesita conocer y almacenar principalmente la información anteriormente expuesta en una base de datos para mantenerla como referencia al momento de realizar el soporte técnico.

[39] Son mejores valores a lo recomendado por el fabricante, ya que este estima un valor de -85 dBm
[40] Valores que se basan en la recomendación del fabricante, el cual establece un valor de 100 %
[41] Valores de frecuencia que los utiliza el fabricante *Ubiquiti Networks,* basándose en IEEE 802.11n

## 2.5 ANÁLISIS DE TRÁFICO

Se realiza primero la monitorización de la capacidad del canal para cada AP principal, y luego se procede a la determinación del tráfico para cada red. Para ello se utiliza los siguientes componentes:

- CACTI
- Módulo Netflow Data Export, perteneciente a la plataforma Cisco Release 12.2.
- Herramienta Orion NPM (*Network Performance Monitor*) con el módulo NTA (*Netflow Traffic Analyzer*)

### 2.5.1 HERRAMIENTAS DE MONITOREO

#### 2.5.1.1 Orion Network Packet Manager [23]

Es una herramienta desarrollada y distribuida por Solarwinds, Inc., para el análisis de la red. El módulo más utilizado es el Orion NTA (*Netflow Traffic Analyzer*), el cual convierte datos estándar de tráfico en gráficos y tablas comprensibles que indican el funcionamiento y uso de una red determinada.
El funcionamiento básico está basado en la recopilación y almacenamiento de datos sobre el estado utilizando los protocolos ICMP, SNMP, y Syslog. Dicha información se almacena un una base SQL abierta para poderla acceder de una manera sencilla.

*a) Principales Ventajas*

- Interfaz web fácilmente utilizable que permiten que la información pueda ser personalizable en función de las necesidades del usuario.
- Decodifica la mayor parte de protocolos estándar.
- Tiene integración abierta con diferentes estándares, como Microsoft SQL Server y las MIB.

- Provee más de 40 informes históricos detallados con los cuales se puede proyectar tendencias futuras; reportes de distribución de protocolos más usados, mayores consumidores de la red y conversaciones más representativas de dispositivos.

*b) Requisitos*

El distribuidor del software recomienda instalarlo en un servidor Windows con una base de datos alojada independientemente, en su propio SQL Server. En la Tabla 2.1 se indican los requisitos mínimos de un servidor Orion NPM.

| Software | Requisitos |
|---|---|
| Sistema operativo | Windows Server 2003 o 2008, incluido R2, con IIS en modo de 32 bits. El IIS debe estar instalado. Solarwinds recomienda que los administradores de Orion NPM dispongan de privilegios de administrador local para garantizar la completa funcionalidad de las herramientas locales de Orion NPM. Solarwinds no admite las instalaciones de Orion NPM en Windows XP, Windows Vista SP2 o Windows 7 en entornos de producción. |
| Servidor web | Microsoft IIS, versión 6.0 y superior, en modo de 32 bits. |
| .NET Framework | Versión 3.5. Se recomienda .NET Framework 3.5 SP1. |
| SNMP Trap Services | Componente de herramientas de supervisión y administración del sistema operativo de Windows. |
| Web Console Browser | Microsoft Internet Explorer versión 6 o superior con secuencia de comandos Firefox 3.0 o superior |

Tabla 2. 1: Requisitos de hardware para un servidor Orion NMP

*c) Instalación y configuración* [24]

Se instala los dos componentes de Orion (NPM y NTA) en un sistema Operativo Windows Server 2008 que cumple con los requisitos de la Tabla 2.1. Para la configuración de Netflow en el ruteador Apolo, se especifica la dirección al servidor 199.3.196.158, puerto 5790 que servirá de colector de información. La fuente de datos es la correspondiente a la interface VLAN 3, ya que es la que soporta la red

inalámbrica de Telydata Cía. Ltda. Cabe mencionar que en esta interfaz también existen otras dos redes para clientes con última milla ADSL, pero se hará énfasis únicamente en la red inalámbrica. La representación lógica de red para el monitoreo se muestra en la Figura 2.4.

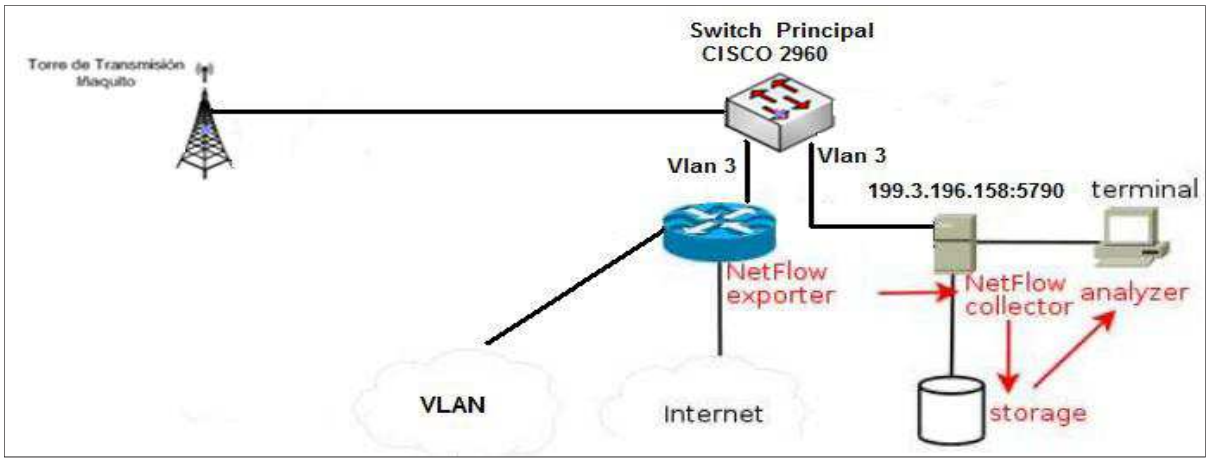



Figura 2. 4: Instalación y configuración de la herramienta Netflow Traffic Analyzer Solarwinds

En la Figura 2.5 se indica la configuración de la interfaz VLAN 3 en el ruteador Apolo, configurando los flujos de ingreso y egreso. En la Figura 2.6 se determina que el servidor colector de datos tendrá la dirección IP 199.3.196.158 y el puerto 5907.

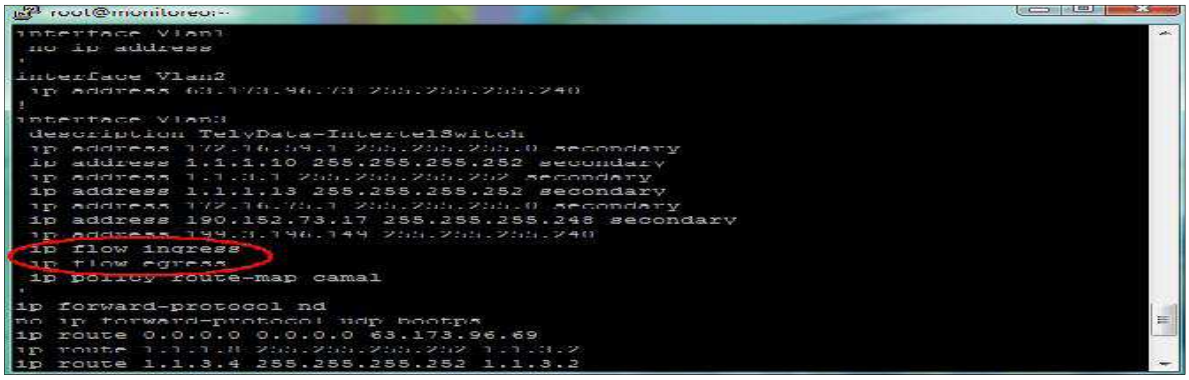



Figura 2. 5: Configuración de la interfaz VLAN 3 con Netflow

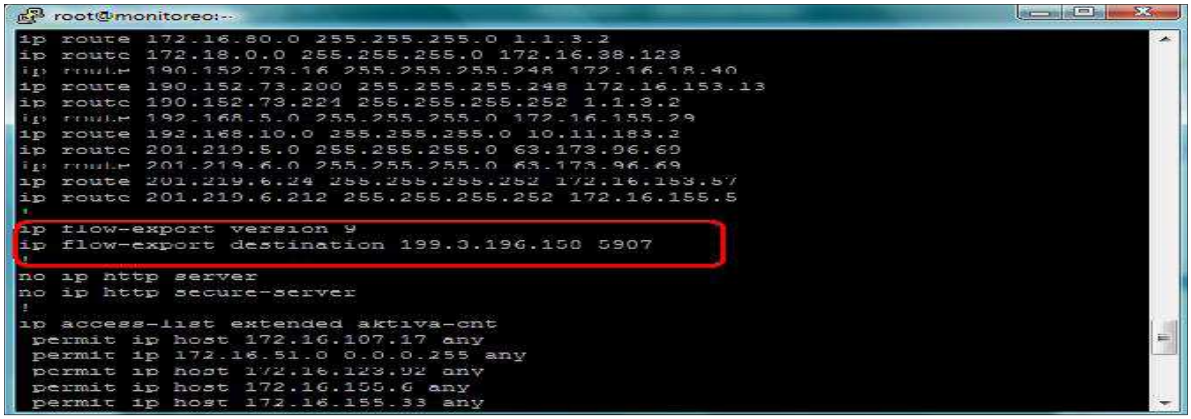



Figura 2. 6: Asignación de la dirección IP y puerto del Colector Netflow

### 2.5.1.2 CACTI [25]

Es una aplicación que inicialmente se estableció bajo la licencia GNU/GPL para el monitoreo del uso de canal de transmisión, así como del rendimiento de una red. Se basa en la herramienta RDDtool para el almacenamiento de datos y funcionalidades gráficas. Provee a los administradores lecturas casi en tiempo real y a largo plazo de los dispositivos. Se puede emplear también para monitoreo de otros aspectos de una red tales como utilización de memoria y CPU.

*a) Principales Ventajas*

- Brinda diagnósticos del rendimiento de una red compuesta de servidores, ruteadores, switches y varios dispositivos que soporten SNMP v1 en adelante.
- Interfaz gráfica organizada, ya que representa dispositivos de red en grupos en diferentes tipos de vistas, ya sea en listas o en forma de árbol, además del soporte que brinda a los plantillas de dicha interfaz.
- Manejo de usuarios, puede administrar usuarios en base a permisos de ciertas áreas de la aplicación
- Ayuda a mejorar la calidad de servicio de manera proactiva, ya que provee estadísticas de latencias y porcentajes de disponibilidad.

*b) Requisitos*

Cacti está disponible tanto para plataformas Windows como Linux. Tomando en cuenta la plataforma Linux, que es donde se realizará la instalación, la versión 0.8.7i básicamente tiene cuatro requisitos importantes:

- Servidor PHP 5.5
- MySQL versión 4.1.18 o mayor, como su base de datos
- RDDtool versión 1.4.5 o mayor
- NET-SNMP

c) *Instalación y Configuración* [26]

La aplicación ya se encuentra instalada en el servidor de la empresa ns1.telydata.net, con la siguiente dirección web: http://201.219.6.242/cacti/index.php, sin embargo, se configuran las comunidades SNMP v1 para monitorear algunos de los puntos de acceso inalámbrico de la red.

### 2.5.2 ESPECIFICACIÓN DE LA CAPACIDAD DEL CANAL EN EL BACKBONE DE LA RIT DADO POR EL FABRICANTE UBIQUITI NETWORKS

Bajo las condiciones con las que se encuentra instalado y configurado el enlace inalámbrico, el fabricante señala que se puede alcanzar una capacidad máxima de 26 Mbps. Teóricamente los 26 Mbps circularían independientemente tanto para la red norte, como para la red sur. La Figura 2.7 es una toma de pantalla del AP principal Norte. De igual manera se ha comprobado con el AP principal del Sur.

Además, se corroboró el valor de capacidad del canal, tomando en cuenta que el backbone de la red cuenta con los datos de configuración presentados en la Tabla 2.2.

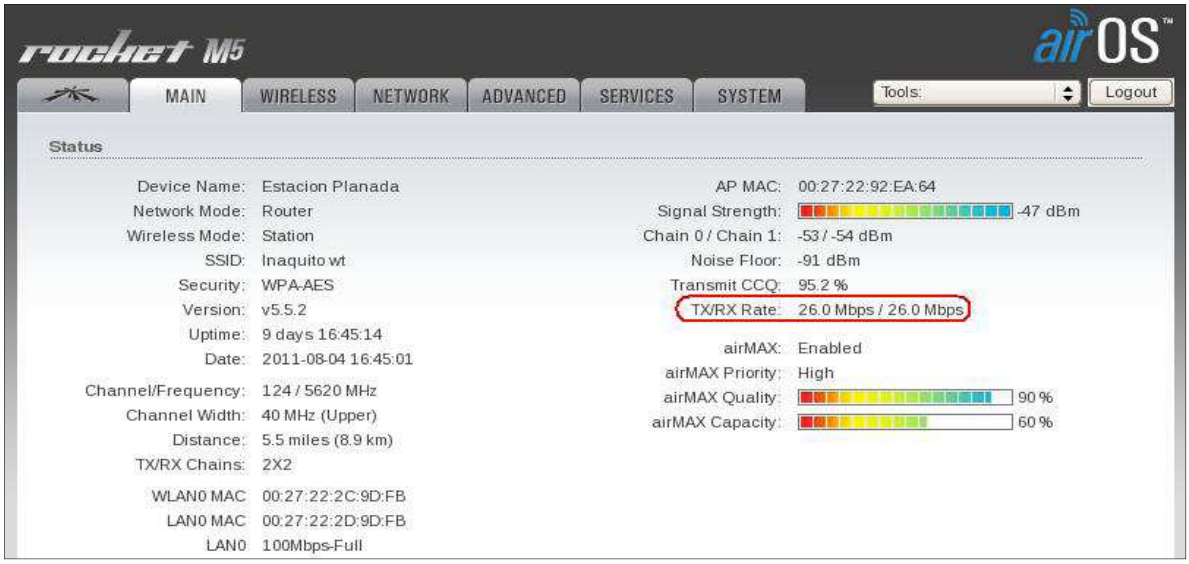



Figura 2. 7: Velocidad de transmisión y recepción del AP Principal Planada

| Parámetro | Dato | Asignación de la capacidad del canal |
|---|---|---|
| **Modo de operación** | MIMO 2X2, *Airmax* | Según la Tabla de velocidades establecidas por *Ubiquiti Networks* del **Anexo 2**, se tiene un máximo de velocidad de transmisión de **26 Mbps** |
| **Antena *Ubiquiti* utilizada** | Rocket Dish M5 | |
| **Modulación Digital** | MCS 9-26, QPSK | |
| **Ancho de Banda Utilizado** | 20 MHz | |

Tabla 2. 2: Capacidad de acceso en el Backbone de la RIT

La velocidad de transmisión depende básicamente de dos parámetros: del ancho de banda utilizado y de la modulación establecida en la configuración del enlace. Respecto a la terminología, el índice MCS[42] especifica el conjunto de los siguientes parámetros:

- *El número de flujos espaciales*: MIMO 2X2, que significa dos flujos de emisión con dos flujos de recepción.

[42] MCS: Índice utilizado por *Ubiquiti Networks* que determina el numero de flujos espaciales, la modulación, la velocidad de codificación y los valores de la velocidad de datos

- *La modulación digital utilizada para la transmisión de datos:* la cual que puede ser BPSK, QPSK, 16QAM, 64QAM.

### 2.5.3 MEDICIÓN DE LA CAPACIDAD DEL CANAL EN EL BACKBONE DE LA RIT

Para la medición de capacidad del canal se usa la herramienta CACTI, la cual toma información de los distintos dispositivos inalámbricos, a través del protocolo SNMPv1. Los AP principales de la red inalámbrica son:

- AP Planada (IP: 172.16.33.1)
- AP Iñaquito Sur (IP: 1.1.3.2)
- AP del camal metropolitano (IP:172.16.41.1 / Alias: 172.16.61.1)

Cabe recalcar que la especificación de la red Iñaquito Sur contiene la red Quito Centro, ya que se toma como punto de referencia geográfico al sector de Iñaquito.
Durante un monitoreo realizado, se toma como referencia el mes de abril, ya que en este mes se representa mayor actividad de los enlaces inalámbricos respecto a los anteriores meses por tener más usuarios.
Se toma como referencia la semana del lunes 23 de abril del 2012 al viernes 27 de Abril del 2012, ya que durante el mes monitoreado se ha encontrado comportamiento similar a las anteriores semanas. Al comparar los días de mayor carga se verifica que los lunes, miércoles y viernes son los que representan mayor consumo por parte de los clientes. Por lo tanto, en las Figuras 2.8, 2.9 y 2.10 se presentan los gráficos para la red Iñaquito Norte; y en las Figuras 2.11, 2.12 y 2.13 se presentan los gráficos para la red Iñaquito Sur.

Una vez tomadas las muestras se procedió a compararlas por días, definiendo así el comportamiento que se presenta en el canal. La fuente donde se toma los gráficos es: http//201.219.6.242/cacti/index.php.

**Observaciones:**

Enlace Iñaquito Norte

- Tanto en la Figura 2.8, Figura 2.9 y Figura 2.10 se verifica un comportamiento del canal asimétrico. El tráfico de entrada es por lo menos cuatro veces mayor que el de salida.

- El pico máximo para la Figura 2.8 es 6.42 Mbps, para la Figura 2.9 es 6.88 Mbps y para la Figura 2.10 es 6.64 Mbps, en donde se denota también que el tiempo de saturación se comprende en dos periodos: de 9h00 a 13h00 y de 14h00 a 22h00.

- Se verifica que el AP nunca deja la actividad, ya que sigue pasando tráfico desde las 24h00 durante la madrugada del siguiente día.

Enlace Iñaquito Sur

- Tanto en la Figura 2.11, Figura 2.12 y Figura 2.13 se verifica un comportamiento del canal asimétrico. El tráfico de entrada es por lo menos cuatro veces mayor que el de salida.

- El pico máximo para la Figura 2.11 es 7.93 Mbps, para la Figura 2.12 es 7.02 Mbps y para la Figura 2.13 es 6.77 Mbps, lo que indica que existe mayor demanda del canal respecto al enlace Iñaquito Norte.

- Se verifica en las figuras mencionadas que no siempre existe tráfico durante la madrugada.

De acuerdo a la Tabla 2.2, cada enlace tiene una capacidad de acceso máxima de 26 Mbps; sin embargo, se ha llegado a un valor máximo de 7.93 Mbps (Figura 2.11), equivalente al 31 % del canal ocupado.

### 2.5.3.1 Iñaquito-Norte

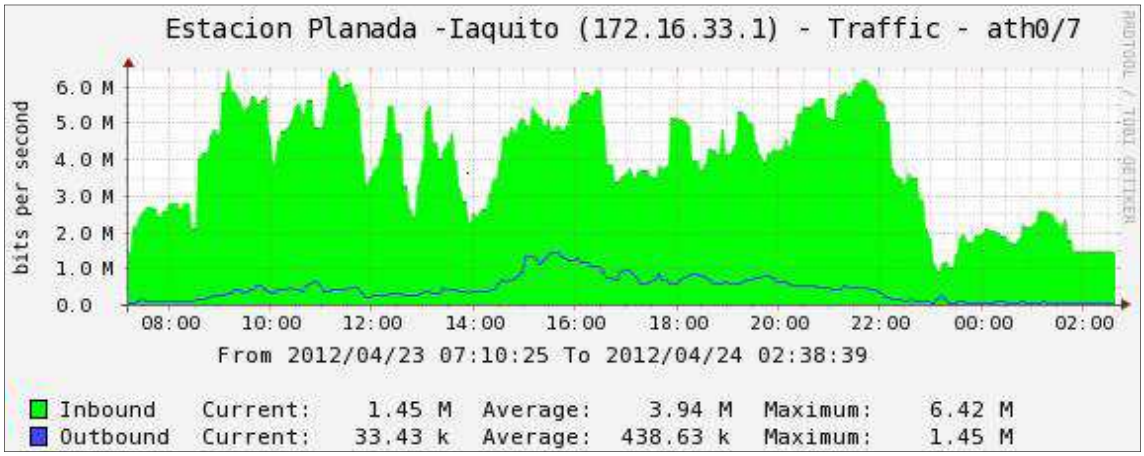



Figura 2. 8: Capacidad del canal RIT Norte. Lunes 23 Abril del 2012

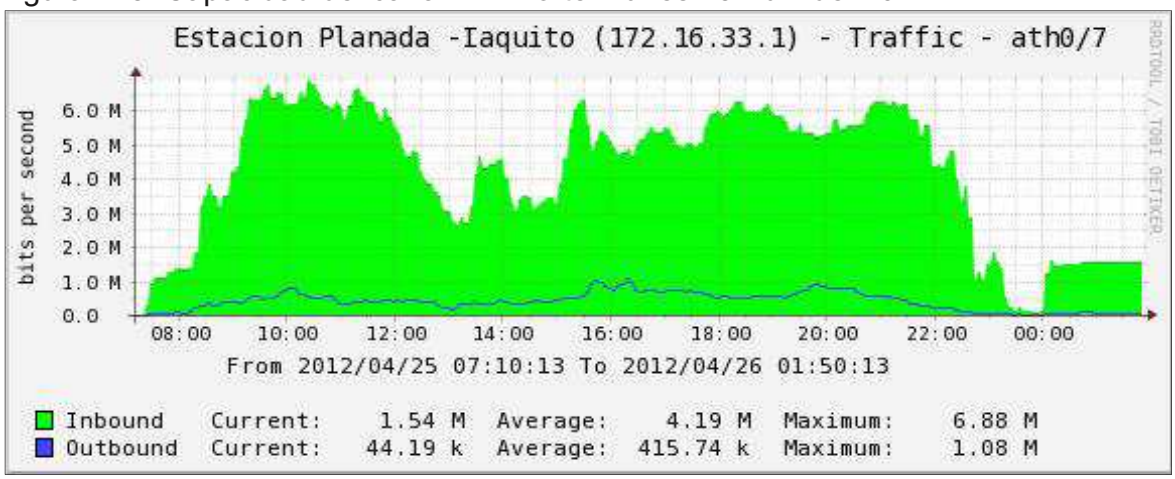



Figura 2. 9: Capacidad del canal RIT Norte. Miércoles 25 de Abril del 2012

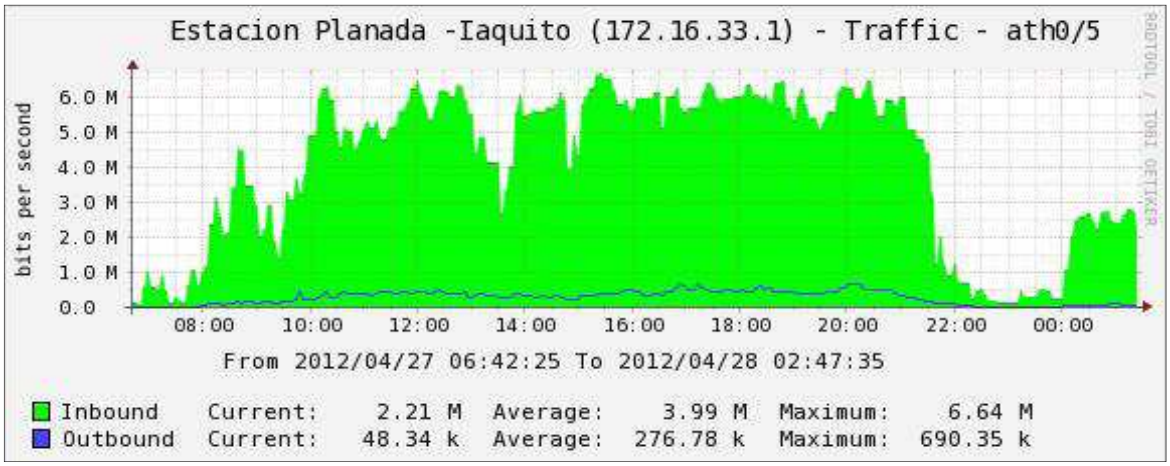



Figura 2. 10: Capacidad del canal RIT Norte. Viernes 27 de Abril del 2012

### 2.5.3.2 Enlace Iñaquito-Sur

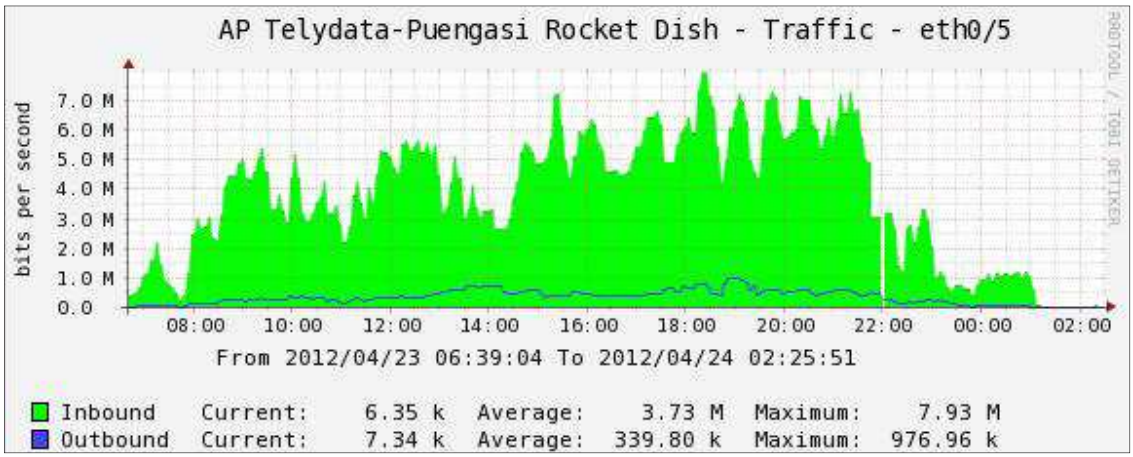



Figura 2. 11: Capacidad del canal RIT Sur. Lunes 23 de Abril del 2012

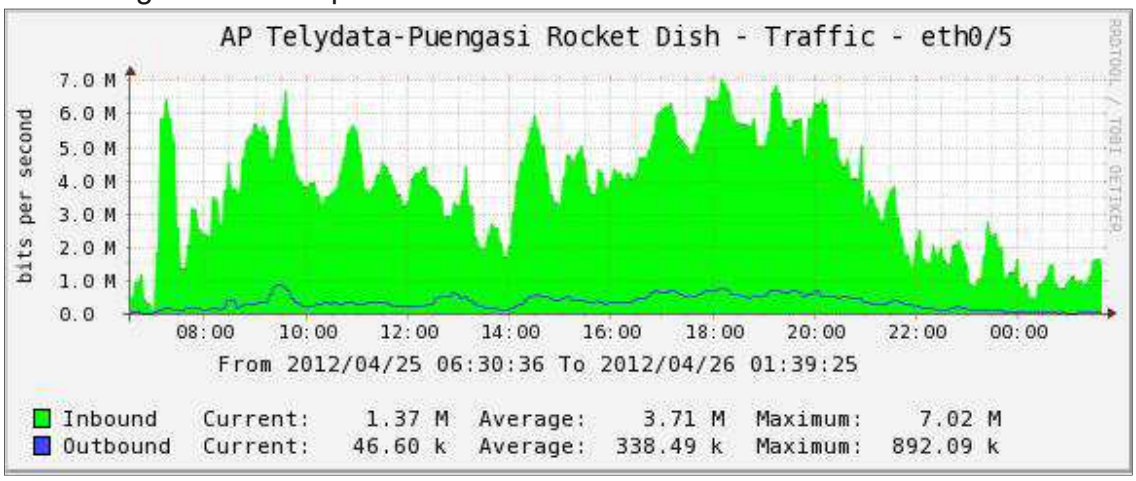



Figura 2. 12: Capacidad de canal RIT Sur. Miércoles 25 de Abril del 2012

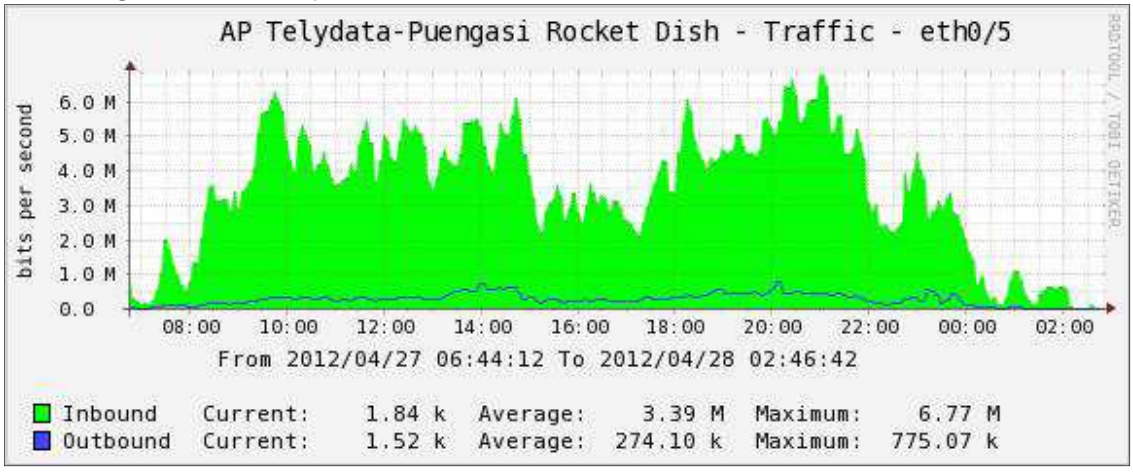
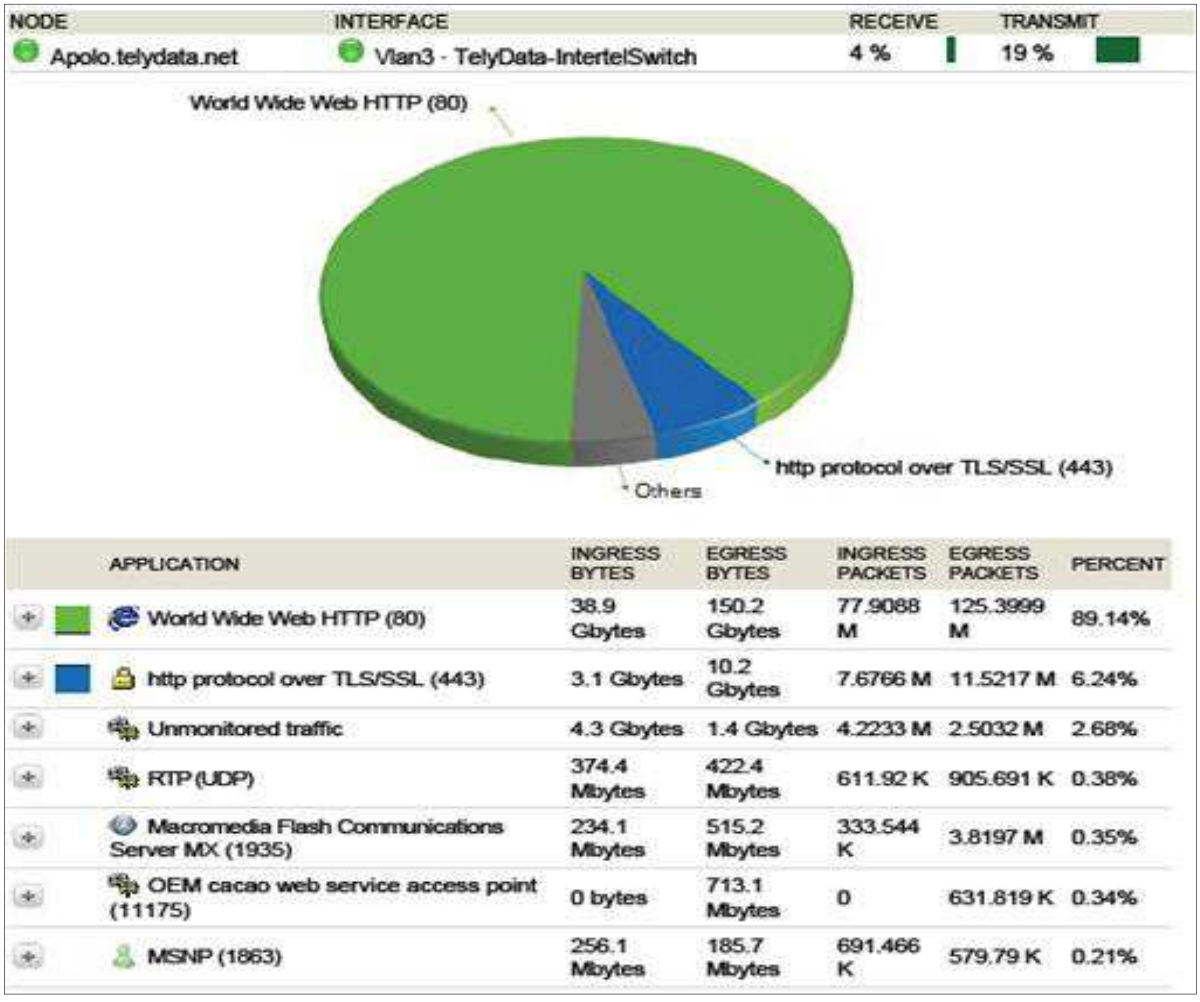



Figura 2. 13: Capacidad del canal RIT Sur. Viernes 27 de Abril del 2012

## 2.6 DESCRIPCIÓN GENERAL DEL TRÁFICO

En la Figura 2.14 se indica una representación diaria de las aplicaciones que circulan a través de la red, tomado de la VLAN 3 del Router de Acceso Apolo, que es la asignada para la última milla inalámbrica dentro de la RIT, y se indica el respectivo porcentaje de uso en cada protocolo o aplicación.




| APPLICATION | INGRESS BYTES | EGRESS BYTES | INGRESS PACKETS | EGRESS PACKETS | PERCENT |
|---|---|---|---|---|---|
| World Wide Web HTTP (80) | 38.9 Gbytes | 150.2 Gbytes | 77.9088 M | 125.3999 M | 89.14% |
| http protocol over TLS/SSL (443) | 3.1 Gbytes | 10.2 Gbytes | 7.6766 M | 11.5217 M | 6.24% |
| Unmonitored traffic | 4.3 Gbytes | 1.4 Gbytes | 4.2233 M | 2.5032 M | 2.68% |
| RTP (UDP) | 374.4 Mbytes | 422.4 Mbytes | 611.92 K | 905.691 K | 0.38% |
| Macromedia Flash Communications Server MX (1935) | 234.1 Mbytes | 515.2 Mbytes | 333.544 K | 3.8197 M | 0.35% |
| OEM cacao web service access point (11175) | 0 bytes | 713.1 Mbytes | 0 | 631.819 K | 0.34% |
| MSNP (1863) | 256.1 Mbytes | 185.7 Mbytes | 691.466 K | 579.79 K | 0.21% |

Figura 2. 14: Distribución de aplicaciones más usadas

Como se puede apreciar en la Figura 2.14, el protocolo que más uso tiene es el HTTP con el 89.14%. El protocolo HTTPS usa el 6.24% del tráfico. Por lo tanto, el servicio en su mayor parte es de navegación web. El 4.62 % del tráfico restante está constituida principalmente por un conjunto de aplicaciones compuestas por:

- Aplicaciones Macromedia Flash Communications Server MX(1935)
- OEM Cacao Web Service Access Point (11175)
- Protocolos RTP[43]
- Protocolo MSNP[44]

Como es de conocimiento, una aplicación puede usar varios protocolos al momento de transmitir paquetes de información. A continuación se lista una descripción de las aplicaciones y protocolos monitoreados por el gestor de tráfico:

a) HTTP [27]: Es un protocolo que se basa en el modelo de red cliente servidor, aplica el esquema solicitud respuesta para realizar transacciones web e intercambio de información. Este protocolo permite usar métodos para indicar la finalidad de una petición (GET, POST, HEAD, PUT). Utiliza el puerto 80.

b) HTTPS [28]: es el protocolo para transacciones web seguras que usa cifrado basado en TSL/SSL. Tanto SSL (*Secure Sockets Layer*) como su sucesor TSL (*Transport Layer* Security) proporcionan conexiones seguras y ayudan a prevenir ataques como man-in-the-middle[45] o eavesdropping[46]. Utiliza el puerto 443.

c) Macromedia Flash Communications Server [29]: Hace referencia a un conjunto de componentes de software que se ejecutan sobre un servidor web para proporcionar aplicaciones streaming como videos, chat, live streaming basados en el formato flash. Utiliza el puerto 1935.

d) Oem Cacao Web Service Access Point [30]: es una aplicación que utiliza el puerto 11175 registrado en IANA, representa a aquellas aplicaciones que realizan intercambio de archivos, ya sean de texto o multimedia.

[43] Ver ítem *e),* de la presente sección
[44] Ver ítem f), de la presente sección
[45] Man in the middle: técnica en el que el atacante adquiere la capacidad de leer, insertar y modificar los mensajes entre dos partes sin que ninguno de ellos conozca que el enlace ha sido vulnerado.
[46] Eavesdropping: hace referencia a escuchas ilegales en medios de información cifrados o no cifrados.

e) RTP [31]: Es un protocolo que facilita la transmisión a tiempo real de flujos multimedia. Proporciona funciones de transporte extremo a extremo de forma multicast o unicast. Algunas de las características importantes que posee el protocolo son: identificación de un tipo de dato, secuenciamiento y marca de tiempo para la sincronización del audio y video. Este protocolo es complementado con otro denominado RCTP[47] para ayudar en la organización de la entrega de datos y control sobre la conexión, ya que por sí solo no presenta mecanismos de control de errores, ni de control de flujo y generalmente se ejecuta con el protocolo UDP.

f) MSNP (1863) [32]: es el protocolo de mensajería instantánea, inicialmente adoptado por Microsoft para plataformas cliente servidor basadas en Windows. Usa tres tipos de servidores para despacho, notificación y panel de control. Usa la arquitectura TCP/IP para el envío de mensajes en forma de texto, con ello realiza tres tipos de operaciones: comandos, mensajes y errores.

Para el análisis sobre el uso de los protocolos y aplicaciones se realiza el monitoreo del trafico semanal durante treinta días desde el 14 de Abril del 2012 hasta el 16 de Mayo del 2012. Se toma como ejemplo la semana del 7 al 13 de Mayo, ya que al compararlas con las otras semanas, esta presenta mayor tráfico, lo que permite describir el tráfico sobre valores máximos encontrados. En el **Anexo 3**, se muestra las tablas de las anteriores semanas correspondientes al periodo mencionado, así como también se indica el cálculo del tráfico diario correspondiente para cada protocolo.

La Tabla 2.3 indica la capacidad total de cada protocolo sobre el canal. HTTP es el que más ocupa con un valor máximo de 18.39 Mbps para el periodo dado. Se verifica también que el día de la semana de mayor consumo es el viernes 11 de Mayo del 2012.

[47] RTCP: Real Time Control Protocol, es un protocolo de control que recoge estadísticas de una sesión streaming con el fin de coordinar la calidad de servicio proporcionada por RTP.

En la Tabla 2.4 se indica un resumen del tráfico máximo y promedio diario por cada categoría.

| Semana del 7 al 13 de Mayo del 2012 | | | | | | | |
|---|---|---|---|---|---|---|---|
| Protocolos | 07/05/12 (Mbps) | 08/05/12 (Mbps) | 09/05/12 (Mbps) | 10/05/12 (Mbps) | 11/05/12 (Mbps) | 12/05/12 (Mbps) | 13/05/12 (Mbps) |
| HTTP | 14.05 | 15.20 | 17.51 | 17.68 | 18.52 | 18.39 | 16.48 |
| HTTPS | 1.06 | 1.17 | 1.23 | 1.17 | 1.27 | 1.24 | 0.99 |
| Macromedia Flash | 0.10 | 0.07 | 0.06 | 0.12 | 0.11 | 0.07 | 0.05 |
| RTP | 0.04 | 0.05 | 0,07 | 0.07 | 0.03 | 0.08 | 0.08 |
| MSNP | 0.00 | 0.03 | 0.04 | 0.08 | 0.09 | 0.05 | 0.13 |
| Oem Web Service | 0.06 | 0.06 | 0.06 | 0.06 | 0.06 | 0.06 | 0.05 |
| Total | 16.7 | 18.1 | 20.71 | 20.96 | 21.92 | 21.71 | 19.42 |

Tabla 2. 3: Capacidad del canal por protocolo[48]

| Categoría (Protocolo y Aplicaciones) | Capacidad Máxima (Mbps) | Capacidad Promedio diario (Mbps) |
|---|---|---|
| Navegación Web HTTP | 18,52 | 16.83 |
| Navegación Web Segura HTTPS | 1.27 | 1.16 |
| Streaming de audio y video | 0.3 | 0.28 |
| TOTAL | 20.09 | 18.24 |

Tabla 2. 4: Tráfico por categoría

Como se puede denotar en los valores totales de la Tabla 2.4, posee valores aproximados a 21.15 Mbps, que es el valor de la capacidad máxima que indica el servidor de monitoreo de Telydata para la interfaz VLAN 3[49]. Cabe mencionar también que en la capacidad del canal medida por categoría (Tabla 2.4) se han

[48] La Tabla se basa de los datos de la sección *NTA Summary , Top 20Applications* del servidor Orion NMP.
[49] Leído el 19 de Mayo del 2012 en el servidor Cacti

incluido dos redes diferentes a la RIT, las cuales se toman de la VLAN 3 para la provisión de Internet en oficina, por ejemplo el caso de la provisión de Internet a la oficina de Intertel, para ellos se ha asignado la red 190.152.73.16/29.

En consecuencia, se pueden establecer tres categorías para el tratamiento del tráfico: aquellas orientadas a la navegación web HTTP o HTTPS, y aquellas relacionadas con el audio y video en línea.

Según el departamento comercial[50], menciona que en el seguimiento postventa del servicio de internet que se realizan a los clientes, las aplicaciones que más se requieren son las que se listan a continuación:

- Navegación vía Web (HTTP): Entre las más usadas están las consultas de Google, y las redes sociales como Facebook o Twitter.

- Navegación Web Segura (HTTPS): Relacionadas con aplicaciones como las de correo electrónico, consultas web a entidades públicas, hotmail, gmail, yahoo y transacciones seguras en general.

- Protocolos Streaming: Las principales aplicaciones que se encuentran son: Youtube, justin.tv, Flash Video en general, videoconferencias Messenger. Dentro de esta aplicación se incluyen los protocolos: RTP, MSNP; y la aplicación Macromedia Flash Communications Server.

### 2.6.2 DESCRIPCIÓN DE LAS APLICACIONES MÁS USADAS POR DOMINIO

En la Figura 2.15, se indican los dominios que sobresalen en la navegación web y que se consideran como los servicios que más se usan con el 82.46% de ocupación. Los cuatro son más requeridos: akamaitechnologies.com, ecutel.com, 1e100.net y facebook.com.

[50] Sra. Leonela Tapia, jefe del Departamento Comercial

A continuación se describe brevemente cada uno de los dominios:

a) Akamai Techonologies, Inc [33]: comprende una plataforma distribuida que almacena el contenido de diferentes sitios web. Algunos de sus clientes son: Microsoft Yahoo, MSN Live, Cnet, IBM, RedHat, MySpace, entre otros.

b) Ecutel (Ecuador Telecom S.A.)[34]: es una empresa que representa jurídicamente a la operadora Claro en Ecuador y su gran expansión en el país ha permitido brindar servicios de calidad como televisión digital, Internet y telefonía IP.

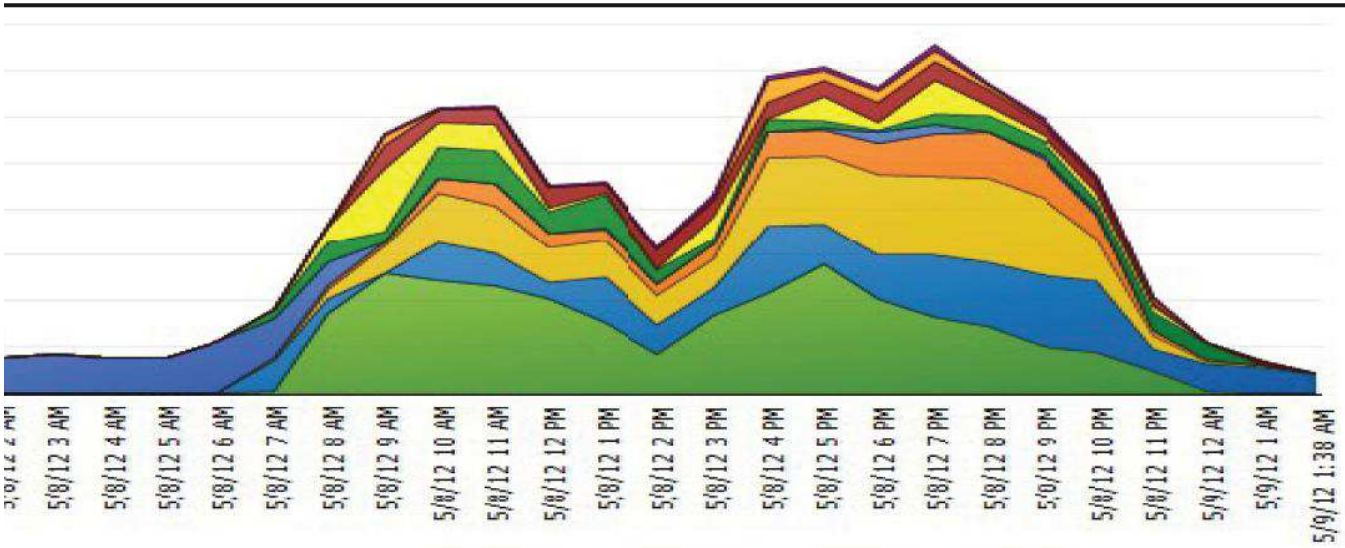
c) 1e100.net [35]: es el nombre de dominio usado para identificar los servidores en la red de Google. En las direcciones IP que aparecen asociados a éste dominio se encuentran incluidas aquellas relacionadas al reproductor de video Youtube.

d) Facebook, Inc [36]: sitio web para redes sociales más popular y utilizado en la actualidad, además de navegación web incluye multimedia.

Como complemento se presenta la Tabla 2.5, donde se listan las redes destino, a las cuales los host de los clientes se conectan frecuentemente. (Ver **Anexo 4**).

| **Asignaciones de direcciones IP (IANA)** | **Organización** | **Área Relacionada** |
|---|---|---|
| **186.5.0.0/17** | Telconet S.A. Guayaquil | DNS, Navegación Web |
| **74.125.0.0/16** | Google | Consultas y Videos on line |
| **208.117.224.0/19** | Youtube | Videos on line |
| **199.9.248.0/21** | Justin Tv | Canales on line |
| **72.246.0.0/15** | Akamai Technologies | Servicios CDN |
| **69.171.224.0/19** | Facebook Inc. | Red Social |
| **65.52.0.0/14** | Microsoft Corp. | Chat y Messenger |
| **200.124.240/20** | Ecuador Telecom. | DNS, Navegación Web |
| **186.46.128.0/19** | CNT | DNS, Navegación Web. |
| **95.211.0.0/16** | Leaseweb | Hosting, Navegación Web |

Tabla 2. 5: Redes IP destino más frecuentes en la RIT [34]




| DOMAIN | INGRESS BYTES | EGRESS BYTES | INGRESS PACKETS | EGRESS PACKETS | PERCENT |
|---|---|---|---|---|---|
| akamaitechnologies.com | 2.3 Gbytes | 10.6 Gbytes | 4.9744 M | 8.5242 M | 22.86% (1.16%) |
| ecutel.net | 7.4 Gbytes | 2.8 Gbytes | 5.8229 M | 4.3247 M | 18.08% (0.92%) |
| 1e100.net | 1.4 Gbytes | 6.7 Gbytes | 2.9908 M | 5.8106 M | 14.32% (0.73%) |
| facebook.com | 1.0 Gbytes | 2.5 Gbytes | 2.1446 M | 3.1526 M | 6.18% (0.31%) |
| webazilla.com | 215.8 Mbytes | 3.1 Gbytes | 347.301 K | 2.1679 M | 5.95% (0.3%) |
| linw.net | 596.2 Mbytes | 2.3 Gbytes | 1.2655 M | 1.876 M | 5.18% (0.26%) |
| avast.com | 492.0 Mbytes | 1.9 Gbytes | 1.0798 M | 1.4241 M | 4.2% (0.21%) |

Figura 2. 15: Aplicaciones por dominio más usada

### 2.6.3 DESCRIPCIÓN DE LOS TIPOS DE CLIENTES

De acuerdo al departamento comercial, para el servicio inalámbrico se tienen dos tipos de clientes:

- Clientes Cyber: son aquellos usuarios que brindan servicio de Internet a terceros, es decir, utilizan el servicio contratado por Telydata para su negocio propio.

- Clientes Residenciales: Son aquellos usuarios de hogar con menor número de computadores. Generalmente utilizan páginas web y correo electrónico.

En la Figura 2.16 se demuestran los usuarios (determinados por su dirección IP) que mayor ocupación del canal en la RIT.

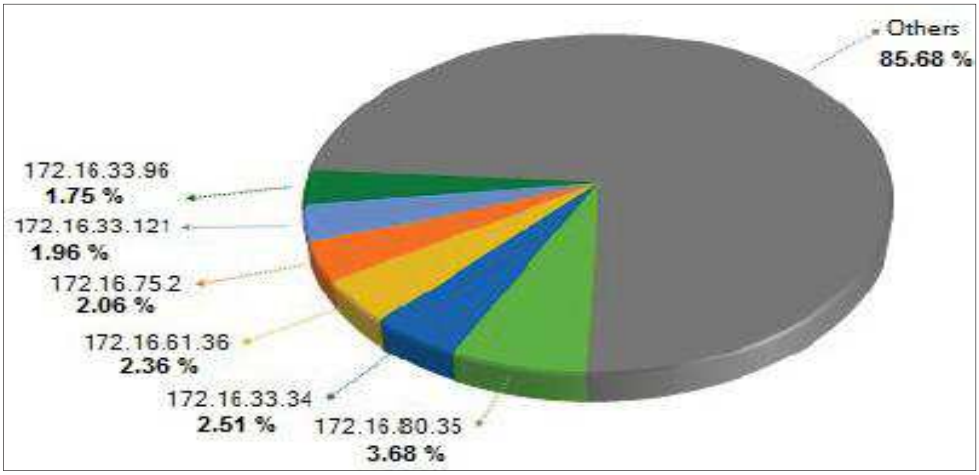



Figura 2. 16: Direcciones IP más usadas[51]

La mayoría de direcciones IP que aparecen en la Figura 2.16 pertenecen a usuarios tipo Cyber. En la revisión de la capacidad asignada a algunos de ellos se denota que exceden de la velocidad contratada. Por ejemplo: la IP 172.16.33.34 tiene contratado una velocidad de 550/250 Kbps, sin embargo, presenta un consumo como se indica en la Figura 2.17.

[51] Gráfico tomado de la sección *NTA Summary , Top 60 Endpoints* del servidor Orion NMP, 7 Mayo del 2013

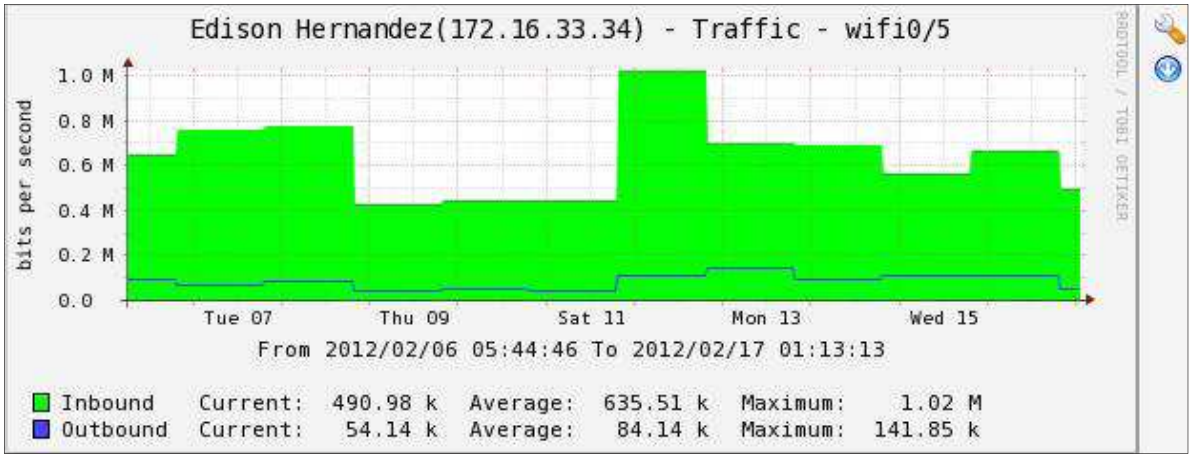



Figura 2. 17: Gráfico Cliente Edison Hernández

El periodo de monitoreo que indica la Figura 2.17 es desde el 6 de Febrero del 2012 al 17 de Febrero del 2012.

El cliente adquiere una capacidad del canal promedio de 635.51 Kbps con un pico máximo de 1020 kbps. Los días de máxima ocupación del canal son para el día martes 7 de febrero del 2012 y domingo 12 de febrero del 2012. Por otro lado, hasta ahora tanto los usuarios residenciales como los corporativos tienen el mismo tratamiento en la red, no se tiene un control de la capacidad de acceso al canal a nivel de core de la RIT.

## 2.7 CALIDAD DE SERVICIO BRINDADA A LOS USUARIOS

Actualmente en Telydata no se maneja un esquema de calidad de servicio. No se especifican parámetros, ni valores referenciales para mantener el enlace operativo. En consecuencia, se establecen algunos parámetros encontrados, con los que opera el departamento técnico de la empresa para asegurar un servicio estable. El monitoreo de los parámetros se realizaron durante el mes de abril (al igual que en la sección 2.5.3), en el cual se utilizaron las siguientes herramientas:

- Comando *mtr* (Linux) para la medición de pérdida de paquetes
- Comando *ping* (Linux) para la medición de la latencia
- Servidor CACTI para la medición de la disponibilidad

1. *Capacidad del Canal*

De acuerdo al monitoreo de la sección 2.5.3 y 2.6, se puede establecer la Tabla 2.6.

| Categoría | Asignación del canal | Porcentaje de uso |
|---|---|---|
| **Navegación Web (HTTP)** | 16.83 Mbps | 91 % |
| **Navegación Web Segura (HTTPS)** | 1.16 Mbps | 6 % |
| **Streaming de Audio y Video** | 0.28 Mbps | 2 % |

Tabla 2. 6: Distribución de la capacidad del canal por categoría

2. *Medición de la tasa de paquetes perdidos*

Los valores se presentan en la Tabla 2.7

| Lugar de Medición | Perdida de Paquetes | |
|---|---|---|
| | Min. | Max. |
| **Backbone RIT Norte ( Respecto al AP La Planada)** | 0 % | 4 % |
| **Backbone RIT Sur ( Respecto al AP Camal Metropolitano)** | 0% | 8% |
| **Desde un cliente del Sur ( Respecto al AP del cliente Alexandra Parreño)** | 1.5% | 8% |
| **Desde un cliente del Norte (Respecto al AP del cliente José Montero)** | 0% | 6 % |

Tabla 2. 7: Medición de la tasa de paquetes perdidos

3. *Medición de la Disponibilidad*

Los valores se presentan en la Tabla 2.8.

| RED | Disponibilidad del Enlace (%) |
|---|---|
| **Enlace Iñaquito Norte ( Respecto al AP La Planada)** | 99.54 |
| **Para un cliente del Norte (Respecto al AP del cliente José Montero)** | 98.26 |
| **Enlace Iñaquito Sur ( Respecto al AP Camal Metropolitano)** | 99.26 |
| **Para un cliente del Sur (Respecto al AP del cliente Alexandra Parreño)** | 97.26 |

Tabla 2. 8: Medición de la Disponibilidad

4. *Medición de Latencia*

Los valores se presentan en la Tabla 2.9.

| Lugar de Medición | Retardo | |
|---|---|---|
| | Min. | Max. |
| **Backbone RIT Norte (Respecto al AP La Planada)** | 8.15 ms | 128.97 ms |
| **Backbone de la RIT Sur (Respecto al AP Camal Metropolitano)** | 4.38 ms | 154.98 ms |
| **Desde un Cliente del Sur al Internet (Respecto a Alexandra Parreño)** | 131.77 ms | 365.50 ms |
| **Desde un Cliente del Norte al Internet (Respecto a José Montero)** | 102.25 ms | 140.45 ms |

Tabla 2. 9: Medición de la latencia

***Aclaración:***

De los cuatro parámetros monitoreados en esta sección, el proyecto se enfocará sobre el primero (*ítem 1. Capacidad del Canal*), por los siguientes motivos:

- No se tiene ningún mecanismo de distribución y gestión de la capacidad del canal para cada cliente de la empresa de acuerdo a su contrato. Incluso, esto fue expuesto en la Sección 2.4 como uno de los requerimientos específicos del gerente técnico de la empresa.

- Tanto los parámetros de paquetes perdidos, disponibilidad del enlace y retardo se encuentran fuera del alcance, ya que esto depende de la infraestructura propia del WISP, que ya ha sido implementada, por lo que requiere de un análisis y rediseño de toda la red para un mejoramiento considerable.

## 2.8 DIAGNÓSTICO Y REQUERIMIENTOS ESPECÍFICOS DE LA RIT

A continuación se puntualizan tres observaciones tanto del área organizacional como técnica:

- Los objetivos del negocio establecidos por la gerencia son mejorar la gestión del servicio para la RIT de la empresa, ya que su futuro comercial es cubrir mayor mercado como WISP en partes distintas al sector urbano y rural de Quito. Sin embargo, como se describió en el presente capítulo no se maneja una especificación del nivel de servicios respecto a la capacidad del canal contratado por sus usuarios.

- Al momento el departamento técnico no tiene la infraestructura bien definida, ni datos de sus clientes en un repositorio debidamente organizado, unicamente se dispone del CRM que es usado para multiples funcionalidades tanto para el marketing como para la informacion técnica, por lo que se hace necesario elaborar una base de datos que aumente la eficacia en el acceso a la informacion de los recursos al momento de ejercer el soporte técnico por el personal de Telydata. En consecuencia, esto ayudaría a reducir el tiempo de solucion frente a una falla. Esto tambien lo plantea el gerente técnico y se verifica en la sección 2.4 del presente Capitulo

- No se ha podido optimizar el trabajo que realiza el soporte técnico de la empresa al no tener un planeamiento ordenado de la red y topologias acualizadas.

# CAPÍTULO 3

# DISEÑO DEL SISTEMA DE GESTIÓN

*En el presente capítulo se plantean los requerimientos generales que debe satisfacer el sitema de gestión. Luego se realiza el diseño de un módulo gestor, un módulo ejecutor y la selección de las herramientas a utilizar según la infraestructura de Telydata Cía. Ltda.*

## 3.1 REQUERIMIENTOS GENERALES DEL SISTEMA DE GESTIÓN

A continuación se puntualizan los requerimientos generales que se establecen para el diseño del sistema de gestión:

- Se necesita diseñar e implementar un entorno de gestión orientado al Departamento Técnico de la empresa que permita mantener un repositorio de información, acerca de los clientes y sus correspondientes datos técnicos organizados (conforme a lo establecido en la sección 2.4.1.).

- Se requiere que el sistema opere en la capa de core de la red, el cual permita interactuar con la red inalámbrica. Para ello, se debe tomar en cuenta los requisitos de infraestructura que la gerencia técnica establece para su implementación como es el uso de software libre.

- Se requiere plantear las especificaciones del nivel de servicio respecto a la capacidad de acceso por cada cliente, de acuerdo a las políticas establecidas por el departamento comercial.

- Se debe incluir en el repositorio de información las especificaciones del tráfico descrito en la sección 2.6, tomando en cuenta para ello, los puertos de los protocolos más usados y las direcciones IP más frecuentes.

- De acuerdo a la sección 2.4, capítulo 2, uno de los requerimientos que se plantea es un reinicio programado de las antenas del backbone para mejorar el rendimiento de las mismas. El departamento técnico de la empresa establece hacerlo una vez por semana.

## 3.2 PLANTEAMIENTO DEL SISTEMA DE GESTIÓN PARA LA RIT

Se plantea un esquema compuesto de un módulo Gestor y un módulo Ejecutor. Cada uno de ellos tendrá como base un servidor independiente, así como un mecanismo de envío de información desde el Gestor al Ejecutor. En la Figura 3.1 se muestra el diagrama general.

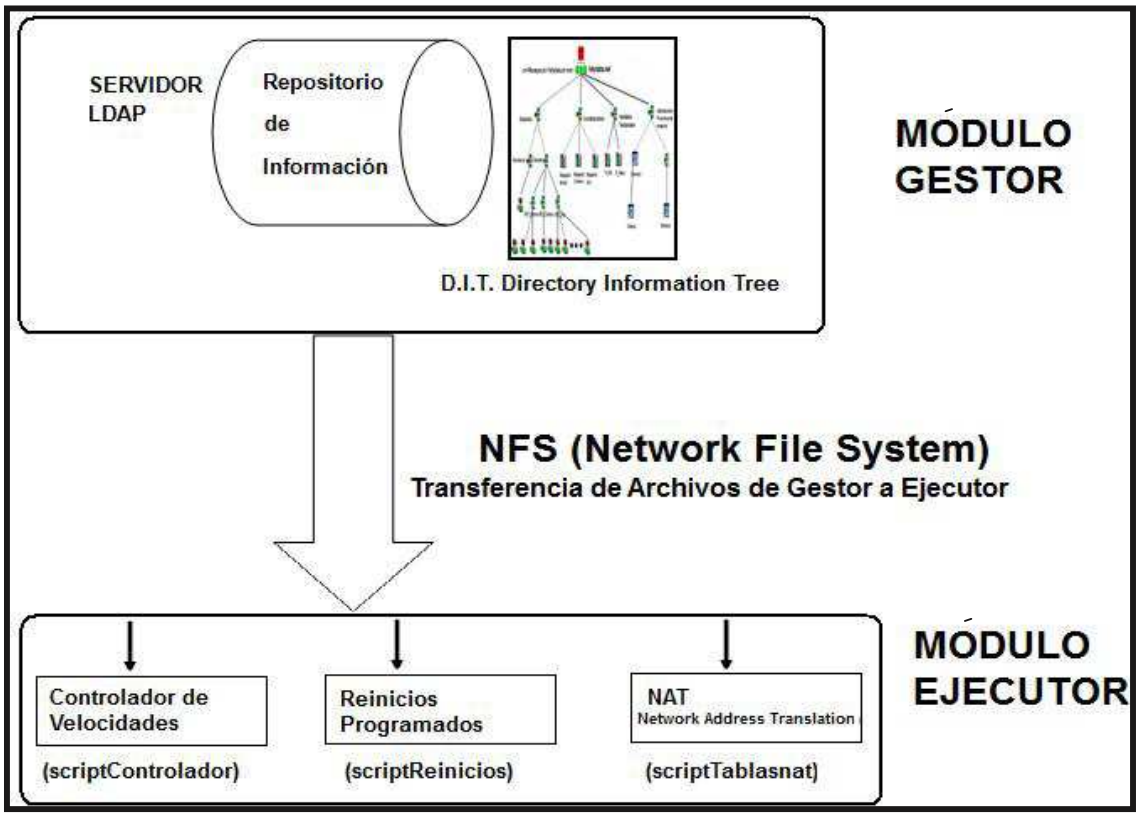



Figura 3. 1: Planteamiento del Sistema de Gestión para la RIT.

El módulo gestor debe cumplir con el requerimiento de almacenar la información en un repositorio con los datos de clientes e información técnica relacionada a ellos. Para ello, se tendrá que definir un modelo de datos específico.
El módulo ejecutor en base a la información almacenada en el repositorio de información, debe cumplir con los requerimientos de control de la capacidad para cada cliente y reiniciar las antenas del backbone periódicamente.

## 3.3 REQUERIMIENTOS ESPECÍFICOS DEL MÓDULO GESTOR

El requerimiento principal es que se opte por un servicio de directorio y su debida administración que considere los siguientes aspectos:

- El modelo de información de políticas según el RFC 3198 recomienda el uso de directorios, específicamente el uso de LDAP, según la definición existente en el RFC 4104 (PCELS).

- Que disponga de un protocolo no muy complejo, para facilitar al personal del departamento técnico el acceso a los datos desde cualquier lugar de forma eficiente.

- Que permita a los datos ser distribuibles y replicables, tal que se pueda usarlos en todos aquellos equipos que necesitan la información. Permitir la realización periódica de un backup no muy complejo.

Teniendo en cuenta dichas puntualizaciones, se elige el uso de directorios como repositorio de información a través del protocolo LDAP

### 3.3.1 DISEÑO DEL MÓDULO GESTOR

Es necesario que el módulo gestor cumpla con las siguientes dos funcionalidades:

A) Almacenamiento de Información: almacena datos como usuarios, localizaciones físicas, datos técnicos de dispositivos inalámbricos y la

referencia temporal del día en que se requiere reinicio de antenas. En la información de cada usuario se incluirá la capacidad del canal contratado. Además incluirá gráficos de las topologías.

B) Entrega de información: las variables de gestión obtenidas desde el repositorio como valores de velocidades, referencia temporal y aquellas relacionadas al tráfico serán entregadas al módulo ejecutor mediante un *script* denominado *datosClientes*.

#### 3.3.1.1 Diseño del modelo de información

En la Tablas 3.1 se indica un resumen que agrupa las principales entidades de información que están relacionadas con la RIT, y sus respectivas definiciones.

| Entidades | Definiciones | Variables de Red |
|---|---|---|
| **Usuarios, que pueden ser técnicos o clientes** | Todo lo que tiene que ver con los datos generales y técnicos de clientes, además incluye los datos del personal de soporte. | Por ejemplo:<br>direcciones IP<br>Nombre, Teléfono, Dirección, email. |
| **Localizaciones Físicas**<br>**Red Iñaquito Norte**<br>**Red Iñaquito Sur**<br>**Red Iñaquito Centro** | Hace referencia a ubicaciones físicas de cada una de las redes inalámbricas. Debería almacenar topologías e información general del enlace. | Rangos de direcciones IP por localidad<br>Ejemplo:<br>RIT Norte : 172.16.33.0/24<br>RIT Sur:172.16.41.0/24<br>Topologías |
| **Entorno Aplicacional** | Números de puertos que se usen con más frecuencia, y direcciones IP destino más frecuentes. | Ejemplos:<br><br>80, 8080, 443, 1863, etc.<br>186.5.0.0/17 |
| **Referencia Temporal** | Hace referencia al día y hora en que se ejecutará el *script* para el reinicio de los dispositivos inalámbricos. | Ejemplos:<br>Lunes<br>Martes<br>Miércoles |
| **Variables técnicas de dispositivos inalámbricos** | Especificación de parámetros técnicos. Se incluirán en la cuenta de cada usuario | Ejemplo:<br>-ccq, signal ,strenght<br>-Frequency |

Tabla 3. 1: Variables de la RIT

Se pretende que todas las entidades sean representadas en un directorio LDAP en base de clases de objetos y atributos.

#### 3.3.1.2 Representación de entidades en Clases y Atributos de LDAP

Las entidades que constituyen el directorio de información según la Tabla 3.1, se representan por clases y atributos del protocolo LDAP, por lo tanto, se presentan a continuación las entidades que manejarán dichas clases y atributos.

##### *3.3.1.2.1 Representación de la Entidad Usuario*

Es la entidad que contiene la información tanto de aspectos básicos como técnicos acerca de los clientes de la RIT, ya que es importante almacenar el perfil de un cliente específico en el directorio. La información básica de cada cliente es:

- *Nombres*: almacenará por lo menos un nombre y un apellido
- *Dirección*: almacenará la dirección domiciliaria o referencia donde viva el cliente
- *Teléfono*: almacenará el teléfono celular o teléfono convencional del cliente
- *Categoría de Servicio*: describe si el cliente es residencial o cyber
- *Mail*: Describe la dirección de correo electrónico del cliente
- *Dirección IP*: contiene la dirección IP de la interfaz WLAN del dispositivo
- *Dirección MAC:* describe la dirección MAC de la interfaz WLAN del dispositivo

Para lograr representar los datos anteriormente mencionados se plantea la asignación de las clases en LDAP presentadas en la Figura 3.2.

Definición de clases [37][38] [39][40]

- *person*: es una clase estructural base para representar personas. Su clase superior es top.

- *inetOrgPerson*: clase estructural que representa a personas quienes están asociadas de alguna manera con una organización; se deriva de la clase organizationalPersonal. Está definida en X.521.

- *ipHost*: clase auxiliar que permite la representación de un host como un dispositivo IP.

    - *ipHostNumber*: es el atributo que representa el número de dirección IP de las estaciones inalámbricas del cliente.

**FORMATO EJEMPLO VALOR: 172.16.33.1**

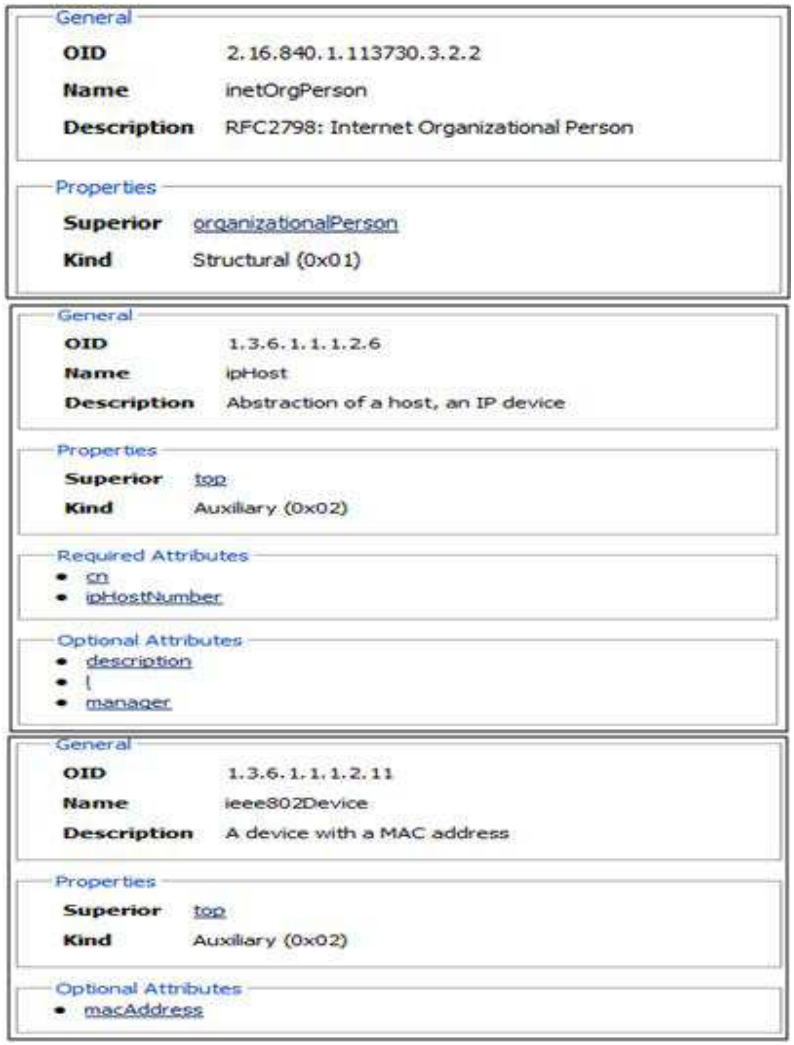



Figura 3. 2: Clases para la entidad Usuario [37]

- *ieee802Device*: es una clase que contiene el valor de la dirección física del dispositivo.
    - *macAddress*: es un atributo que representa el valor de dirección MAC de un dispositivo de red. Almacenará las direcciones MAC de los dispositivos inalámbricos.

**FORMATO EJEMPLO VALOR: 00:27:22:DE:8B:E9**

Por otro lado, respecto a la información técnica y de acuerdo al Capítulo 2, sección 2.3.4, se debería tomar en cuenta los aspectos configurados en las antenas como:

- *Equipo:* describe el tipo de dispositivo *Airmax Ubiquiti Networks* instalado.
- *SSID*: la identificación de la conexión inalámbrica a nivel de última milla.
- *Frecuencia de Operación:* contiene el dato de frecuencia a la que se encuentra operando el AP principal. (Valores *5.XXX*[52] *GHz* o *2.XXX GHz*).
- *Velocidad Contratada:* contiene la velocidad contratada por el cliente (Valor: $V_{bajada}$ / $V_{subida}$ *Kbps*).
- *CCQ:* contiene el valor de la calidad de la conexión. (Valor: *X %*).
- *Nivel de Señal*: el nivel de la señal recibida respecto al ruido del enlace inalámbrico (Valor: - XX dBm).

No se tiene clases en LDAP para definir dicha información por lo tanto se diseña un esquema para las variables de *Airmax Parameters*.

### *3.3.1.2.2 Diseño del Esquema Airmax Parameters*

Cuando no se reúne las necesidades con los archivos *schema* predefinidos por defecto en un directorio LDAP, se puede crear un esquema personalizado. Para ello se debe asignar un OID previamente registrado en IANA. El OID servirá para la definición de todos los nuevos tipos de atributos y clases. En la Figura 3.3 se indica

[52] El digito X representa cualquier valor numérico entre 0 y 9.

que para Telydata se ha asignado el número PEN[53] 40922, por lo tanto el prefijo OID es: 1.3.6.1.4.1.40922. Se elegirá el sufijo 1 para la red inalámbrica. En la Figura 3.4 se plantea la definición de la clase *Airmax Parameters* y sus atributos. La clase *Airmax Parameters* fue establecida en un archivo denominado *parameters.schema* y se incluirá en la configuración del servidor, Capitulo 4, para satisfacer el almacenamiento de la información técnica de los dispositivos CPE.


[IANA] Your request for a Private Enterprise Number has b... Mensaje 4 de 10

Remitente [IANA] Pen System Response
Destinatario aandrade@telydata.net
Responder a do-not-respond@iana.org
Fecha 29.11.2012 12:09

Dear Alejandro Augusto Andrade Mafla,

Your application [Request-53645] has been completed and a PEN has been assigned.

Your organization's PEN is: 40922

Below is a copy of your PEN record. This record will be visible in the IANA PEN registry within an hour.

The IANA PEN registry is available at the following address:

http://www.iana.org/assignments/enterprise-numbers

Figura 3. 3: Asignación de un numero PEN

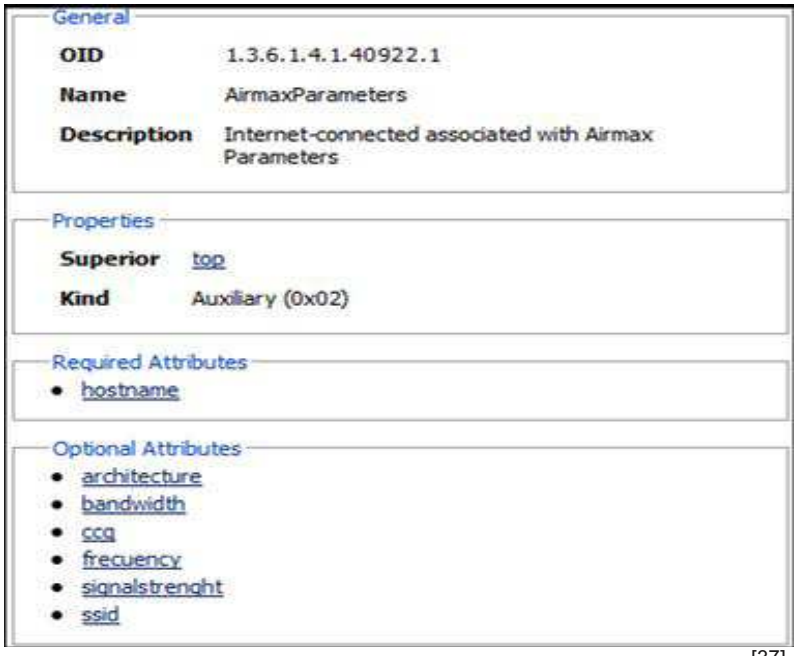



Figura 3. 4: Definición de la clase *Airmax Parameters* [37]

[53] PEN (Private Enterprise Number): son números asignados por IANA a un registro público en Internet para identificar un conjunto de objetos definidos por una empresa. Usados típicamente en SNMP, X.500 y LDAP

A continuación se describen los atributos de la clase *Airmax Parameters*:

- *ssid*: nombre de la identificación de la red (LAN 802.11 Service Set ID) de la estación inalámbrica del cliente.

  **FORMATO EJEMPLO VALOR: La Planada WT**

- *frequency*: almacena el valor de la frecuencia en que opera el punto de acceso, y con el que está conectado la estación inalámbrica del cliente.

  **FORMATO EJEMPLO VALOR: 5260 MHz**

- *bandwidth*: almacena el valor de la capacidad del canal establecido en el contrato entre el departamento comercial y el cliente.

  **FORMATO EJEMPLO VALOR: 1300/300 Kbps**

- *ccq*: almacena el valor del porcentaje de la calidad de conexión del enlace. Definición CCQ de transmisión, sección 2.4.1. Capítulo 2.

  **FORMATO EJEMPLO VALOR: 97 %**

- *signalstrength*: almacena el valor del nivel de la señal inalámbrica recibida en la estación del cliente. Definición Intensidad de la señal, sección 2.4.1, Capitulo 2.

  **FORMATO EJEMPLO VALOR: - 63 dbm**

En la Tabla 3.2 y Tabla 3.3 se resume las clases y atributos pertenecientes a la entidad Usuario.

| Mapeo de Información de la Entidad Usuario | | | |
|---|---|---|---|
| **Ítem** | AttributeType | ObjectClass | Schemas OpenLDAP |
| **Nombres** | Cn | Person | core.schema<br>inetorgperson.*schema* |
| **Dirección** | streetAddress | organizationalPerson | Core.schema |
| **Teléfono** | telephoneNumber | | |
| **Categoría del Servicio** | businessCategory | inetOrgPerson | inetorgperson.schema |
| **Mail** | Mail | | |
| **Dirección IP** | ipHostNumber | IpHost | nis.schema |
| **Dirección MAC** | macAddress | ieee802Device | nis.schema |

Tabla 3. 2: Mapeo de Información técnica para la entidad Usuario

| Datos principales de la Infraestructura Inalámbrica Ubiquiti Networks | | | |
|---|---|---|---|
| **Ítem** | AttributeType | ObjectClass | Schemas OpenLDAP |
| **Equipo** | Hostname | *Airmax Parameters* | parameters.schema |
| **SSID** | Ssid | | |
| **Frecuencia de Operación** | Frequency | | |
| **Velocidad** | Bandwidth | | |
| **CCQ** | Ccq | | |
| **Nivel de Señal** | Signalstrength | | |

Tabla 3. 3: Mapeo de Información técnica para la entidad Usuario para la Infraestructura Ubiquiti Networks

*3.3.1.2.3 Entidad Localizaciones Físicas*

Esta entidad se refiere a las ubicaciones en las que se sitúa cada red. En este caso identificará rangos de direcciones IP y la topología de la subred respectiva. Para ello se plantean las clases presentadas en la Figura 3.5.

Definición de clases [41] [42]

- *Locality*: es una clase estructural que representa localidades como ciudades, países u otras regiones geográficas y se describe a través del atributo opcional "L". Se usará para representar las localidades de la RIT y almacenar las topologías.

- *pcelsIPv4AddrValueAuxClass*: Clase auxiliar que representa el valor de un conjunto de direcciones IPv4, rangos de direcciones IPv4

    - *hosts*: Almacena las redes específicas de la RIT. Esto a través de su atributo requerido pcelsIPv4AddrList

**FORMATO EJEMPLO VALOR: 172.16.41.0/24,172.16.61.0/24**

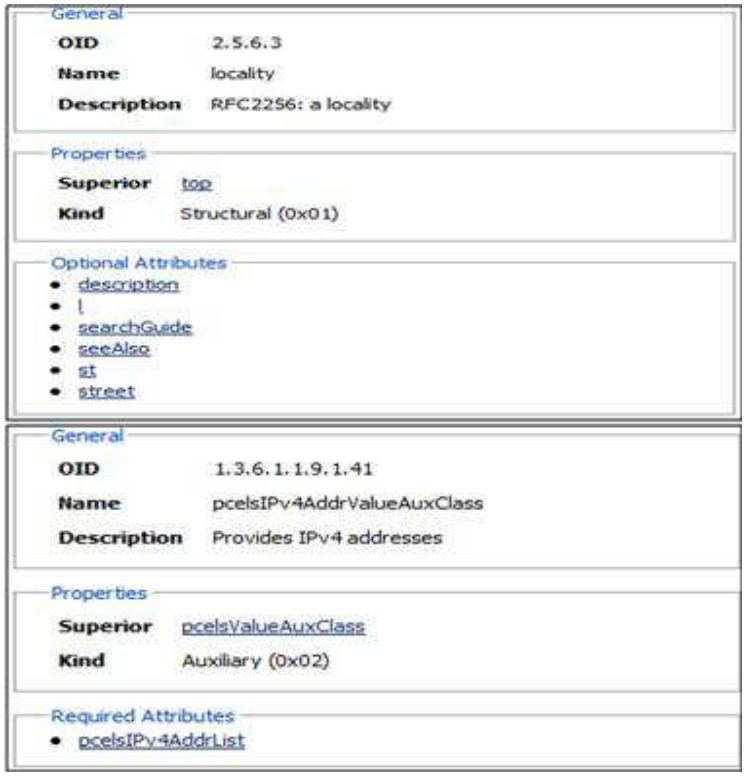



Figura 3. 5: Clases para la Entidad Localizaciones Físicas [37]

*3.3.1.2.4 Entidad Entorno Aplicacional*

En esta entidad se almacena la información de aspectos como direcciones IP destinos más frecuentes y aquellas relacionadas a las aplicaciones más usadas, representadas a través de puertos. Se los representa como datos de tipo entero o como cadena de caracteres. Para ello se plantean las clases presentadas en la Figura 3.6.

Definición de clases [42]

- *pcelsRoleCollection*: es una clase estructural que permite definir un conjunto de elementos gestionados que tienen un rol común. Su clase superior es *pcimPolicy*. Se usará para representar el rol que cumple la entidad tráfico.

- *pcelsIntegerValueAuxClass*: es una clase auxiliar que provee un conjunto de enteros o rangos de enteros. Será usada para la representación de puertos.
- *pcelsDestinationPortVariableAuxClass*: Es una clase auxiliar que almacenará los puertos de red en el rango de 0…65535.

**FORMATO EJEMPLO VALOR: 80,8080,21,25,110,1936**

- pcelsIPv4AddrList: es un atributo requerido para representar un conjunto de direcciones de red IPv4. Se usará para almacenar las redes IP destino más frecuente.

**FORMATO EJEMPLO VALOR: 186.125.0.0/16,186.5.0.0/17**

En la Figura 3.6 se representa las clases anteriormente mencionadas.

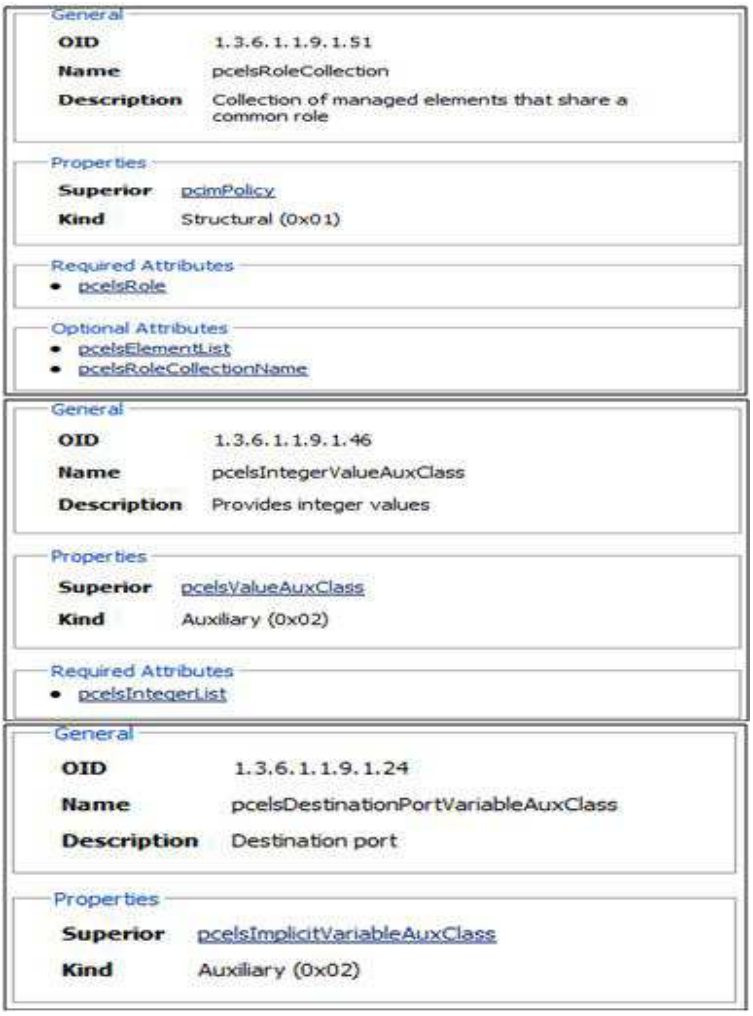



Figura 3. 6: Clases de la entidad Entorno Aplicacional [37]

*3.3.1.2.5 Entidad Variable Temporal*

Hace referencia al día de la semana en que ejecutarán los reinicios programados. Se describe a través de una sola clase, como se aprecia en la Figura 3.7.

Definición de la Clase [43]:

- pcimTPCDayOfWeekMask: describe una máscara para representar el día de la semana. Será usado para asignar el día en que el equipo inalámbrico debe reiniciarse.

El formato ejemplo de la clase pcimTPCDayOfWeekMask es:

**FORMATO EJEMPLO VALOR: '0111110' B**


Figura 3. 7: Clase Variable Temporal [37]

### 3.3.1.3 Diseño del árbol de Directorio (DIT)

El diseño del DIT para la empresa Telydata, se realiza en base a las entidades anteriormente expuestas, según la Tabla 3.1.

Determinación del RDN (RootDN)

Según el RFC 2247, el cual provee un marco de nombrado jerárquico general para directorios basados en LDAP, recomienda el uso de la clase dcObjetct, que permite a un atributo DC (*Domain Component*) presentar a una entrada principal. El dominio de Telydata que está registrado es: *telydata.net*, por lo tanto se elegirá el nombre de la siguiente manera*:*

***dc=telydata,dc=net***

Para el base DN general se hace uso del atributo requerido CN (*Common Name*) de la clase estructural *Person*, en donde se elige el nombre de un objeto administrador que en este caso se lo llama *Manager*, por tanto, el base DN es:

***cn=Manager,dc=telydata,dc=net***

*3.3.1.3.1 Entradas Principales del Árbol*

Para cada rama principal del árbol se elije un atributo *OU (organizationalUnit)*, específicamente se lo utiliza para representar las entradas de segundo nivel del árbol DIT. A continuación se presentan las entradas:

ou= Usuarios: Esta rama del árbol almacenará los datos de dos clases de usuarios que se referencian en la Tabla 3.1. Estos son: técnicos y clientes

- DN1: ou=Tecnicos,ou=Usuarios,cn=Manager,dc=telydata,dc=net; presenta toda la información relacionada a los técnicos.
- DN2: ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net; representan toda la información de los clientes.
    - DN3: ou=RIT_Norte, ou=Clientes, ou=Usuarios, cn=Manager, dc=telydata, dc=net; representa toda la información relacionada a los clientes del norte de Quito.
    - DN4: ou=RIT_Centro,ou=Clientes,ou=Usuarios,cn=Manager, dc=telydata,dc=net; representa toda la información relacionada a los clientes del centro de Quito.
    - DN5: ou=RIT_Sur,ou=Clientes,ou=Usuarios,cn=Manager, dc=telydata,dc=net; representa toda la información relacionada a los clientes del sur de Quito.

ou= localizaciones: esta rama del árbol hace referencia de la ubicación geográfica de las redes inalámbricas, sus topologías y rangos de direcciones IP que se referencian en la Tabla 3.1

- DN6: l=Quito_Sur,ou=Localizaciones,cn=Manager,dc=telydata,dc=net; representa la ubicación del tramo Quito Sur y contiene la topología Iñaquito Sur.

- DN7: l=Quito_Centro,ou=Localizaciones,cn=Manager,dc=telydata, dc=net; representa la ubicación del tramo Quito Centro, no contiene la topología ya que estará incluida en el anterior DN (DN6).

- DN8: l=Quito_Norte,ou=Localizaciones,cn=Manager,dc=Telydata,dc=net; representa la ubicación del tramo Quito Norte y contiene la topología Iñaquito Norte.

ou=Entorno Aplicacional: esta rama del árbol hace referencia al ambiente aplicacional que se explica en la Tabla 3.1.

- DN9: cn=Tecnico,ou=EntornoAplicacional,cn=Manager,dc=telydata,dc=net; representa todos los aspectos a tomarse en cuenta para el manejo del tráfico.

    - DN10: pcelsRole=Conexiones,cn=TraficoCorporativo,cn=Tecnico, ou=EntornoAplicacional,cn=Manager,dc=telydata,dc=net; representa las direcciones IP de las redes externas con las que se conecta frecuentemente.

    - DN11: pcelsRole=Puertos,cn=TraficoCorporativo,cn=Tecnico, ou=EntornoAplicacional,cn=Manager,dc=telydata,dc=net; representa los puertos con los que se conecta frecuentemente.

ou=VariablesTemporales: esta rama del árbol almacenará el día que se realizará el reinicio programado.

- DN12: pcimTPCDayOfWeekMask='0111110'B,cn=VariableHoraDia, ou=VarialesTiempo,cn=Manager,dc=telydata,dc=net; representa la información del día en que se realizará el reinicio de la antena.

En resumen, se proponen 12 entradas que proveerán de la información básica al sistema de gestión. En la Figura 3.8 se indica el árbol de información construido.

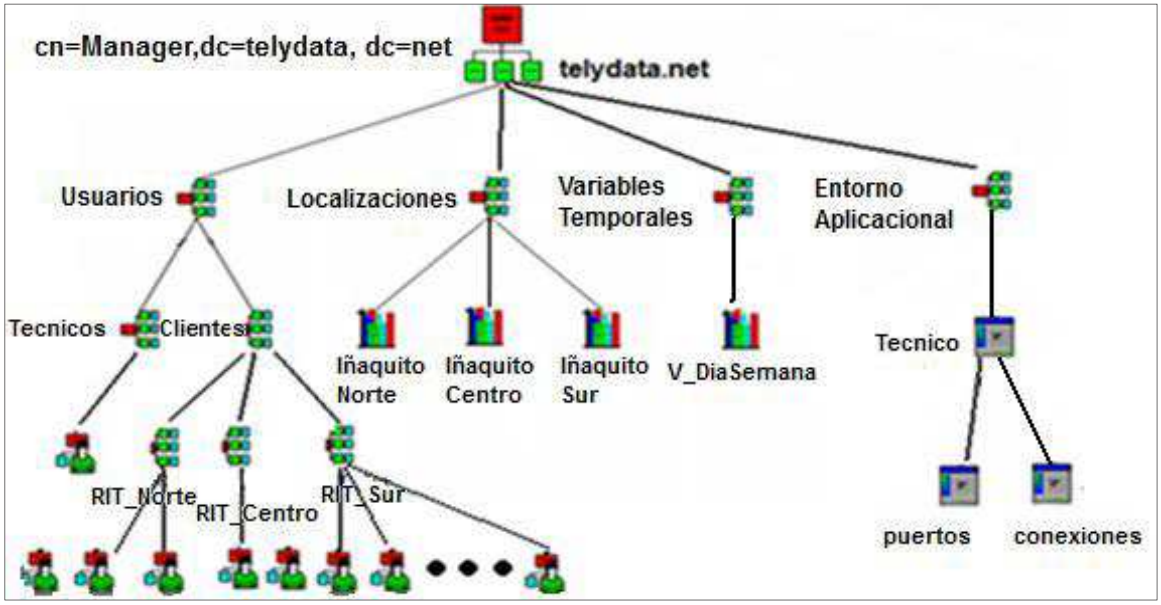



Figura 3. 8: Árbol DIT Propuesto

### 3.3.1.4 Definición de los Esquemas

A continuación se definen los esquemas con los elementos básicos que se hacen necesarios para la configuración del repositorio de información son:

- core.schema:
  - clases de atributos básicos definidos en los RFC 2252 a 2256
  - RFC 1274: Cosine and Internet X.500 Schema
  - RFC 2079: Definition of an X.500 Attribute Type and an Object Class to Hold Uniform Resource Identifiers

  - RFC 2247: Using Domains in LDAP/X.500 Distinguished Names
  - RFC 2377 : Naming Plan for Internet Directory-Enabled Applications

- cosine.schema: Basado en el RFC 1274- Cosine and Internet X.500 Schema
- inetorgperson.schema: definido en el RFC 2798.
- nis.schema: basado en objetos de los RFC 2252 y RFC 2307.

Respecto al modelo de información basado en políticas se describen los nuevos esquemas en LDAP para su posterior implementación:

- telycim.schema: De acuerdo a DMTF[54] C.I.M. versión 2, sirve para implementar los objetos de la parte superior de la jerarquía del modelo PCIM
- telydatapcim.schema: De acuerdo con el RFC 3703
- telydatapcels.schema: De acuerdo con el RFC 4104.
- parameters.schema: esquema propio para la representación de información técnica de la infraestructura inalámbrica de Telydata.

La mayor parte de los esquemas vienen ya definidos, generalmente cuando se instala un servidor LDAP, sin embargo los cuatro últimos mencionados no lo están. Se los implementará en el Capítulo 4.

#### 3.3.1.5 Información del DIT que requiere ser mapeada en el módulo Ejecutor

En resumen, el DIT contiene tres entradas principales con la información necesaria. A continuación se describe:

- La entrada “Entorno Aplicacional” emite los datos que se requieren para configurar el controlador de la capacidad del canal que se va a diseñar. Puntualmente lo que se requiere son las direcciones IP destino y los puertos.

[54] DMTF(Distribuited Management Task Force): es una organización que desarrolla, mantiene y promueve estándares de gestión fundamentales para la interoperabilidad de múltiples proveedores de sistemas.

Con esta entrada se generarán dos archivos denominados: ConexionesIP.txt y PuertosRed.txt

- De la entrada “Usuarios” se requiere los datos del cliente, puntualmente se requiere la velocidad contratada y la dirección IP de cada cliente. A partir de esta se genera un archivo denominado controlador-velocidades.conf.

- De la entrada “Variables Temporales” se requiere la máscara del día en que el dispositivo inalámbrico sea reiniciado. Con esta entrada se generará un archivo denominado InfoTemporal.txt

La única entrada que existe en el árbol DIT y no genera archivos es “Localizaciones”, ya que solo contiene información de ubicación, estaciones principales y topologías

#### 3.3.1.6 PRINCIPALES OBJETOS DEL ÁRBOL

En la Figura 3.9 se muestran los principales objetos del DIT. A continuación se listan un conjunto de DN válidos para traducir o enviar la información requerida al módulo ejecutor:

- Respecto a los usuarios técnicos:

  ou=Tecnicos,ou=Usuarios,cn=Manager,dc=telydata,dc=net

- Respecto a los usuarios Clientes:

  ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net

- Clientes del Norte:

  ou=RIT_Norte,ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net

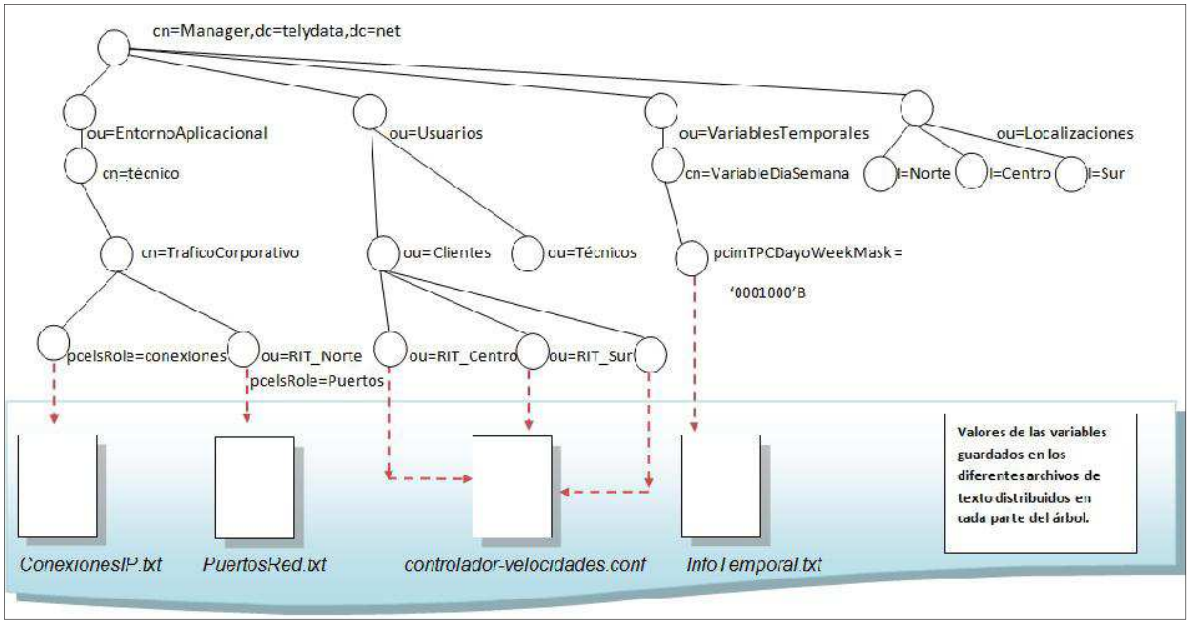



Figura 3. 9: Principales Objetos del DIT

- Clientes del Centro:

  ou=RIT_Centro,ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net

- Clientes del Sur:

  ou=RIT_Sur,ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net

- Respecto a las Localizaciones:

  ou=Localizaciones,cn=Manager,dc=telydata,dc=net

- Localización Norte

  l=Quito_Norte,ou=Localizaciones,cn=Manager,dc=telydata,dc=net

- Localización Centro

  l=Quito_Centro,ou=Localizaciones,cn=Manager,dc=telydata,dc=net

- Localización Sur

  l=Quito_Sur,ou=Localizaciones,cn=Manager,dc=telydata,dc=net

- Respecto a las conexiones

  pcelsRole=Conexiones,cn=Trafico,cn=Tecnico,ou=EntornoAplicacional, cn=Manager,dc=telydata,dc=net

- Respecto a los puertos:

  pcelsRole=Puertos,cn=Trafico,cn=Tecnico,ou=EntornoAplicacional, cn=Manager, dc=telydata,dc=net

- Respecto a los reinicios programados:

  pcelsRole=Reinicios,cn=Trafico,cn=Operativo,ou=EntornoAplicacional, cn=Manager, dc=telydata,dc=net

### 3.3.2. MAPEO DE INFORMACIÓN REQUERIDA PARA EL MÓDULO EJECUTOR

Cada parte del directorio posee variables necesarias para realizar selección, indexado o filtrado de datos. Si se describe el modelo de información y de nombrado en LDAP, se verifica que se dispone de dos partes principales: el DN que conforma un bloque de información dentro del DIT y los diferentes atributos que son parte de este. En consecuencia, el mecanismo de indexado se puede realizar como se indica en la Figura 3.10.

Para obtener el dato que se desea, primero se necesita ubicar la entrada a través de una referencia DN y luego seleccionar el atributo ya sea requerido (MUST) u opcional (MAY).

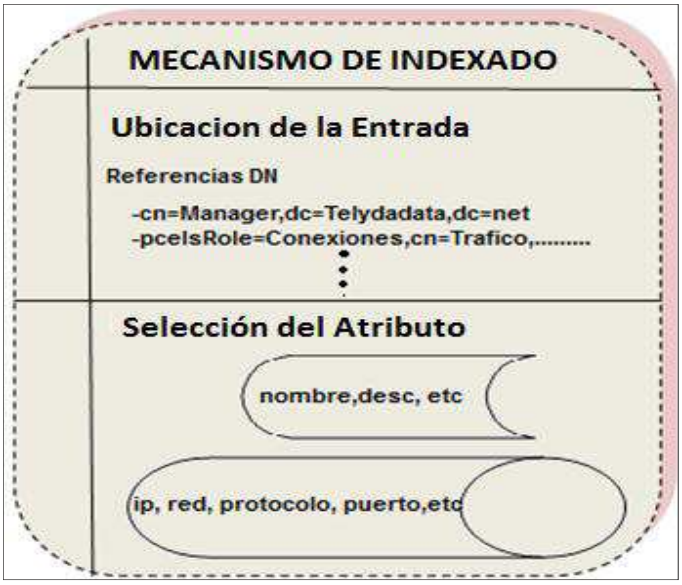



Figura 3. 10: Filtrado de datos [9]

Por ejemplo si se requiere obtener la información de los puertos que se usan en la red, se debe hacer referencia a:

*DN:*
*pcelsRole=Puertos,cn=TraficoCorporativo,cn=Tecnico,ou=EntornoAplicacional, cn=Manager,dc=telydata,dc=net*

*Atributo:*
*pcelsIntegerList: 80, 8080, 443, 1936, 1935, 20, 21, 110*

Por lo tanto el mapeo de información consiste en indexar los datos que se encuentran en los atributos de cada entrada, seleccionándolos según se requiera.

### 3.3.2.1 Mecanismo de filtrado de Información

Para el filtrado de información se debería utilizar una herramienta que permita hacer solicitudes de búsquedas (*ldapsearch)* con sus distintas opciones. Se plantea el uso de dos herramientas del sistema operativo Linux, son: *la tubería pipe ( | )* y el comando *grep*. Con dichas herramientas se podría obtener una lista de valores de

algún objeto específico. Por ejemplo, la forma de obtener la lista de puertos sería a través de la siguiente manera:

*ldapsearch –LLL –x –b pcelsRole=Puertos,cn=TraficoCorporativo,cn=Tecnico, ou=EntornoAplicacional,cn=Manager,dc=telydata,dc=net | grep ´pcelsIntegerList´*

### 3.3.2.2 Diseño del *script* datosClientes

Es un *script* que pretende filtrar y ordenar los datos obtenidos del repositorio de información, en cada uno de los archivos obtenidos según la sección 3.3.1.5. Se plantea que todos los archivos deberán ser reunidos en un solo directorio denominado /etc/gestor, tal como se muestra en la Figura 3.11.

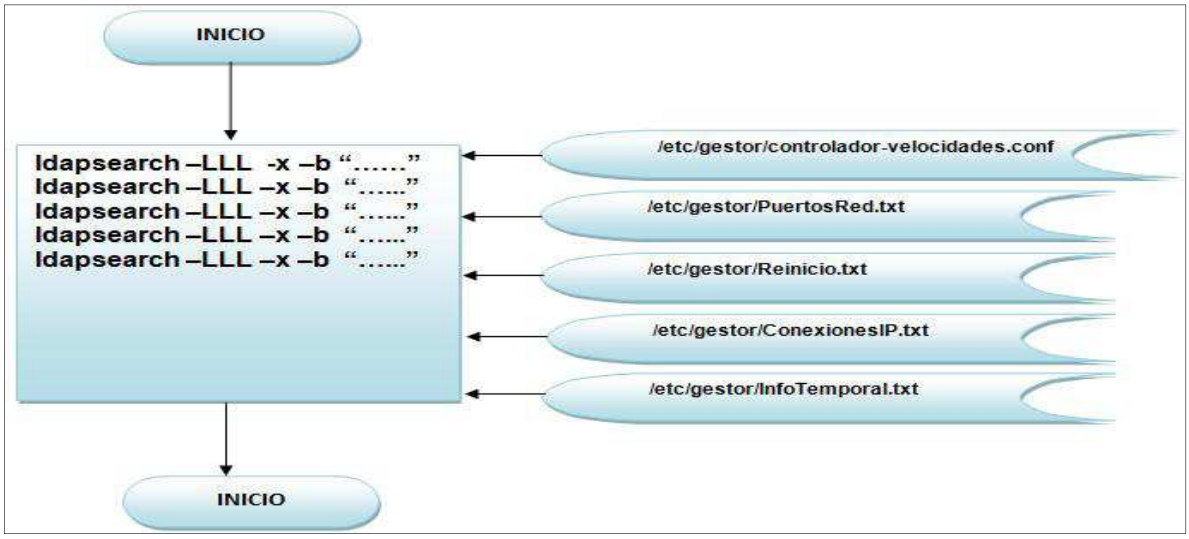



Figura 3. 11: Diseño del *script* datosClientes

Para la generación del archivo controlador-velocidades.conf se debe tener un método, tal que permita arreglar los valores de velocidades según el formato:


Los valores de cada columna serán detallados en la sección 3.4.1.1.2

### 3.3.2.3 Planteamiento de transferencia de archivos requeridos hacia el módulo ejecutor

La ejecución del *script* datosClientes genera los archivos especificados en la Figura 3.9, los cuales son necesarios para transferirlos al modulo ejecutor y permitir su funcionamiento. Se plantea que la transferencia de dicha información se la realice por medio de un montaje simple de red de acuerdo a la Figura 3.12.

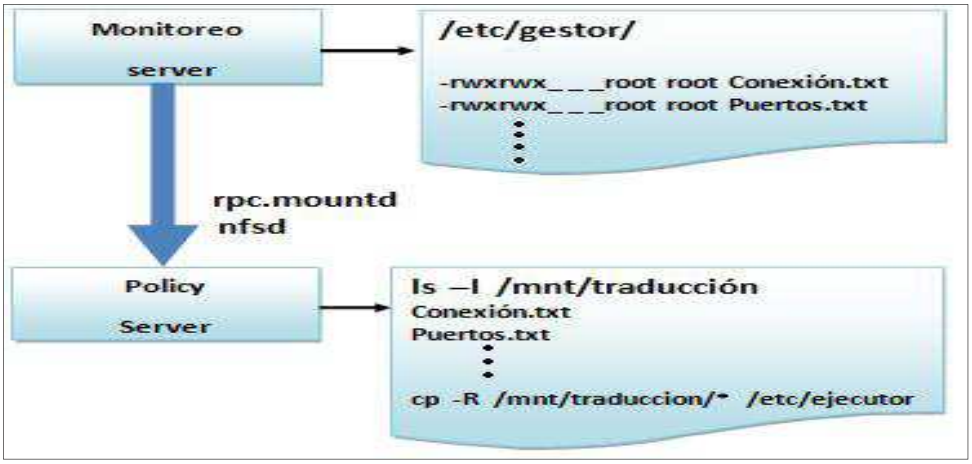



Figura 3. 12: Transferencia de archivos al módulo Ejecutor

Como se muestra en la Figura 3.12 se propone utilizar el protocolo NFS (Network File System) por los siguientes motivos:

- NFS es apropiado utilizarlo en una red de área local. Tanto el modulo gestor como el modulo ejecutor se encuentran en la red interna de Telydata.
- Para implementaciones no muy complejas, NFS no presenta mucha carga de procesamiento.
- El modulo gestor realizará actualizaciones constantes que inmediatamente deberán ser efectuadas sobre el modulo ejecutor. NFS posibilita la escritura y modificación inmediata de de archivos entre servidor y cliente.

Como se muestra en la Figura 3.12, el montaje simple requerirá de dos herramientas para un sistema operativo Linux.

La primera es *rpc.mountd* la cual ayudaría a satisfacer solicitudes de montaje. La segunda es el servicio *nfsd*, el cual ejecuta operaciones en el archivo actual (como lectura o escritura).

## 3.4. REQUERIMIENTOS ESPECÍFICOS DEL MÓDULO EJECUTOR

A continuación se especifican los requerimientos para cada componente:

a) Para el controlador de la capacidad de acceso[55]:

- Conjuntamente con el departamento comercial, se necesita especificar los valores de velocidades mínimas y máximas para cada usuario.
- En caso de que exista saturación se pueda garantizar una velocidad mínima, caso contrario alcance la máxima.
- En el controlador de la capacidad de acceso se necesita incluir el trafico analizado en la sección 2.6, del capítulo 2, y su priorización
- Priorizar puertos asociados a la navegación web, chat en línea, Macromedia flash que usan protocolos panto TCP y UDP. Además de las direcciones IP más frecuentes analizadas en la sección 2.6.2. del Capítulo 2.

b) Para los reinicios programados:

- Se requiere que se lo haga una vez por día, y solamente de las antenas principales pertenecientes al backbone tanto norte como sur, ya que son estas las que se encuentran operativas las 24 horas durante los 7 días a la semana.
- Además, para lograr el reinicio se requieren las direcciones IP, usuarios y contraseñas de las antenas pertenecientes al backbone norte y sur.

[55] ***Aclaración***: en el contexto comercial, también puede denominarse ancho de banda a un recurso de la red relacionado con la capacidad del canal para cada uno de los usuarios.

e) Para el componente NAT

- Se requiere que permita al módulo realizar la traducción y ruteo de direcciones IP desde su red interna, hacia el Internet.
- El servidor del módulo ejecutor al estar expuesto al Internet a través de una dirección IP pública requiere incluir un firewall básico.

En cuanto a la plataforma, el servidor que se vaya a instalar en el core de la red de Telydata, debe por política gerencial, ser parte de una plataforma de código abierto. Es decir, el diseño y la implementación deben tomar en cuenta el uso de Linux y sus diferentes herramientas para los tres componentes antes mencionados.

### 3.4.1 DISEÑO DEL MÓDULO EJECUTOR

El módulo ejecutor estará conformado por tres *script*s a desarrollar pertenecientes a cada componente, los cuales permitan satisfacer los requerimientos específicos anteriormente expuestos, por lo tanto, sus componentes son:

- *Controlador de la capacidad de acceso*: a través del *scriptControlador*.
- *Reinicios Programados*: a través del *reinicioScript.*
- *NAT*: a través de *scripttablasnat*

Previamente al diseño se establecen los perfiles de usuario y sus correspondientes valores de capacidad de acceso al canal que se necesitan controlar.

#### 3.4.1.1 Definición de perfiles de usuario respecto al canal de datos

Como se verificó en el capítulo 2, el WISP cuenta con dos tipos de usuarios: el uno es tipo Cyber y el otro es Residencial. Al no existir un control, ambos son tratados de la misma manera respecto al tráfico de datos y el acceso a la capacidad del canal.
La única diferencia entre los usuarios, desde el punto de vista comercial, es la capacidad de acceso contratada y su compresión.

*3.4.1.1.1. Cliente tipo Cyber*

Dado que este tipo de usuarios son los que más demandan el servicio, también representan un nivel mayor de ingreso económico para la empresa, la gerencia requiere de un proceso de mejoramiento del servicio y la definición de la calidad en ellos, por tanto el objetivo del negocio se enfoca en este tipo de usuarios. Otro punto importante que se toma en cuenta al dar prioridad a este tipo de usuarios, es su crecimiento a largo plazo en los sectores donde el acceso de Internet es limitado y en donde los distintos ISP existentes en el mercado no han podido llegar todavía.
La aplicación más utilizada por los Cyber es la navegación web en general, incluyendo a aplicaciones como: Youtube, Messenger, Facebook y Google para consultas en general. Por lo tanto, es necesario priorizar la navegación web a través del puerto 80, 8080, 443, de mensajería 1835 y 1963, aplicaciones multimedia con el puerto 1935.

*3.4.1.1.2 Especificación del Nivel de Servicio (SLS) para cliente tipo Cyber*

Se plantean los SLS, valores mínimos generales respecto al tráfico de la Red. Para ello, se considera la Tabla 3.4, la cual demuestra que existen cuatro tipos de velocidades comercializadas en Telydata para usuarios Cyber.

| Servicio | Velocidad ( *Downstram*[56] / *Upstream*[57] ) | # de Computadoras |
|---|---|---|
| **Cyber** | 950 / 300 Kbps | Máximo 6 |
| **Cyber** | 1300 / 300 Kbps | Máximo 8 |
| **Cyber** | 1800 / 300 Kbps | |
| **Cyber** | 2200 / 300 Kbps | Maximo12 |

Tabla 3. 4: Comercialización de velocidades Clientes Cyber[58]

Para mantener la calidad mínima respecto a la capacidad de acceso de cada tipo de clientes Cyber, se plantea la Tabla 3.5 que indicará valores mínimos y máximos de la

[56] Downstream: velocidad con la que los datos son transferidos desde el proveedor al cliente, denominada también velocidad de bajada.
[57] Upstream: velocidad con la que los datos son transferidos desde el cliente al proveedor, denominada también velocidad de subida.
[58] Información provista por la Srta. Leonela Tapia, Departamento Comercial Telydata.

capacidad para cada uno de ellos, con asignación *Downstream*. Hay que recalcar que la Tabla 3.5 también obedece a políticas comerciales en cuanto a costo beneficio por cliente. Se establece como política un límite mínimo de 600 Kbps para clientes Cyber Tipo A, ya que ningún cliente de la empresa debe tener una velocidad menor a 600 kbps. También se tiene la posibilidad de que si un cliente tiene asignado una capacidad de acceso al canal diferente a los que se indica a la Tabla 3.4, se asigne automáticamente un valor mínimo para su enlace.

| **CLIENTE CYBER** | **Especificaciones de la capacidad de acceso (asignación *Downstream*)** | |
|---|---|---|
| | Mínimo | Máximo |
| **Tipo A** | 600 Kbps | 950 Kbps |
| **Tipo B** | 951 Kbps | 1300 Kbps |
| **Tipo C** | 1301 Kbps | 1800 Kbps |
| **Tipo D** | 1801 Kbps | 2200 Kbps |

Tabla 3. 5: Especificaciones de velocidad para Clientes Cyber Downstream

Otro aspecto que hay que recalcar según la Tabla 3.4 establecida por el departamento comercial, es que la velocidad de subida es fijada a 300 Kbps para cada uno de los clientes Cyber, por lo que se establece también como política comercial asignar una velocidad de 350 Kbps como velocidad de subida máxima. Por lo tanto, se plantea la Tabla 3.6 que indicará los valores mínimos y máximos de la capacidad de acceso para cada uno de los clientes Cyber con asignación Upstream

| **CLIENTE CYBER** | **Especificaciones de la capacidad de acceso (asignación *Upstream*)** | |
|---|---|---|
| | Mínimo | Máximo |
| **Tipo A** | 300 Kbps | 350 Kbps |
| **Tipo B** | 300 Kbps | 350 Kbps |
| **Tipo C** | 300 Kbps | 350 Kbps |
| **Tipo D** | 300 Kbps | 350 Kbps |

Tabla 3. 6: Especificaciones de velocidad para Clientes Cyber Upstream

*3.4.1.1.3 Cliente tipo Residencial*

Este tipo de usuarios son minoría respecto al total que hay al momento en Telydata dentro de su red inalámbrica. Por tanto, representan menor demanda y consumo, el número de computadoras de red interna del cliente es menor e intervalos de saturación menores.
Las aplicaciones que usan este tipo de usuarios son similares a las que usan los clientes tipo Cyber, sin embargo, brindan mayor énfasis al consumo de aplicaciones de correo electrónico y chat en línea. A estos clientes se provee el servicio de Internet contratado sin ningún tipo de gestión de acceso a la capacidad del canal.

*3.4.1.1.2 Especificación del nivel de servicio para cliente Residencial (SLS)*

En coordinación con el departamento comercial[59] y con el fin de diferenciar a este tipo de usuarios respecto a los Cyber, no se tendrá un tratamiento especial sobre el tráfico, ni tampoco tendrá priorización del mismo. Estos clientes no demandan mayor capacidad del canal, poseen menor número de computadores en su red interna y su horario de saturación generalmente es en las noches. También cabe recalcar que actualmente solo hay este tipo de clientes para el enlace Iñaquito-Norte.

### 3.4.1.2 Diseño del controlador para la capacidad de acceso de los clientes

Se plantea que el controlador se instale sobre un servidor Linux, el cual esté constituido por las herramientas de iproute2, en específico de iptables y tc. El servidor deberá tener conexión con la RIT tal como se muestra en la Figura 3.13, la cual indica que los paquetes que cruzan a través del servidor deberían ser marcados con las herramientas *iptables* y *tc*. La herramienta tc aplica los filtros y asigna clases para el manejo de tráfico. El controlador asignará una parte de la capacidad de acceso al canal a cada cliente (representado por su dirección IP) y tendrá un valor fijo con la oportunidad de que un cliente esté en la posibilidad de tomar prestada capacidad del canal de otro cliente (si este no lo ocupa).

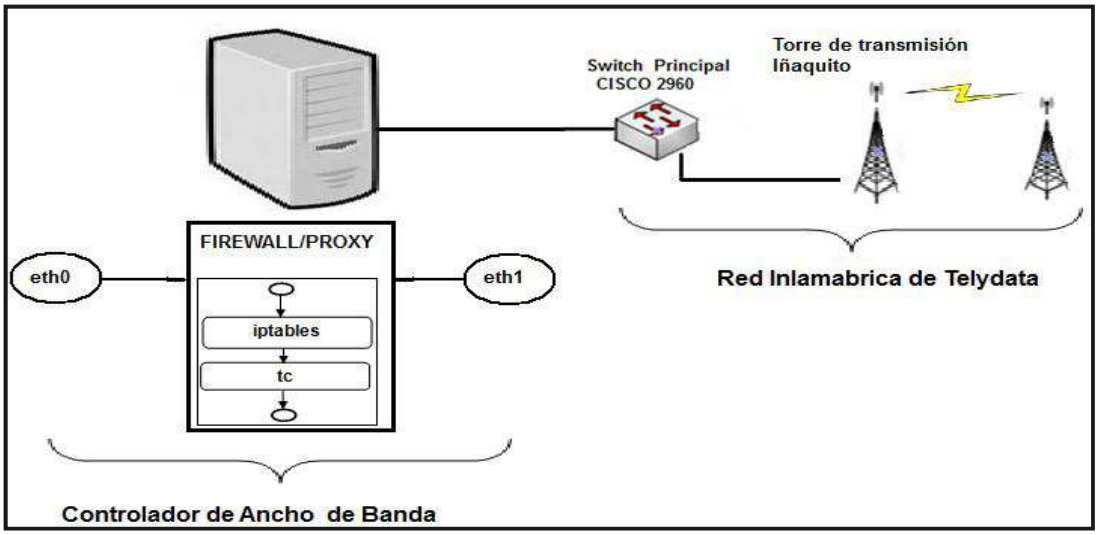



Figura 3. 13: Descripción general del controlador de la capacidad de acceso

*3.4.1.2.1 Mecanismo propuesto para el control de la capacidad de acceso y priorización del tráfico*

Se plantea construir un árbol básico de clases, uno será para la interfaz de bajada y el otro será para la interfaz de subida del servidor. En la Figura 3.14, se esquematizan los arboles propuestos.

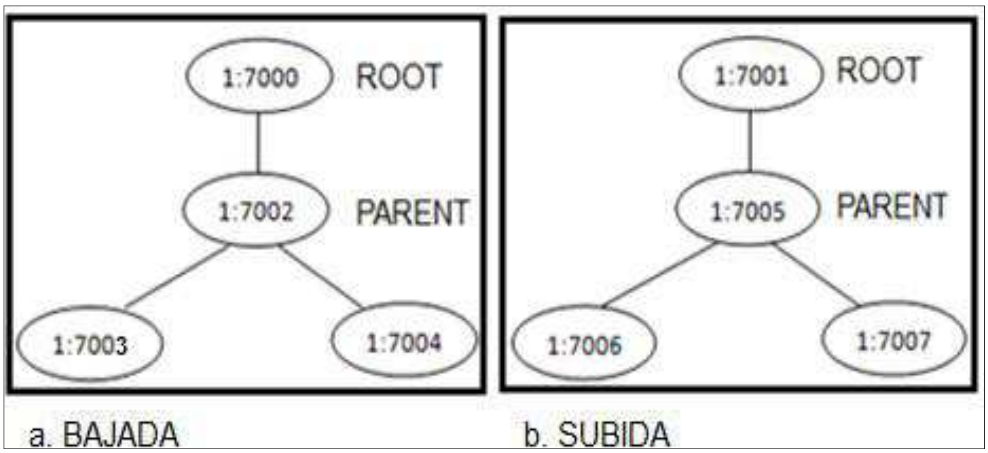



Figura 3. 14: Diseño del Árbol de clases para el control de tráfico

Tomando como ejemplo la Figura 3.14, ambos árboles muestran la estructura de clases para un solo usuario representado por una dirección IP. Para el cuadro del

literal a), la clase 1:700 es la clase raíz, a la que se le asigne la velocidad total que soporta el enlace de la RIT. A la clase 1:7002 se le asigna únicamente la velocidad de un cliente en específico (representado por su dirección IP). Dicha clase tendrá dos clases hijas: 1:7003 y 1:7004. La clase 1:7003 será la banda que lleva mayor prioridad (asignándole el valor de 1), por lo tanto, tendrá mayor porcentaje de velocidad del que fue asignada al usuario, con el fin de que en ella se asignen los puertos y conexiones IP a priorizar. La banda 1:7004 será la banda de menor prioridad (asignándole el valor de 3) y la que tenga el menor porcentaje de velocidad de la asignada al cliente para manejar tráfico residual o no importante.

Para el grafico del literal b), la explicación es análoga a la realizada con la interfaz de subida (literal a)). La clase raíz es denominada ROOT y siempre se mantiene con el mismo valor. La clase denominada PARENT se asigna con tantos valores conforme direcciones IP se tengan. Se podría elegir cualquier número para las clases, sin embargo se ha puesto como ejemplo al valor 7000, ya que es un número grande y fácilmente diferenciable de valores pequeños. Por otro lado, se tienen disponibles algoritmos para lograr la implementación de dicho árbol, como es el caso de HTB (*Hierarchical Token Bucket*) y SFQ (*Stochastic Fairness Queuing*)[59], los cuales deberían ser definidos con las herramientas tc de Linux.
De esta manera se plantea controlar la velocidad que se asigna a cada usuario a través de su dirección IP, y al mismo tiempo darle más prioridad al tráfico que se disponga

#### *3.4.1.2.2 Partes del controlador*

Se plantean once partes para definir el controlador de capacidad. A cada parte se puede nombrar una variable (ubicada entre paréntesis) como se indica a continuación:

[59] SFQ, definido en la sección 1.3.2.1.1

1. Interfaz Externa (*eth0*). Es la interfaz del servidor que tendrá salida al Internet. Se conectará con el switch de Core del WISP.

2. Dirección IP de la interfaz externa. Es aquella dirección IP asignada para la interfaz externa

3. Interfaz Interna (*eth1*). Es la interfaz del servidor que se conecta con las redes inalámbricas internas y brinda conexión a los clientes. Para ello se debería conectar con el switch de distribución del WISP.

4. Dirección IP de la interfaz interna. Es aquella dirección IP asignada para la interfaz interna.

5. Velocidad de Subida (*total_velocidad_up*): Es la velocidad máxima de subida establecida a nivel del backbone inalámbrico.

6. Velocidad de Bajada (*total_velocidad_down*): Es la velocidad máxima de bajada que establecida a nivel a nivel del backbone inalámbrico.

7. Clase inicial (*clase_inicio*): es la clase inicial que se asigna por defecto. A partir de este valor se asignarán el resto de clases según el árbol de la Figura 3.14.

8. Direcciones IP Clientes: Son aquellas direcciones IP configuradas en las estaciones inalámbricas de cada cliente y que son provistas desde el repositorio.

9. Velocidad Contratada: Es la velocidad asignada a cada cliente según lo acordado con el departamento comercial. Dato provisto desde el repositorio. Se tendrán cuatro tipos:

    - Velocidad garantizada de bajada (*velocidad_down*): es la velocidad fija que se debería entregar al usuario; aun en caso de saturación.

- Velocidad tope de bajada (*tope_down*): es la velocidad de bajada máxima con la que el usuario podrá navegar, el valor de lo contratado.

- Velocidad garantizada de subida (*velocidad_up*): es la velocidad fija de subida que se debería entregar al usuario; aun en caso de saturación.

- Velocidad tope de subida (*tope_up*): es la velocidad de subida máxima con la que el usuario podrá navegar, el valor de lo contratado.

Los valores de referencia específicos se encuentran puntualizados en las Tablas 3.5 y 3.6, según corresponda.

10. Puertos (*puertoprio*): la colección de puertos usados con mayor frecuencia en la red, y a los que se necesitaría asignar prioridad. Los valores son provistos desde el repositorio. Sus valores provienen de la Sección 2.6.

11. Conexiones IP (*iprio*): los grupos de direcciones IP en el Internet usados con mayor frecuencia, y a los que se pretende sean priorizados. Esta información es provista por el repositorio. Sus valores provienen de la Sección 2.6.2.

*3.4.1.2.3 Diseño del script bash controlador*

El programa principal se formará de varias funciones que permiten la obtención de datos y ejecución de reglas para el tratamiento de tráfico. En la Figura 3.15 se expone el flujograma del programa principal que se pretende realizar.

A continuación se realiza la declaración de funciones del *SCRIPT BASH*:

- Cargar la configuración (*cargar_conf ( )*): permitirá cargar los datos a variables globales con el fin de utilizarlas en el resto de funciones. La información de los

datos de velocidad y dirección IP vendrán del archivo denominado controlador-velocidades.conf y los valores de las clases vendrán de otro denominado controlador.conf. Se listan los datos que se requieren cargar:

- Dirección IP del cliente
- Velocidad de bajada
- Velocidad tope de bajada
- Velocidad de subida
- Velocidad tope de subida
- Asignación de valores para las clases y marcas (de acuerdo a la Figura 3.14). Estas clases y marcas necesitan ser hechas para eth1 y eth0.

- Cargar conexiones IP (*cargar_conexionesIP ( )*): esta función leerá del archivo ConexionesIP.txt, que contiene a todas las direcciones IP en Internet a las que se quiere dar prioridad.

- Cargar puertos (*cargar_Puerto()*): esta función leerá del archivo PuertosRed.txt todos los puertos relacionados a las aplicaciones que se requieren asignar prioridad.

- Marcado y Priorización (*configuracion_iptables ()*): se encargará de realizar el marcado de paquetes con la herramienta iptables de Linux. Las marcas serán asignadas específicamente a los puertos y direcciones IP cuyos valores han sido provistos desde la función cargar_conf (). Para ello esta función debería tener dos sub funciones:

  - *Iptables_alfa ():* la cual deba implementar las tablas generales para el nombramiento de reglas y reenvío de paquetes.
  - *Iptables_beta ():* por medio de las reglas ya nombradas, realiza el marcado con los valores que ya han sido almacenados en la función *cargar_conf().*

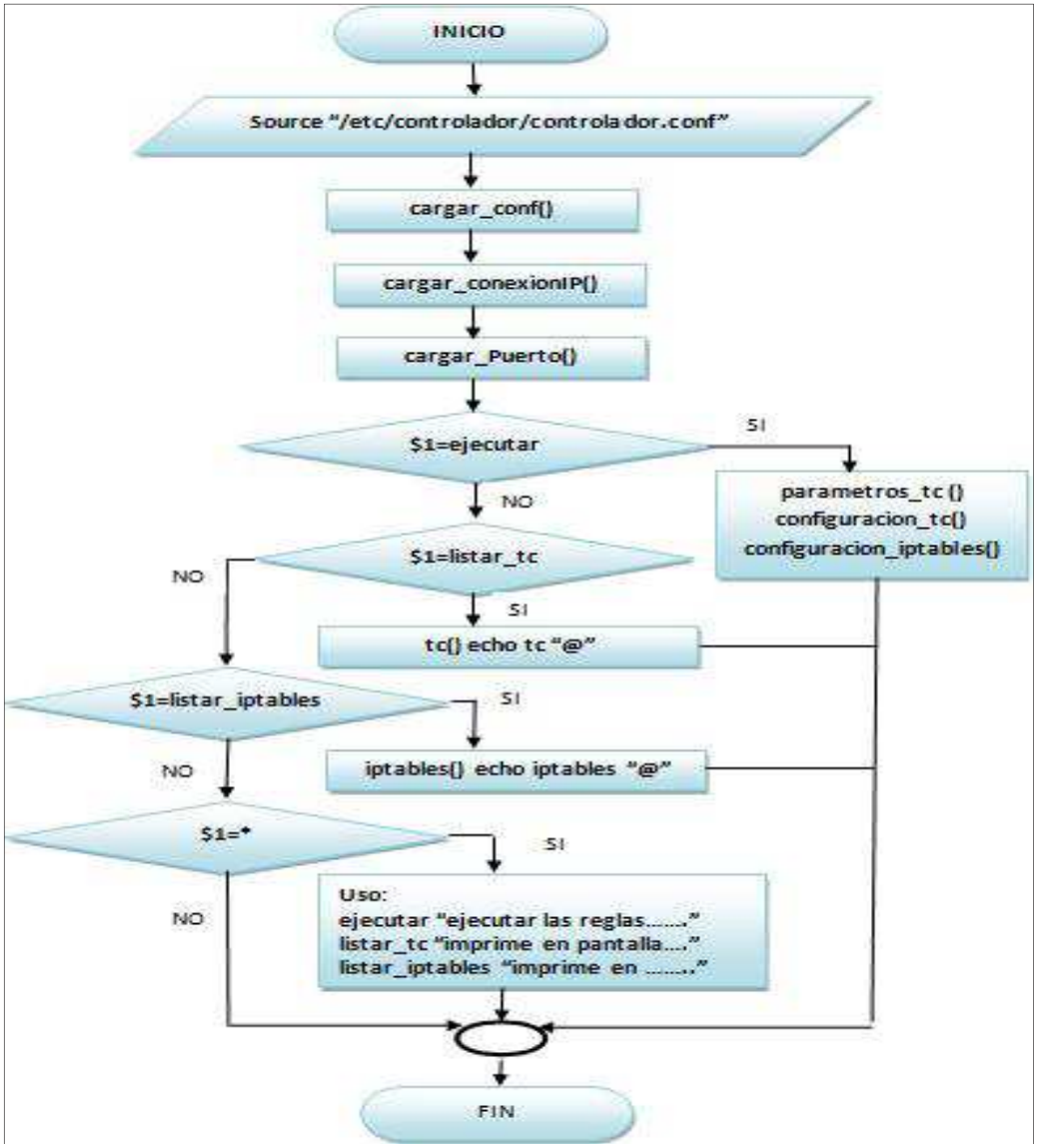



Figura 3. 15: Flujograma del *script* controlador

- Parámetros requeridos para tc (*parametros_tc ( )*): se plantea esta función para posibilitar la configuración del esquema HTB en el controlador de capacidad del canal. Se definirán tres clases (de acuerdo a la de Figura 3.14): ROOT, PARENT, LEAF. Cada clase debería obtener los siguientes los siguientes datos:

- *rate*: velocidad asignada como mínima para cada cliente. Sección 3.4.1.2.2. Literal 9. Velocidad garantizada.
- *ceil*: representa la velocidad máxima que se asignará para cada cliente. Sección 3.4.1.2.2. literal 9. Velocidad tope.
- *burst y cburst*: es máxima cantidad de datos (medida en bytes) que pueden ser enviados antes del arribo de nuevos *tokens*. Su valor depende de las características que soporta el buffer[60] en la arquitectura Linux que se esté utilizando. A continuación se demuestra un ejemplo del cálculo de *burst* para una arquitectura Linux (plataforma i386) [44]:

$$Burst = max\_rate * timer\_resolution$$

Donde:

*max_rate*: es la velocidad máxima a la que se pueden desencolar datos desde la interfaz de red. (10 Mbps).

*timer_resolution*: marca de tiempo límite del procesamiento para el kernel Linux.(10 ms).

$$Minimo\ Burst = \frac{10\ Mbit}{seg} * \frac{1\ byte}{8\ bits} * 10 * 10^{-3} seg \approx 12\ kbytes$$

- *Clase Padre*: se refiere a la clase PARENT de la Figura 3.14.
- *Interfaz*: hace referencia a las dos interfaces: eth0 o eth1.
- *Prioridad*: asignará valores a las bandas más prioritarias y menos prioritarias.

- Configuración de TC (*configuración_tc ()*): función que se encargará de asignar las clases, filtros, y disciplina de colas con la herramienta tc de iproute2. Cumplirá con los siguientes aspectos:

[60] Buffer: Ubicación de memoria reservada para el almacenamiento temporal

- Creación del árbol de la Figura 3.13. Debería crear un árbol por cada IP (cliente).
- Se encarga de la asignación de filtros, quienes asignarán los paquetes a las distintas clases respectivas. Esto también permitirá posteriormente realizar un marcado a través de la función iptables()

Será importante que el *script* se pueda ejecutarlo según el parámetro de entrada, con el fin de tener opciones de impresión en pantalla cuando se requiera obtener los resultados que se han generado:

- ejecutar: ejecuta las tres funciones que comprenden el controlador de capacidad: cargar_conf(), configuración_tc(), configuracion_iptables()
- listar_tc: imprime las clases y disciplinas de colas que se generen
- listar_iptables: imprime las reglas creadas con iptables.
- * (representa cualquier valor ingresado como parámetro): imprime una pequeña leyenda acerca de lo que realiza cada parámetro

#### 3.4.1.3 Diseño de los reinicios programados

Los reinicios se realizarán una vez por semana de manera automática. El departamento técnico ha determinado que el reinicio sea durante la madrugada, a las 4 am, con el fin de no provocar cortes en el servicio. El departamento técnico también recomienda que el día de reinicio sean los miércoles de cada semana, sin embargo existe la posibilidad de cambiarlo según el atributo pcimTPCDayOfWeekMask. El valor del atributo es un binario de siete dígitos. En la Figura 3.16 se representa un ejemplo.


Figura 3. 16: Valor del atributo pcimTPCDayOfWeekMask

En donde:

- Cada digito corresponderá a un día de la semana, empezando desde el domingo que se le asigna en la primera posición de izquierda a derecha (MSB, Bit Más Significativo).

- Si el digito tiene el valor de 1 en cualquier posición, el reinicio de las antenas se realizará en el día que corresponda a esa posición.

No puede haber más de un digito que tenga asignado el valor de 1, ya que el reinicio se lo requiere solo una vez por semana.

*3.4.1.3.1 Diseño del script bash reinicio*

Para realizar el reinicio de una antena se requiere de: a) Dirección IP, la cual representará al dispositivo al cual se realizará el reinicio. b) Fecha de ejecución, según el día y hora especificada, c) Usuario y contraseña de ingreso del dispositivo WiFi Ubiquiti Networks vía CLI.

Se propone el uso del protocolo SSH (*Secure Shell*), el cual permite acceder a máquinas remotas de forma segura, además porque es soportado por las antenas Ubiquiti Networks para acceso remoto. A continuación se indica el flujograma del *scriptReinicio*:


Figura 3. 17: Diseño del *scriptReinicio*

#### 3.4.1.4 Administración y ejecución de los distintos scripts bash

Se tendrán tres *script*s para el funcionamiento del sistema. Para ello se plantea el uso del administrador de procesos *cron* de Linux. La programación de tareas se realizará en base a los siguientes criterios:

1. En el caso de que exista interferencias, se realizan cambios de datos de la frecuencia de las antenas en al menos una vez por día.
2. Peticiones de usuarios para realización de aumentos de capacidad de acceso al canal. Al menos una vez por día.
3. Actualizaciones de datos técnicos como direcciones IP, ccq, nivel de señal. Al menos dos veces por día.
4. Instalaciones nuevas, ingreso de nuevos datos de clientes al realizar nuevas instalaciones del servicio de Internet. Por lo menos cinco mensuales.
5. Clientes que se retiran. No existe tiempo de frecuencia estimado
6. Respecto a los reinicios programados, el departamento técnico de Telydata lo realiza por lo menos una vez por semana.

En consecuencia se plantea la ejecución de *script*s de la manera presentada en la Tabla 3.7.

| ***Script*** | **Tiempo de Ejecución** |
|---|---|
| **DatosClientes** | Dos veces por día. Primera: 12h00, Segunda: 22h00 |
| **Controlador** | Dos veces por día. Primera: 12h00, Segunda: 22h00 |
| **Reinicios** | Un día por semana. Por defecto son los miércoles a las 04h00, con posibilidad para cambiar de día |

Tabla 3. 7: Asignación del horario de ejecución de los *Script*s

## 3.5 RESTRICCIONES DEL SISTEMA

El repositorio de información está básicamente limitado por la forma en cómo se define los atributos y sus valores, ya que para su correcto funcionamiento se debería cumplir con los formatos ejemplos establecidos desde la sección 3.3.1.2. No existe mecanismo riguroso de validar los datos previamente a ser guardados. Por otro lado, se debería tener el enlace de última milla en buenas condiciones para que se pueda cumplir con los objetivos establecidos respecto a la capacidad de acceso al canal. En varias ocasiones se tienen causas externas que afectan al enlace, como por ejemplo, cortes de energía eléctrica, factores climáticos que provocan ruido e interferencias en la transmisión de datos.

## 3.6 DETERMINACIÓN DE LA PLATAFORMA PARA EL MÓDULO GESTOR

La herramienta software para el repositorio de información que incluye al servidor LDAP debe cumplir con los requerimientos de la sección 3.3.

Existen aplicaciones tanto de licencia libre como comercial que tienen múltiples funcionalidades para servicios de directorios.

### 3.6.1 OPENLDAP [45]

Es una implementación para LDAP de código abierto, que abarca procesos de ingeniería sobre múltiples plataformas, incluyendo Linux, Solaris, Mac OS 10.2, y Windows. Es el servidor por defecto en Linux está básicamente compuesto de los siguientes paquetes:

- Servidor ldap (slapd)
- Servidor de replicación LDAP (slurpd)
- Kit de desarrollo de software (ldap)
- Utilidades, herramientas y ejemplos

También se compone de una suite de herramientas cliente LDAP estándares. Los principales lanzamientos funcionales de openLDAP incluyen:

- OpenLDAP v1, fue una recopilación general de la última versión de un proyecto desarrollado en la universidad de Michigan y la consolidación de cambios adicionales.
- OpenLDAP v2, lanzada con mejoras importantes, incluyendo soporte para Internet Protocol v6 (IPv6) en Agosto del 2000, en backend implementa la base de datos transaccional basada en Berkeley Database (BDB). Soporte para Simple Autenticación y Seguridades, así como otros backends experimentales. También se incluyó el motor de sincronización con soporte de replicación, interfaz de presentación y otras mejoras a nivel de base de datos. Para Junio del 2005, se introdujo la aplicación en N-vías MultiMaster, Stand-by master con la posibilidad de modificar y eliminar elementos del esquema en tiempo de ejecución.

### 3.6.2 LOTUS DOMINO [46] [47]

Es una plataforma propietaria de IBM, con capacidades para desarrollar directorios empresariales a través de una interfaz LDAP, especialmente para el trabajo de correo electrónico, incluyendo POP3 e IMAP, mensajes mejorados en seguridad y servicios web. El repositorio de información, puede tener una base de datos DB2 (base de datos relacional nativa de IBM). Lotus Notes es un sistema cliente servidor para coordinación empresarial y correo electrónico. En su plataforma, la parte del servidor recibe el nombre de Lotus Domino, mientras que el cliente recibe el nombre de Lotus Notes.

### 3.6.3 APACHE DIRECTORY SERVER [48]

Es un servidor diseñado para una plataforma LDAP y X.500, programado completamente en java bajo la licencia de Apache Software. Este proyecto ha sido certificado con LDAP v3 en dos áreas específicas; en OpenGroup (*Apache DS*), y en

las herramientas de directorio basadas en Eclipse (*Apache Directory Studio*). El Apache Directory Studio cuenta con herramientas potenciales tales como:

- LDAP browser: el cual que permite administrar la información del DIT.
- LDIF[61] Editor: el cual permite editar y estructurar archivos LDIF.
- Apache LDAP API: las interfaces de programación para LDAP, usan el lenguaje de programación orientado a objetos para java
- Schema Editor: herramienta para editar archivos de esquemas en el formato definido por RFC 4512.

Una característica importante es que la aplicación soporta múltiples plataformas entre estas se encuentran: Mac OS X, Linux y Windows.

### 3.6.4 FEDORA DIRECTORY SERVER [49]

Un servicio de directorio basado en LDAP inicialmente desarrollado por Tim Howes y Mark Smith denominado como Netscape Directory Server. En el año 2005 se lo denominó como Fedora Directory Server, siendo una aplicación muy robusta que implementa las dos especificaciones de LDAP (v2 y v3). Soporta múltiples bases de datos y presenta mayor disponibilidad de servicio. Una de las ventajas principales es que dispone de administración gráfica y mediante consola.

### 3.6.2 ACTIVE DIRECTORY [50]

Es un directorio de propósito especial diseñado para administrar la información de diversos objetos como impresoras, usuarios y equipos; mantiene numerosas operaciones de lectura y consultas frente a un menor número de cambios y actualizaciones. El nombre Active Directory fue utilizada por Microsoft, a partir de Windows 2000 para representar un almacén centralizado de información de uno de sus dominios de administración. Esta aplicación se encuentra fundamentada en un

[61] LDIF(LDAP Data Interchange Format): es una representación simple formada por un número indeterminado de pares de atributo y valor.

esquema LDAP versión 3, por lo que permite integrar otros sistemas que soporten el protocolo. También soporta todo lo concerniente las políticas de seguridad.

### 3.6.3 ANÁLISIS COMPARATIVO Y ELECCIÓN DE LA HERRAMIENTA

Las dos aplicaciones principales que presentan mayores fuentes de documentación y han sido ampliamente desarrolladas son: Active Directory y openLDAP. Por lo tanto serán tomadas a consideración en el presente estudio.
Se plantean los siguientes criterios a evaluar:

- Características administrativas: que se identifique la confiabilidad y viabilidad de la administración del directorio a implementar.
- Rendimiento: que el producto haga referencia a los resultados deseados.
- Costos: será importante referenciar el costo propio del producto, así como también considerar los costos que conlleva el soporte y mantenimiento de los mismos.
- Integración con la Infraestructura de la RIT: es indispensable que el producto pueda integrarse fielmente con la infraestructura con la cual se opera actualmente en la RIT.

Ponderación de criterios

Se han establecido ponderaciones numéricas como base comparativa de acuerdo a las características específicas de cada criterio. Cabe señalar que al no tener la suficiente experiencia de las herramientas openLDAP y Active Directory, ciertos factores tienen un componente subjetivo que se ha intentado reducir al mínimo usando tablas de puntuaciones, y habiendo obtenido previamente fuentes fidedignas de información presentadas a continuación:

- Tesis: ROMERO, Célida Fabiola “Análisis comparativo entre productos que proveen servicio de directorio pertenecientes a tecnologías propietaria y de

libre acceso, aplicado a laboratorios en ambientes educativos. Escuela Politécnica del Litoral. Ecuador, 2008.

- Guía detallada de administración de Active Directory [51].
- Guía del Administrador OpenLDAP Software 2.4 [52].

Cada criterio mencionado al inicio de esta sección tendrá valores que determinan la ponderación y varían del 1 al 5, tal como se muestra en la Tabla 3.8.

Para lograr simplificar la nomenclatura, Active Directory se abrevia como AD, mientras que OpenLDAP se abrevia como OL.

| Valor | Descripción |
|---|---|
| **1** | No cumple con la característica en ningún sentido |
| **2** | Cumple con la característica con deficiencias |
| **3** | Cumple con la mitad de la característica |
| **4** | Cumple con la característica pero con alguna restricción |
| **5** | Cumple con la característica a cabalidad |

Tabla 3. 8: Ponderación de criterios para Servidor LDAP

#### 3.6.3.1 Características Administrativas.

Se debe considerar que se tenga ambiente gráfico, control de acceso simple, escalabilidad en cuanto a costo y posea utilidades de respaldo.

Según la Tabla 3.9, se describe:

- *Ambiente Grafico*.- Tanto AD como OL pueden trabajar ambiente grafico para la administración de la información, sin embargo OL no cuenta con una herramienta grafica nativa del producto provisto por OpenLDAP Foundation[62], primordialmente su utilización está basada en terminal.

[62] Fundación que promueve el desarrollo LDAP Open Source y coordina sus actividades

| Criterio | AD | OL |
|---|---|---|
| | Evaluación | Evaluación |
| **Ambiente Grafico** | 5 | 4 |
| **Manejo de Control de Acceso** | 5 | 5 |
| **Escalabilidad** | 4 | 5 |
| **Documentación** | 5 | 5 |
| **Utilidades de Respaldo** | 5 | 4 |
| **Utilidades de Restauración** | 4 | 5 |
| **Viabilidad y facilidad de Administración** | 5 | 3 |
| **TOTAL** | **33** | **31** |

Tabla 3. 9: Criterios a evaluar respecto a las características administrativas

- *Control de Acceso*.- Cada uno de los productos disponen de control de acceso básico y mecanismos de seguridad para acceder a la información del directorio, el más utilizado es a través de listas de control de acceso (ACL), las cuales consisten en listas que otorgan permisos a un objeto, así como las operaciones que están permitidas sobre un objeto.

- *Escalabilidad*.- se pretende establecer que tan factible es utilizar el producto y si las implementaciones futuras de mayor escala pueden crecer conforme la demanda de la empresa sin realizar costosas modificaciones. El producto que ofrece mayor escalabilidad es OL, ya que la compañía fabricante de AD expone que por un determinado número de máquinas se necesita establecer una nueva controladora de dominio y por ende considerar gasto incurrido.

- *Documentación*.- OL y AD tienen suficiente documentación.

- *Utilidades de Respaldo.*- Cada uno de los productos posee utilidades de respaldo de información. Se tiene la limitante de que en OL para algunas operaciones el servicio debe ser detenido antes de realizar un proceso de

respaldo, mientras que en AD se tiene la utilidad denominada ntbackup, lo cual es posible realizar dicho proceso mientras el servicio se está ejecutando.

- *Utilidades de Restauración.-* OL es mejor debido a que no conlleva pasos exhaustivos con el sistema operativo, tal como lo efectúa AD. OL posee la herramienta slapd y el archivo ldif que aumentan eficiencia en el proceso de respaldo.

- *Viabilidad y facilidad de Administración*.-OL se torna más difícil de administrar ya que la sintaxis para creación, modificación y eliminación de usuarios o grupos del directorio es más complejo si no se tienen conocimientos previos. Además, si no se cuenta con un Browser gráfico externo para la administración se debe utilizar la herramienta terminal.

#### 3.6.3.2 Costos

Se debe considerar que el software sea gratuito y que el costo tanto de la implementación como del mantenimiento del sistema donde funcione la aplicación sea relativamente rápido, sin tener mucha complejidad, ya que se va a trabajar en un ambiente de producción.

Según la Tabla 3.10, se describe:

| Criterio | Active Directory | OpenLDAP |
|---|---|---|
| | Evaluación | Evaluación |
| **Costo del Software** | 1 | 5 |
| **Costo del Hardware** | 5 | 5 |
| **Costo de Implementación** | 5 | 4 |
| **Costo de Mantenimiento** | 5 | 4 |
| **TOTAL** | 16 | 18 |

Tabla 3. 10: Criterios a evaluar respecto a los Costos

- *Costo del Software.-* El costo de adquisición del software marca la diferencia entre AD con OL, ya que es necesario comprar el sistema operativo sobre el cual residen. AD viene incluido en Windows 2003 o 2008 Server como un servicio a ser instalado posteriormente por el administrador de red. El costo de este sistema operativo se ve afectado por el número de Licencias asociadas al servidor. OL viene incluido en varias distribuciones del sistema operativo Linux como un paquete de instalación. La licencia es pública y sin costo alguno.

- *Costo del Hardware.-* depende de la arquitectura que cada uno de los sistemas operativos anfitriones además de la carga en procesamiento y capacidad que se va a demandar con el servidor de directorios.

- *Costo de Implementación*.- se considera el costo de efectuar el proceso que soporta la instalación, configuración y pruebas respectivas sobre el directorio, la implementación en AD resulta un proceso no muy extenso, basado en el tiempo que conlleva realizarlo. AD posee una interfaz gráfica que permite realizar la implementación a través de asistentes con la ayuda respectiva, por ejemplo Microsoft TechNet Learning Center. En cambio OL requiere conocimientos básicos de programación para las configuraciones de archivos de texto.

- *Costo de Mantenimiento.-* este se ve reflejado en el pago del personal encargado, en éste caso al administrador de la Red, quien es el responsable del funcionamiento del servicio de directorio. El mantenimiento en OL puede resultar un poco más costoso ya que se requiere que el administrador conozca las distintas herramientas relacionadas del sistema operativo Linux.

### 3.6.3.3 Rendimiento

Se debe considerar que el acceso al repositorio de información, y el procesamiento de operaciones, sean lo más rápidas con el fin de tener un sistema eficiente.

Según la tabla 3.11, se describe:

| Criterio | Active Directory | OpenLDAP |
|---|---|---|
| | Evaluación | Evaluación |
| **Tiempo de Login** | 4 | 5 |
| **Manejo Simultaneo de Conexiones LDAP** | 5 | 5 |
| **Procesamiento de Operaciones** | 5 | 4 |
| **TOTAL** | 14 | 14 |

Tabla 3. 11: Criterios a evaluar respecto al rendimiento

- *Tiempo de Login*.- el tiempo involucrado en un inicio de sesión es importante si se tiene mucha demanda de usuarios sobre el servidor de directorios. Según el trabajo de investigación "Análisis comparativo entre productos que proveen servicio de directorio pertenecientes a Tecnologías propietaria y de libre acceso, aplicado a laboratorios en ambientes educativos" [53]; se determinó que bajo las mismas condiciones hardware, el producto OL tiene menor tiempo de respuesta.

- *Manejo simultáneo de conexiones LDAP*.- Ambos productos poseen buenas características de distribución y concurrencia.

- *Procesamiento de Operaciones*.-En la investigación referenciada [53], se describe que la tasa de búsqueda de AD es aproximadamente de 999 operaciones/seg con respecto a las 4.5 operaciones/seg de OL, considerando un solo cliente. Si se considera 10 clientes consultando al directorio, se tiene 2.199 operaciones /seg de AD versus 18.2 operaciones/seg de OL. En cuanto a las operaciones de modificación AD procesa 27.7 operaciones /seg, frente al

procesamiento de 9.3 operaciones/seg de OL. El documento indica que las pruebas han sido realizadas teniendo las mismas características hardware del sistema operativo anfitrión.

#### 3.6.3.4 Integración con la infraestructura de la RIT

Se debe considerar que el personal técnico esté familiarizado con el sistema operativo anfitrión sobre el cual resida la aplicación, y que dicho sistema operativo sea compatible con la plataforma de la RIT. Además que obedezca con la política del Gerente Técnico.

| Criterio | Active Directory | OpenLDAP |
|---|---|---|
| | Evaluación | Evaluación |
| **Experiencia del Personal Técnico** | 4 | 5 |
| **Plataforma de Operación** | 2 | 5 |
| **TOTAL** | **6** | **10** |

Tabla 3. 12: Criterios a evaluar respecto a la integración con la RIT

Según la Tabla 3.12, se describe:

- *Experiencia del personal Técnico*.- El personal técnico está capacitado para operar en cualquier tipo sistemas informáticos. Sin embargo, para el sistema operativo Linux, la distribución CentOS 5 es la herramienta de trabajo diaria y se tiene más experiencia en ella.

- *Plataforma de Operación*.- La plataforma de operación en el core de la red está compuesta por GNU/Linux. Cabe recalcar que la política empresarial también fomenta el uso de software libre.

### 3.6.4 ELECCIÓN DE LA HERRAMIENTA

Se realizará el análisis comparativo en base a las puntuaciones obtenidas al evaluar cada herramienta. En la Tabla 3.13 se presenta un resumen.

| Producto | Active Directory | OpenLDAP |
|---|---|---|
| **Características Administrativas** | 33 | 31 |
| **Costos** | 16 | 18 |
| **Rendimiento** | 14 | 14 |
| **Integración con la RIT** | 6 | 10 |
| **TOTAL** | 69 | 70 |

Tabla 3. 13: Puntaje total Criterios vs. Productos LDAP

Como se puede verificar en la Tabla 3.13, mayor puntaje total lo obtiene OpenLDAP, ganado principalmente en cuanto a costos e integración con la RIT. Active Directory tiene ventaja en cuanto a características administrativas, específicamente sobre la viabilidad y facilidad de administración porque no requiere sintaxis muy compleja. Sin embargo, no se considera que esto sea un factor limitante, ya que la implementación será un proceso de aprendizaje constante, contando previamente con conocimientos de programación básica de *script*s. En consecuencia, se selecciona al paquete OpenLDAP como herramienta que implemente el repositorio de información por ser mejor que Active Directory específicamente en términos de Integración con la RIT.

La herramienta OpenLDAP no posee un gestor grafico para facilitar la integración con el usuario, por lo tanto se debe elegir una interfaz gráfica para la administración del servidor.

### 3.6.5 DETERMINACIÓN DEL INTERFAZ DE ADMINISTRACIÓN PARA EL MÓDULO GESTOR

A continuación se describen algunos de los interfaces de administración compatibles y más conocidos para OpenLDAP.

#### 3.6.5.1 Jxplorer [54]

Es un explorador LDAP de código abierto originalmente desarrollado por eTrust del laboratorio de Computer Associates, basado en Java. Es un estándar de propósito general que puede ser usado para leer o investigar cualquier directorio LDAP, o

cualquier directorio X.500. También posee características de seguridad con soporte para SSL/TSL[63] y autenticación SASL[64], válido para sistemas operativos Linux, MacOS y Windows.

### 3.6.5.2 Softerra LDAP Browser [55]

Es un producto desarrollado por la compañía Softerra como una herramienta de administración LDAP amigable diseñada para trabajar con la mayoría de servidores incluyendo Active Directory y Novell Directory Services. Es un gestor de búsqueda avanzado que soporta operaciones de actualización y navegación avanzadas. Además, dispone de reportes de directorio personalizables. En su versión gratuita, no soporta modificación del árbol DIT. Permite gestionar las entradas LDAP usando la sintaxis básica de SQL y llevar a cabo operaciones de consulta avanzadas que no se logran a través de las herramientas LDAP básicas.

### 3.6.5.3 Apache Directory Studio LDAP Browser [56]

Soporta la plataforma LDAP y X.500. Fue programado e implementado completamente en Java bajo la licencia de Apache Software[65]. Es parte del proyecto Apache Directory Studio el cual ha sido certificado con LDAP en sus diferentes áreas, tanto en OpenGroup (*Apache DS*) como en las herramientas de directorio basadas en Eclipse (*Apache Directory Studio*). La característica más importante es que facilita la administración de la información en cualquier servidor LDAP, creando y modificando entradas en el árbol de información de Directorio (DIT).

### 3.6.5.4 Elección de la interfaz de administración

En la Tabla 3.14 se lista una serie de criterios que se requiere tomar en cuenta para la interfaz de administración. Las iniciales para las herramientas a analizar son:

[63] SSL/TSL (Secure Socket Layer/Transport Layer Security): provee confidencialidad en el transporte de datos y protección de la integridad para las conexiones LDAP

[64] SASL (Simple Authentication and Security Layer): mecanismo externo a LDAP que permite a un cliente y servidor adoptar mecanismos de autenticación.

[65] Es una licencia de software libre creada por la Apache Software Foundation(ASF), que exige al usuario que únicamente conserve el reconocimiento del código fuente a ASF.

- JX: abreviado para JXPLORER
- SLB: abreviado para Softerra LDAP Browser
- ADSLB: abreviado para Apache Directory Studio LDAP Browser

| Criterios | JX | SLB | ADSLB |
|---|---|---|---|
| **Licencia** | No Requerida | Requerida | Apache License |
| **Interfaz Gráfico** | Normal | Excelente | Avanzado |
| **Gestión de directorio** | SI | NO | SI |
| **Soporte de *Schemas*** | SI | SI | SI |
| **Documentación** | SI | SI | SI |
| **Versiones LDAP** | v2 | v2,v3 | v2,v3 |

Tabla 3. 14: Evaluación de la interfaz cliente LDAP

El criterio más importante que se requiere para la gestión del directorio es que tenga las tres operaciones básicas, añadir, borrar y modificar entradas o atributos y tener un buen método de filtrado de información. Apache Studio LDAP Browser cumple con dichas necesidades de tener un interfaz gráfico avanzado para la administración del árbol DIT. Por otra parte, Softerra LDAP Browser posee una amigable interfaz gráfica con un método de filtrado avanzado como el denominado LDAP-SQL[66], pero la desventaja de este explorador es que al usar su versión gratuita no permite modificar el Árbol DIT, aun así, será válida para usarla en computadores que estén fuera de la red de Telydata.

## 3.7. DETERMINACIÓN DE LA DISTRIBUCIÓN LINUX PARA EL MÓDULO EJECUTOR

Aparte de la política que presenta la gerencia técnica de usar el sistema operativo Linux, se presentan a continuación algunos aspectos que validan el uso de la plataforma Linux:

[66] SQL: (Lenguaje de Consulta Estructurado): es un lenguaje de base de datos normalizado y estandarizado que permite consultar y manejar información. LDAP-SQL permite manejar los datos almacenados en LDAP con el lenguaje SQL

- Para el manejo de tráfico posee herramientas potentes como iproute2 que ayudan a implementar aplicaciones como firewalls, funciones de NAT, controladores de tráfico y balanceo. Si en Windows se quisiera adoptar estas funcionalidades se tendría que comenzar desde cero a través de parches y productos propietarios, ya que no cuenta con herramientas para realizar estas labores.

- Es gratis, la gran mayoría de distribuciones de Linux son gratuitas pero un tanto complejas de manejar. Por otra parte, para no infringir la ley, no se puede usar software pirata, peor aún en un ambiente empresarial.
- Linux es más inmune frente a virus respecto a Windows, para dispositivos en producción se torna en un problema fuerte.
- Es más flexible en el entorno del diseño presentado en el presente capítulo, ya que se pueden corregir errores de programas en *script*s de manera relativamente rápida. Al ser el código libre y modificable se puede mejorarlo cada día.

### 3.7.1 DISTRIBUCIONES LINUX [57]

Se llama distribución a un conjunto de programas y ficheros recopilados, organizados y preparados para su instalación. Casi los principales distribuidores de Linux ofrecen la posibilidad de bajarse el código de Internet vía FTP (*File Transfer Protocol*).
Una distribución contiene básicamente el kernel Linux, bibliotecas y paquetes de software. Técnicamente para la elaboración de servidores, cualquiera de las distribuciones puede servir, sin embargo, algunas son preferidas para oficios preferidos, a continuación se listan algunas de ellas:

- Red Hat Enterprise Linux (RHEL) Server
- SuSE Linux Enterprise Server
- CentOS
- Debian

- Slackware
- Ubuntu.

### 3.7.2 ANÁLISIS COMPARATIVO Y ELECCIÓN DE LA HERRAMIENTA LINUX [58] [59]

A continuación se describe algunas distribuciones que podrían ayudar a la implementación: Debian, Slackware, CentOS o Ubuntu.

#### 3.7.2.1 Debian

Posee el mayor repositorio de software libre de todos los sistemas operativos. Es una distribución estable, madura y muy popular. Ofrece una excelente herramienta de gestión de paquetes APT (*Advanced Packet Tool*). Requiere mayor conocimiento para lograr configuraciones avanzadas. Cuenta con el formato de paquetes .deb, y los módulos se mantienen por medio de la gestión de paquetes con la herramienta dpkg. Una característica importante es que existen módulos de actualización que permiten eliminar dependencias obsoletas. Debido a su gran popularidad, es relativamente fácil obtener ayuda.

#### 3.7.2.2 Slackware

Posee un núcleo original de Linux, no parcheado y es básicamente la distribución más antigua, con una jerarquía de directorios clásica, su instalación es manual. No se tiene una buena gestión de dependencias, siendo esta su debilidad importante, los gestores rpm o dpkg están lejos de ser usados con facilidad ya que pueden causar problemas de dependencias redundantes[67], falta de dependencias o paquetes de conflicto.

[67] Dependencias que instala paquetes ya instalados en otra versión, o cadena de dependencias que se relacionen unas de otras.

#### 3.7.2.3 CentOS

Distribución cimentada en RHEL[68]. La totalidad de paquetes binarios están basados en RPM y recompilados desde las fuentes binarias, posee un comando potente de administración denominado *yum* para actualizaciones e instalación de programas. Para el funcionamiento como servidores se lo utiliza con o sin necesidad de una interfaz gráfica, sin interfaz gráfica puede ofrecer un mayor rendimiento. Una característica importante es que CentOS ofrece estabilidad y seguridad al trabajar por tiempos prolongados.

#### 3.7.2.4 Ubuntu

Distribución basada en la filosofía Ubuntu, la cual se fundamenta en que el software debe estar disponible gratuitamente, y que las herramientas deben ser usadas por gente en su idioma local y de forma simple. La gente puede estar en posibilidad de personalizar y modificar el software a la forma que se considere más apropiada. Se cuenta con un buen apoyo de comunidades a través de correo y foros en web. Una característica importante es que el sistema detecta automáticamente dispositivos móviles como memorias, USB, cámaras digitales y tarjetas de memoria, tecnologías como WiFi y ahorro de energía. Respecto a los sistemas de escritorio se tiene Ubuntu con KDE (Kubuntu), XFCE (Xubuntu), entre otros.

#### 3.7.2.5 Análisis comparativo y selección de la herramienta

En la Tabla 3.15 se presenta una lista de requisitos principales que deben ser satisfechos por la distribución Linux ordenados por ítems. Las tablas con el desglose de las características para cada distribución se encuentran en el **Anexo 5**.

[68] RHEL (Red Hat Enterprise Linux) Sistemas Operativos basados en Linux desarrollados por Red Hat para fines comerciales.

| Característica | Referencia | Debian | Slackware | CentOS | Ubuntu |
|---|---|---|---|---|---|
| **Software Libre** | Item1 | 15 | 25 | 25 | 15 |
| **Estabilidad** | Item2 | 25 | 25 | 25 | 20 |
| **Seguridad** | Item3 | 20 | 25 | 25 | 20 |
| **Velocidad de arranque del sistema** | Item4 | 25 | 25 | 25 | 25 |
| **Velocidad de respuesta del sistema** | Item5 | 15 | 20 | 25 | 20 |
| **Documentación y ayuda** | Item6 | 25 | 15 | 20 | 15 |
| **TOTAL PUNTOS** | | 125 | 135 | 145 | 115 |

Tabla 3. 15: Selección de la distribución Linux

La mejor herramienta para la implementación del módulo ejecutor es una infraestructura basada en Linux adoptando la distribución CentOS**.** Dicha distribución posee una mejor puntuación en la mayoría de características establecidas, especialmente en cuanto a estabilidad y velocidad de respuesta del sistema.

# CAPÍTULO 4

# IMPLEMENTACIÓN, PRUEBAS Y COSTOS

*La implementación se realiza sobre la infraestructura de acceso en la empresa Telydata, iniciando con la implementación del módulo gestor, luego con la implementación del módulo ejecutor. Se realiza la configuración de NFS para la compartición de Información. Posteriormente se estipula el costo referencial del proyecto y las pruebas correspondientes.*

## 4.1 TOPOLOGÍA DEL SISTEMA

El sistema está compuesto como lo indica la Figura 4.1. El ***módulo Gestor*** consta de un servidor Linux denominado Monitoreo Server, actualmente se encuentra operativo, funciona con CentOS 5.8 y su dirección IP es: 201.219.6.247/28. A este dispositivo se configurará el servidor LDAP. El ***Módulo Ejecutor*** está formado de tres elementos básicos: Un Servidor Linux al cual se le denominará Policy Server, un Ruteador Cisco 3725 (Apolo), y las diferentes antenas de la infraestructura Ubiquiti Network. Policy Server no existe actualmente en la red de la empresa.

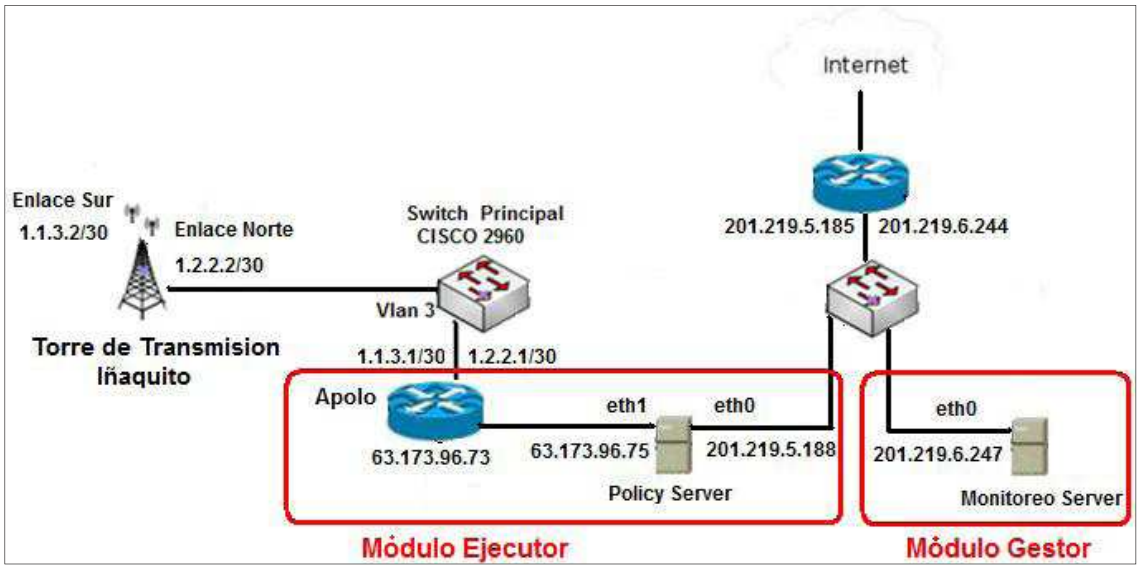



Figura 4. 1: Topología del sistema a Implementar

## 4.2 IMPLEMENTACIÓN DEL MÓDULO GESTOR

En este módulo se instala y configura un servidor LDAP con la interfaz gráfica para la administración de los datos establecidos en el modelo de información. También se implementa un *script* denominado *datosClientes* para el mapeo de la información.

### 4.2.1 REQUERIMIENTOS DE HARDWARE DEL SERVIDOR LDAP

El servidor LDAP estará constituido por la herramienta OpenLDAP. Antes de proceder con su instalación y configuración se analiza el estado actual de la PC Monitoreo para saber si tiene condiciones de mantener un repositorio de información ya que actualmente está destinado para las funciones básicas de Soporte Técnico en el NOC.

*Capacidad de Procesamiento*

Se cuenta con las siguientes características de procesador:

- Intel(R) Pentium(R) Dual Core 3.40GHz

Para un servidor LDAP, se recomienda un CPU multiprocesador. Actualmente se cuenta con un procesador doble núcleo, por lo que cumpliría parcialmente con la recomendación establecida. Consecuentemente, es necesario realizar un sondeo general sobre el uso del CPU. Para ello se establece monitoreo el lunes 19 de noviembre del 2012, desde las 11h00 (considerada como una hora desde la cual se tiene máximo uso del servidor de monitoreo) para la toma de datos.
Se utiliza el comando head -n 1 /proc/stat [60], el cual brinda características del kernel y el sistema. En la Figura 4.2, se indica la doble ejecución de dicho comando, la primera vez que se ejecuta el comando se muestran los resultados del uso del CPU para T=0 (T0) segundos, y la segunda vez que se ejecuta es para T=1 (T1) segundos.

Los cuatro primeros campos resultantes subrayados en línea roja en cada ejecución son:

```
[root@monitoreo ~]# head -n 1 /proc/stat
cpu  3300768 15797 219631 14982820 104318 5864 6448 0
[root@monitoreo ~]# head -n 1 /proc/stat
cpu  3301004 15797 219646 14982993 104359 5864 6451 0
```

Figura 4. 2: Uso del CPU actual. Monitoreo Server

- 1er. Número de jiffies[69] usados por el CPU en modo usuario
- 2do.Número de jiffies en modo usuario de baja prioridad
- 3ro. Número de jiffies usados por el sistema.
- 4to. Jiffies de inactividad.

El cálculo se lo realiza utilizando la Ecuación 4.1 [61] [62]:

$$\text{ocupación CPU} = 100 * \frac{(\text{carga cpu T1} - \text{carga cpu T0})}{(\text{uso total cpu T1} - \text{uso total cpu T0})} \qquad \text{Ec. 4.1}$$

Para tomar en consideración el tiempo, se ha tomado muestras cada 30 minutos desde las 11h00 hasta las 19h00, los valores están presentados en el **Anexo 6**.

De la ecuación 4.1, se describe:

- *carga cpu*: es la suma de los tres primeros campos
- *uso total cpu*: es la variable *carga cpu* sumado el cuarto campo(tiempo de inactividad)

Por lo tanto, usando la ecuación 4.1 y los datos de la Figura 4.2, se determina:

$$\text{ocupación CPU} = 100 * \frac{(3536447 - 3536196)}{(18519440 - 18519016)} = 59.19$$

[69] Jiffie: en Linux, es un término que permite medir la frecuencia del reloj interno del sistema

Con ello se establece que el uso del CPU en una sola toma de datos es aproximadamente del 59.19 %. Por lo tanto, con los datos recogidos del **Anexo 6** se tiene un máximo de carga de 54.81 % y un promedio de 24.53 % del uso de CPU. Por otra parte, al trabajar con el servidor no se verifica procesamiento excesivo, ni problemas en sus funcionalidades, así que aun se tiene por lo menos el 40 % de la capacidad de procesamiento disponible.

#### 4.2.1.1 Capacidad de memoria [63] [64]

Para la estimación de memoria RAM se consulta a foros del producto openLDAP, en donde se resalta lo siguiente:

- La cantidad de memoria necesaria dependerá del número de entradas que se requiera almacenar y el número de atributos que use cada entrada.

- También depende de la capacidad y método de búsqueda en el repositorio, así que se debería considerar el número de índices y el tamaño cache de la base de datos.

Con el fin de conocer muestras reales se realiza una implementación básica de prueba de un servidor LDAP mostrado en el **Anexo 7** con los siguientes datos:

Datos:

Tamaño del archivo de indexado (/var/lib/ldap): 12 MB
Numero de índices: 6
Tamaño por entrada: 1.06 KB
Número de entradas inicial de Pruebas: 12.

Además, de los foros openLDAP [64] se destacan y se aplican los siguientes los siguientes cálculos referenciales (aplicados a nuestro entorno):

TamañoCacheDB= Tamaño_archivo_index * No. de Índices Ec. 4.2.

TamañoCacheDB=12 MB * 6

TamañoCacheDB=72 MB

Para 12 entradas:

TamañoCache= Tamaño_por_entrada * No._de_entradas Ec. 4.3.

TamañoCache= 1.06 KB * 12

TamañoCache= 12.76 KB

Para 3000 entradas:

TamañoCache= Tamaño_por_entrada * No._de_entradas Ec. 4.4

TamañoCache= 1.06 KB * 3000

TamañoCache= 3.1 MB

Usando los resultados de las Ecuaciones 4.2 y 4.4, se calcula la memoria RAM requerida para el caso en que se tuvieran 3000 entradas en el repositorio:

TamañoTotalRAM= TamañoCacheDB + TamañoCache Ec. 4.5.

TamañoCacheRAM= 72 MB + 3.1 MB

TamañoCacheRAM= 75.2 MB ≈ 75.2 MB

Por lo tanto, se necesita 75.2 MB (valor estándar: 128 MB o 512 MB) en memoria RAM adicional a lo actualmente tiene el servidor de monitoreo. Con el comando *free -m* se verifica la cantidad de memoria que actualmente posee el servidor:

```
[root@monitoreo ~]# free -m
             total       used       free     shared    buffers     cached
Mem:           991        953         38          0         20        128
-/+ buffers/cache:        805        186
Swap:         1983        150       1833
[root@monitoreo ~]#
```

Figura 4. 3: Uso de memoria actual. Monitoreo Server

De la Figura 4.3, se puede denotar que se tiene actualmente 1 GB de memoria y se encuentra usado el 96 %. Por lo tanto, para realizar un repositorio de datos en el

esquema base que se plantea, es necesario añadir por lo menos 128 MB de memoria RAM. Sin embargo para mejorar las capacidades de memoria, se reemplazará la memoria actual por 2 GB.

#### 4.2.1.2 Capacidad de Almacenamiento

Está relacionado con la cantidad de espacio que requiere el sistema operativo anfitrión, los programas y aplicaciones que se requieren en el servidor. Se recomienda para el servidor LDAP usar configuraciones de discos RAID 1 [70] o RAID 0+1, ya que si un disco falla mecánicamente, los datos del conjunto siguen siendo accesibles por los usuarios. También dependerá de la cantidad de información, operaciones concurrentes de lectura y disponibilidad de la base de datos. Con el comando *df –h*, se indica el uso actual de un disco.

```
[root@monitoreo ~]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup00-LogVol00
                      107G  9.4G   92G  10% /
/dev/sda1              99M   27M   68M  28% /boot
tmpfs                 496M     0  496M   0% /dev/shm
```

Figura 4. 4: Uso del disco. Monitoreo Server

Se puede verificar en la Figura 4.4 que se está usando el 10% del volumen lógico más el 28 % del sistema de archivos local, teniendo un total del 38 % del disco.
El tamaño de almacenamiento de cada entrada al disco duro es 203 bytes y el número de clientes actualmente no excede de 100. Se tienen al menos 25 entradas nuevas mensuales y se estima que el almacenamiento durará por lo menos 5 años.

No. de Entradas Totales= (No. de entradas al mes * 12) * 5
No. de Entradas Totales= (25 * 12) * 5= 1500 entradas

Capacidad en Disco= No. de Entradas Totales*Tamaño de Entrada
Capacidad en Disco= 1500 (1.05 KB) ≈ 1.6 GB

Se tiene disponible un total de 93 GB, por lo que no es necesario añadir mayor cantidad de almacenamiento en la PC de monitoreo.

[70] RAID 1: resulta útil cuando el rendimiento en lectura es más importante que la capacidad.

## 4.3 INSTALACIÓN Y CONFIGURACIÓN DEL REPOSITORIO DE INFORMACIÓN

### 4.3.1 INSTALACIÓN Y CONFIGURACIÓN DEL PAQUETE OPENLDAP

Se procede primero con la instalación del paquete openldap-2.3.43-25.el5_8.1 incluido el paquete authconfig-5.3.21-7-el a través de la herramienta yum con el siguiente comando: *yum –y install openldap-clients openldap-servers authconfig.* Luego se alista el arranque del servicio *ldap* en los modos multiusuario con soporte de red y multiusuario con entorno gráfico, garantizando su ejecución cuando el servidor se reinicie. Para ello se usa el comando: */sbin/chkconfig --level 35 ldap on.*

#### 4.3.1.1 Configuración del archivo slapd.conf

El archivo slapd.conf es la fuente central de configuración de un servidor openLDAP independiente. El contenido del archivo slapd.conf empieza con la sección de esquemas:

```
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/nis.schema
include         /etc/openldap/schema/misc.schema
include         /etc/openldap/schema/telycim.schema
include         /etc/openldap/schema/telydatapcim.schema
include         /etc/openldap/schema/telydatapcels.schema
include         /etc/openldap/schema/parameters.schema
```

Figura 4. 5: Inclusión de Esquemas

La Figura 4.5 incluye nueve esquemas. Los primeros cinco son los que vienen definidos por defecto en el paquete openLDAP instalado en la distribución CentOS.

```
[root@monitoreo ~]# cat /etc/openldap/schema/parameters.schema

attributetype ( 1.3.6.1.4.1.40922.4.2.2
        NAME 'hostname'
        DESC 'FQDN of the server'
        EQUALITY caseIgnoreIA5Match
        SUBSTR caseIgnoreIA5SubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.26{256} )


attributetype ( 1.3.6.1.4.1.40922.4.2.3
        NAME 'architecture'
        DESC 'hardware architecture of server'
        EQUALITY caseIgnoreIA5Match
        SUBSTR caseIgnoreIA5SubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.26{256} SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.40922.4.2.4
        NAME 'bandwidth'
        DESC 'type of network connection for server'
        EQUALITY caseIgnoreMatch
        SUBSTR caseIgnoreSubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128} )

attributetype ( 1.3.6.1.4.1.40922.4.2.5
        NAME 'ccq'
        DESC 'CCQ parameter configured'
        EQUALITY caseIgnoreMatch
        SUBSTR caseIgnoreSubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{64} )

attributetype ( 1.3.6.1.4.1.40922.4.2.6
        NAME 'signalstrenght'
        DESC 'Signal Strenght parameter configured'
        EQUALITY caseIgnoreIA5Match
        SUBSTR caseIgnoreIA5SubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.26{256} SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.40922.4.2.7
        NAME 'frecuency'
        DESC 'frecuency parameter configurated'
        EQUALITY caseIgnoreMatch
        SUBSTR caseIgnoreSubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{256} SINGLE-VALUE )

attributetype ( 1.3.6.1.4.1.40922.4.2.8
        NAME 'ssid'
        DESC 'ssid name  configured'
        EQUALITY caseIgnoreMatch
        SUBSTR caseIgnoreSubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{64} )


objectclass ( 1.3.6.1.4.1.40922.4.1.1
        NAME 'AirmaxParameters'
        DESC 'Internet-connected associated with Airmax Parameters'
        SUP top AUXILIARY
        MUST ( hostname )
        MAY ( architecture $ bandwidth $ ccq $ signalstrenght $ frecuency $ ssid
        ))
```

Figura 4. 6: Archivo parameters.schema

Los siguientes cuatro esquemas son definidos como sigue:

- Telycim.schema: es el nombre del archivo que contiene la definición de DMTF CIM 2.5  LDAP Schema.

- telydatapcim.schema: es el nombre del archivo que contiene el esquema definido mediante el RFC 3703.

- telydatapcels.schema: Es el nombre del archivo que contiene el esquema definido mediante el RFC 4102.

Como se mencionó en el diseño Capítulo 3, IANA estableció el número PEN 40922 para Telydata Cía. Ltda. A partir de este número se elige:

- el sufijo 40922.4. para la gestión en el área WiFi.
- el sufijo 40922.4.1. para asignación de clases de objetos.
- el sufijo 40922.4.2. para la asignación de tipos de atributos.

Continuando con la configuración del archivo slapd.conf, en la Figura 4.7 se indica lo siguiente:

- El backend Berkeley DB 4 como base de datos a utilizar, ya que mejora las características de indexado y caché para acelerar el rendimiento manteniendo copias de datos locales.
- El sufijo principal para el contexto de nombrado
- El nombre distinguido raíz que corresponde a la entrada principal del árbol.

```
database        bdb
suffix          "dc=telydata,dc=net"
rootdn          "cn=Manager,dc=telydata,dc=net"
```

Figura 4. 7: Base de datos y Contexto de nombrado

Continuando con la configuración del archivo slapd.conf, en la Figura 4.8 se presenta las líneas empleadas para la definición de la contraseña usada por el rootDN. La contraseña está cifrada usando hash MD5 (SSHA)[71].

```
rootpw          secret
rootpw          {SSHA}j2rdtZKa7KaiPlId0xakJK/QTx0oRmcV
```

Figura 4. 8: Password de rootDN

[71] MD5/SSHA (Salted Secure Hash Algorithm): método de hashing (generación de una serie única de datos) para la codificación del password de usuarios. Recomendado para la generación y almacenaje de contraseñas en un servidor LDAP.

En la Figura 4.9 se indica la definición de los parámetros de ubicación física correspondientes a la base de datos y los permisos correspondientes. Los permisos son de lectura, escritura, y ejecución para el propietario (ldap); lectura y escritura para el grupo, sin permisos para otros. Dicha configuración se encuentra dentro del archivo slapd.conf.

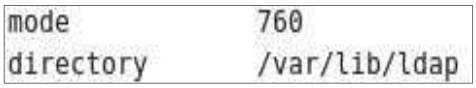



Figura 4. 9: Permisos de archivos de base de datos

La Figura 4.10 muestra la última parte del archivo slapd.conf, en donde se define el conjunto de índices para mantener las búsquedas de manera óptima. En esta definición se incluye un índice para el atributo *bussinesCategory* con regla de coincidencia exacta *eq (equality)*, el cual permitirá filtrar los datos cuyo valor sea igual a la cadena *Corporativo*. Esto es requerido ya que se necesita indexar a los usuarios corporativos quienes son a los que se les aplicará el controlador de la capacidad. El índice *pres (presence)* acelera la búsqueda de entradas que contienen un atributo específico. El índice *sub (substring)* registra la información necesaria para llevar a cabo coincidencias simples de subcadenas sobre los valores de los atributos.

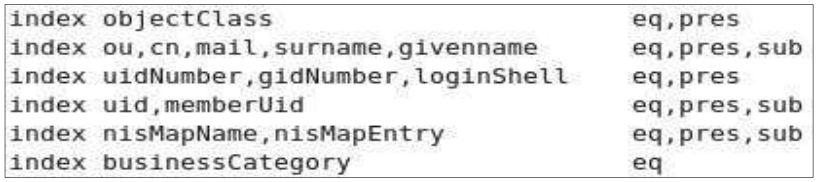



Figura 4. 10: Conjunto de índices para búsquedas

#### 4.3.1.2 Configuración del archivo ldap.conf

En la Figura 4.11 se definen tres parámetros del archivo ldap.conf que se describen a continuación:

- URI (Identificador Uniforme de Recurso): que identifica a LDAP como el protocolo que se usará a nivel de capa aplicación, la dirección IP del servidor de monitoreo y el puerto respectivo.

- La entrada base del árbol DIT (*Directory Information Tree*).
- Ubicación del directorio que contiene los certificados necesarios para realizar conexiones seguras y confidencialidad en capa transporte.

```
URI ldap://201.219.6.247:389/
BASE dc=telydata,dc=net
TLS_CACERTDIR /etc/openldap/cacerts
```

Figura 4. 11: Configuración del archivo ldap.conf

#### 4.3.1.3 Creación del DN de partida

El DN de partida es *cn=Manager,dc=telydata,dc=net*, mediante el cual se implementarán las entradas subsiguientes del Arbol DIT. Para la creación del DN de partida se ha creado el archivo Manager.ldif el cual contiene la clase inetOrgPerson con los atributos cn (*common name*) y sn (*surname*) tal como se muestra en la Figura 4.12.

```
[root@monitoreo openldap]# cat Manager.ldif
dn: cn=Manager,dc=telydata,dc=net
cn: Manager
sn: Manager
objectClass: inetOrgPerson
[root@monitoreo openldap]#
```

Figura 4. 12: Contenido del archivo Manager.ldif

Para poder compilar el servidor openLDAP se debe tomar en cuenta que el propietario de los archivos de base de datos de OpenLDAP debe ser *ldap.* En la Figura 4.13 se indica que el archivo Manager.ldif es añadido a la entrada raíz del directorio (*dc=telydata,dc=net*) para formar el dominio completo. También se indica el inicio del servicio ldap.

```
[root@monitoreo openldap]#
[root@monitoreo openldap]# /usr/sbin/slapadd -v -l /etc/openldap/Manager.ldif
added: "cn=Manager,dc=telydata,dc=net" (00000570)
[root@monitoreo openldap]# service ldap start
Starting slapd:                                            [  OK  ]
[root@monitoreo openldap]#
```

Figura 4. 13: Inicio del Servidor LDAP

### 4.3.2 INSTALACIÓN DEL ADMINISTRADOR GRÁFICO DEL MÓDULO GESTOR

Se hace uso del proyecto Apache Directory Studio elegido en el capítulo 3, sección 3.6, el cual ofrece una interfaz gráfica que permitirá administrar el acceso a los datos del servidor LDAP. Para su instalación se requiere de tres aplicaciones importantes: Eclipse SDK versión: 4.2.1. (Juno), Java JDK 7u9 para RHEL y Java JRE 7u9 para RHEL.

#### 4.3.2.1 Instalación y Configuración de Java

Para la instalación, los paquetes jdk 7u9 y jre 7u9 son obtenidos de la página web de la compañía Oracle [64]. Para su instalación se usan los comandos:

- *rpm –Uvh /opt/Backup/jdk-7ue-linux-x64.rpm*
- *rpm –Uvh /opt/Backup/jre-7ue-linux-x64.rpm*

A continuación se instalan las librerías y alternativas para el uso de Java con cada componente, tanto para java, javaws, y javac. Para eso se utilizan los siguientes comandos:

- alternatives –install /usr/bin/java java /usr/java/jdk.1.7.0_09/jre/bin/java 2000
- alternatives –install /usr/bin/javaws javaws /usr/java/latest/jre/bin/javaws 2000
- alternatives –install /usr/bin/javac javac /usr/java/latest/bin/javac 2000
- alternatives –install /usr/bin/jar jar /usr/java/jdk.1.7.0_09/bin/jar 2000

En la Figura 4.14 se verifica la instalación de java en Servidor de Monitoreo:

```
[root@monitoreo ~]# java -version
java version "1.7.0_09"
Java(TM) SE Runtime Environment (build 1.7.0_09-b05)
Java HotSpot(TM) 64-Bit Server VM (build 23.5-b02, mixed mode)
[root@monitoreo ~]# javac -version
javac 1.7.0_09
[root@monitoreo ~]#
```

Figura 4. 14: Comprobación de la instalación Java 1.7

Se adecua las variables de ambiente de Java, agregando las líneas de código *export JAVA_HOME="/usr/java/jdk1.7.0_09"* y *export JAVA_HOME="/usr/java/jre1.7.0_09"* en el archivo /etc/profile para que no se pierdan la reiniciar el servidor.
Con ello se finaliza la instalación de Java y se procede con la instalación de eclipse.

#### 4.3.2.2 Configuración de Eclipse Juno

Se descarga el paquete Eclipse Juno del sitio web *http://www.eclipse.org/downloads/* y se lo descomprime en el directorio /opt descomprime:

- tar –xvfz eclipse –SDK-4.2.1-linux-gtk-x86_64.tar.gz -C /opt

Se añade permisos al directorio /opt/eclipse para permitir la ejecución del programa Eclipse Juno:

- chmod –R +r /opt/eclipse

Se crea el archivo ejecutable /usr/bin/eclipse y se añaden dentro de este las variables de entorno del programa eclipse para que le programa se pueda abrir vía terminal de Linux. El contenido del archivo /usr/bin/eclipse se muestran en la Figura 4.15.

```
export ECLIPSE_HOME="/opt/eclipse"

$ECLIPSE_HOME/eclipse $*
```

Figura 4. 15: Configuración del archivo ejecutable de Eclipse

Finalmente se crean lanzadores para facilitar la ejecución de eclipse desde el escritorio de CentOS. Se asigna el archivo /usr/share/applications/eclipse.desktop con el código presentado mostrado en la Figura 4.16.


Figura 4. 16: Configuración de la entrada GUI de Eclipse

Con ello, queda lista la instalación de Eclipse, útil para la implementación del módulo Gestor.

#### 4.3.2.3 Instalación de ApacheDS LDAP Browser

Inicializado Eclipse se procede a instalar el plug-in[72] de Apache Directory Studio Browser. Desde el menú principal, se añade nuevo software, especificando la URL: http//directory.apache.org/studio/update/1.x. Luego se elige Apache Directory Studio LDAP Browser. Existen varios componentes como es el caso de ApacheDS LDIF Editor, diseñado para editar archivos con extensión LDIF con la sintaxis similar a la del código fuente original. Edita valores para diferentes tipos de atributos. Existe también el ApacheDS Schema Editor que permite la edición de archivos tipo schema en el formato OpenLDAP, también permite la edición fácil de tipos de atributos y clases de objetos para Apache Directory Server y OpenLDAP. Ninguno de estos dos últimos componentes adicionales es requerido por el momento, debido a que la creación de archivos LDIF y Schema se los realiza directamente con el editor *vim*. A continuación se resume el procedimiento explicado, mediante la Figura 4.17.

[72] Plug-in: Programa que puede anexarse a otro para aumentar sus funcionalidades

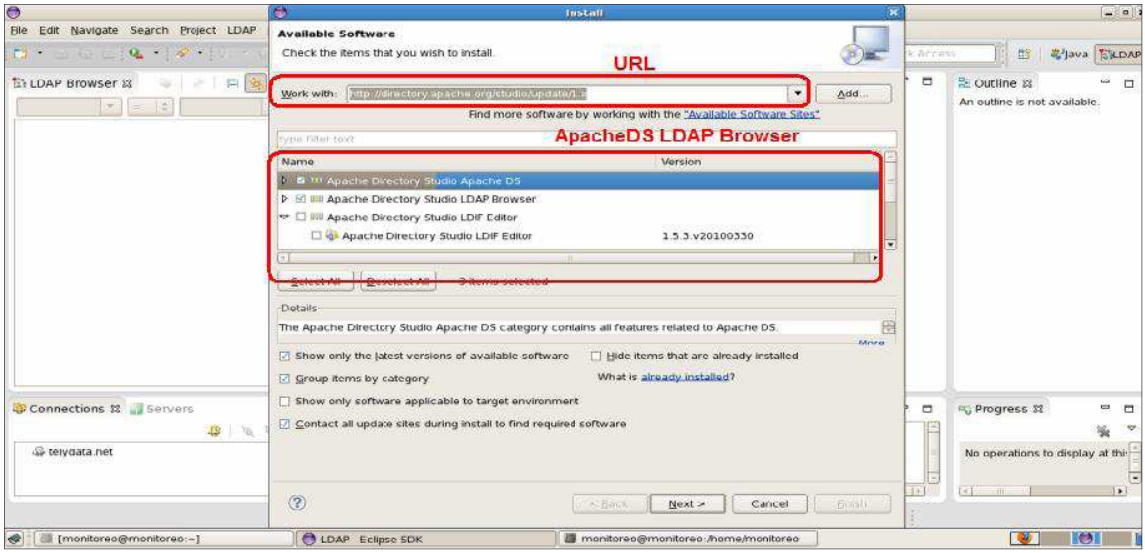



Figura 4. 17: Paquete ApacheDS LDAP Browser

En la Figura 4.18 se indica la aceptación de los términos de Apache License versión 2.0.

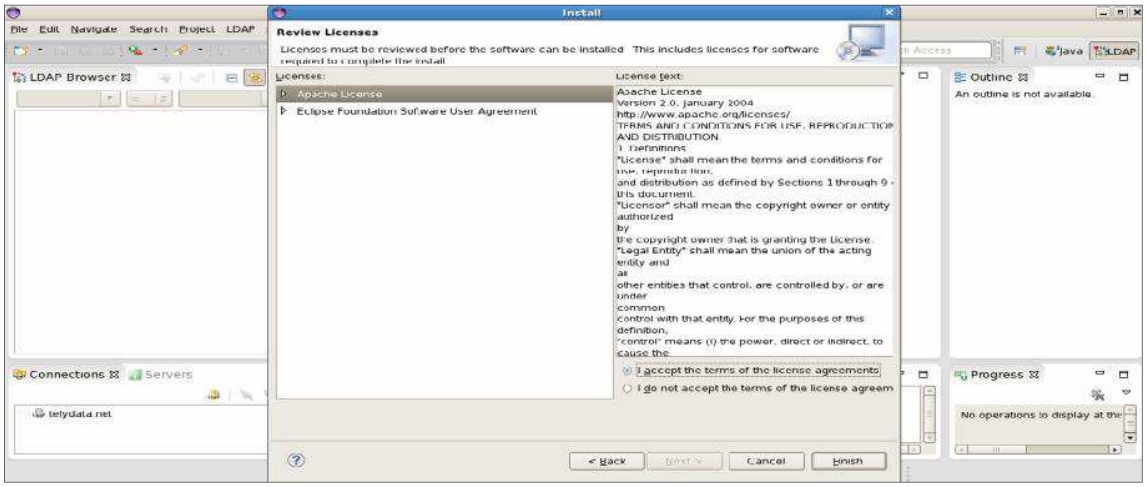



Figura 4. 18: Términos de Licenciamiento Apache Directory Studio

Posterior a ello, Eclipse realiza la actualización e instalación de los paquetes requeridos y queda ApacheDS LDAP Browser completamente instalado y listo para enlazarlo con el servidor LDAP.

#### 4.3.2.4 Creación de la conexión con el Servidor LDAP

Para crear una conexión LDAP se especifica un nuevo proyecto LDAP Browser y se selecciona en *LDAP conection,* como se indica en la Figura 4.19.

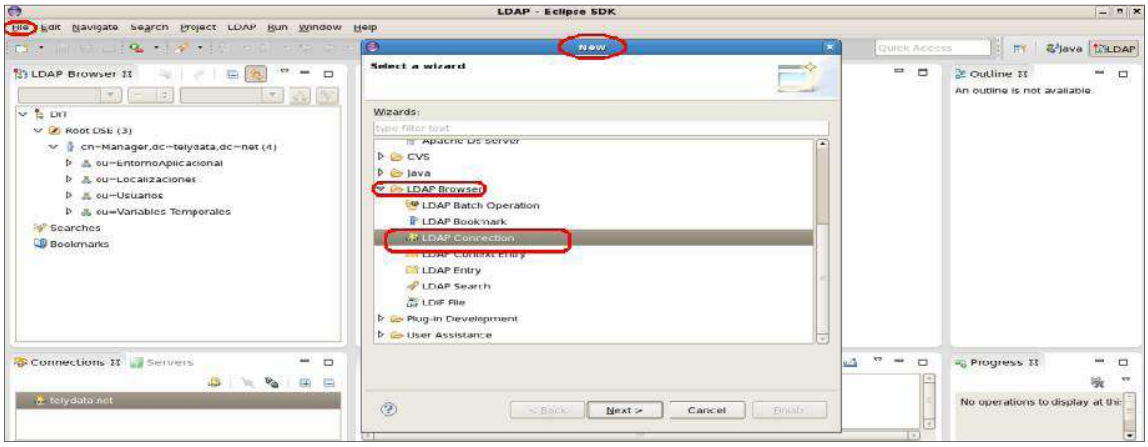



Figura 4. 19: Conexión con el servidor LDAP

A continuación se añaden los datos del servidor LDAP que se está ejecutando en el host de Monitoreo. El nombre de la conexión es telydata.net, el hostname es 201.219.6.247, puerto 389. No se ha asignado un método de encripción para la conexión. Posteriormente se establece la autenticación básica para la conexión con el usuario del servidor LDAP a través de los siguientes datos:

- BindDN or User: cn=Manager,dc=telydata,dc=net
- Password: **********

Para comprobar la autenticación básica con el servidor presionamos “check authentication”, y debería a continuación salir el mensaje: *“The authentication was successful”*, como se indica en la Figura 4.20. Con ello, se verifica la correcta conexión al servidor.

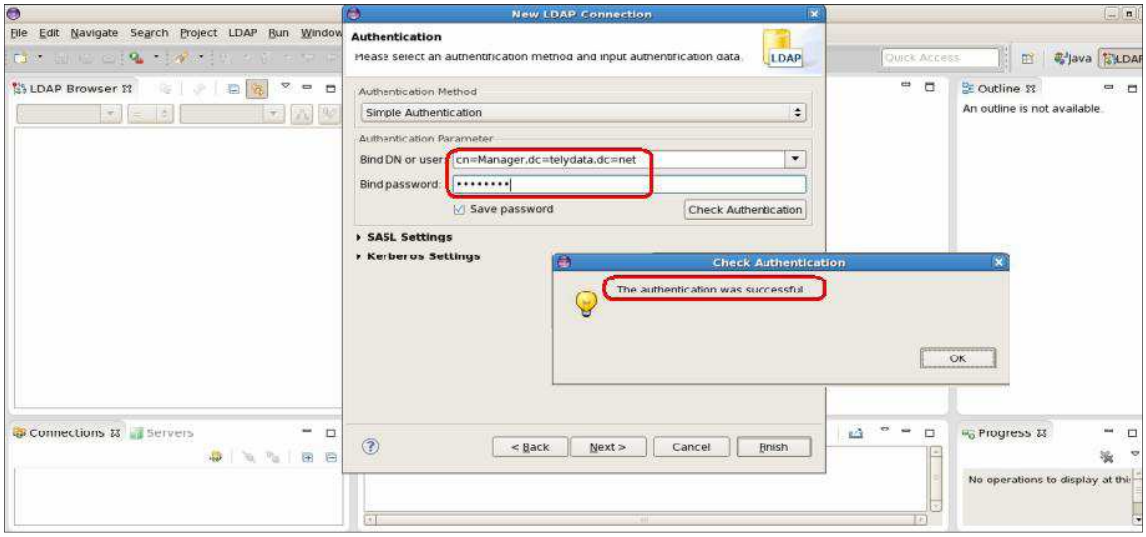



Figura 4. 20: Autenticación básica para el servidor LDAP

Luego se especifica el baseDN a obtener en el browser. Para ello, se activa la opción: Obtener Base DNs desde el RootDSE[73] y queda habilitada la entrada principal establecida por ApacheDS.

Con la opción límite de cuentas (*count Limit*) se determina el número máximo de entradas devueltas desde el servidor cuando se realiza consultas al directorio, en este caso usamos el valor por defecto, igual a 1000.

En el parámetro Tiempo Limite (*Time Limit*), se especifica el tiempo máximo, en segundos, para que el servidor realice búsquedas, el valor por defecto es 0 e indica que no existe un valor de limitación alguno.

En el parámetro omisión de referencia por alias (*Alias Deferencing*), utilizada para no permitir que una entrada LDAP contenga el DN de otra entrada LDAP relacionada, ya sea dentro del mismo servidor LDAP o en una réplica de este. Se elige las opciones *Search* y *Finding Base DN* para realizar búsquedas normales dentro del baseDN diseñado: *cn=Manager,dc=telydata,dc=net.*

[73] RootDSE(Root DSA Especific Entry): Definido en LDAP v3, utilizado para especificar un nivel base dentro del DIT.

En el área Manejo de Referencias (*Referrals*[74] *Handling*), se utiliza para el que seguimiento de referencias del DIT se lo realice manualmente. En la Figura 4.21 se expone lo anteriormente explicado.

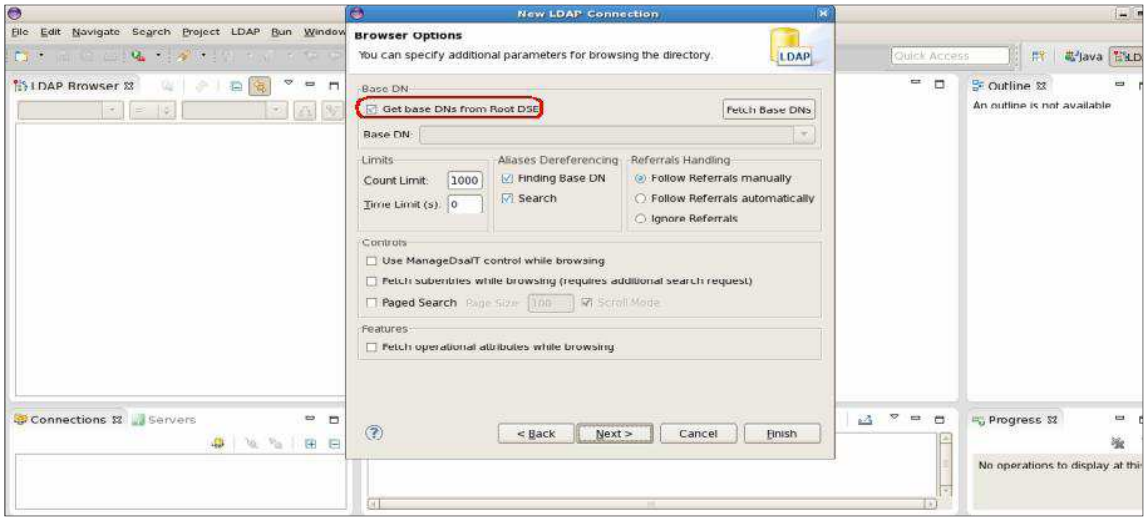



Figura 4. 21: Opciones de Browsing para ApacheDS LDAP Browser

Finalmente se tiene la entrada principal dc=telydata, dc=net ya añadida, lista para la construcción del árbol DIT especificado en el capítulo 3. Como se puede verificar en la Figura 4.22, se tienen en la parte inferior *Search Logs*, que indica que el resultado de la conexión con el servidor OpenLDAP es correcto, se indica la fecha, y el número de entradas (1 entrada) actualmente.

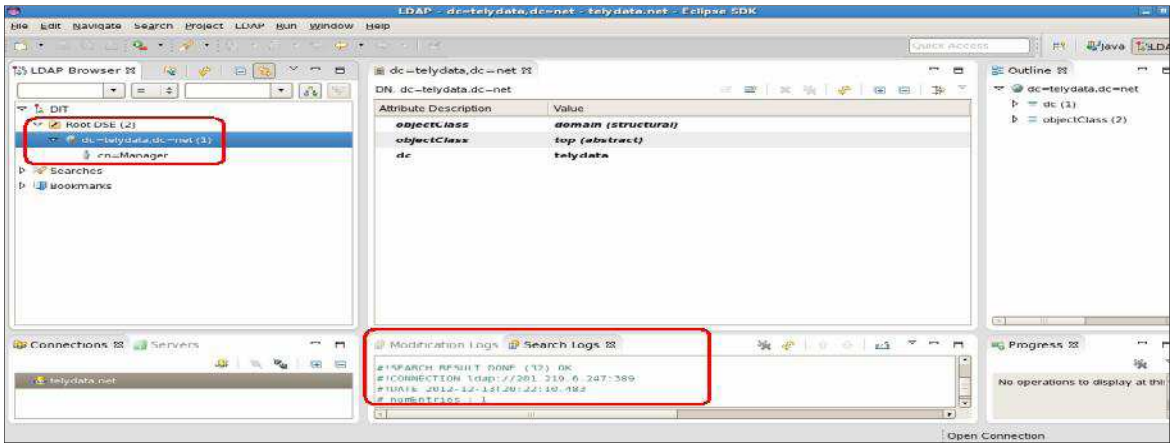



Figura 4. 22: Entrada Principal añadida a ApacheDS LDAP Browser

[74] Referral: es un conjunto de objetos que contienen la URL de una entrada LDAP.

#### 4.3.2.5 Interfaz de Administración de Entradas

En la Figura 4.23 se indica las partes principales del Administrador ApacheDS LDAP Browser, las cuales permitirán trabajar con la modificación, exploración y búsqueda de información. Las vistas de la ventana principal del administrador están ordenadas como se describe a continuación:

1. LDAP Browser View: ubicada en la parte superior izquierda de la ventana, muestra el árbol DIT formado, para realizar búsquedas comunes y marcadores de la conexión seleccionada.

2. Área de edición ubicada en la mitad de la ventana, especifica las utilidades:

   a. *Entry Editor*, muestra los atributos y sus valores de una entrada seleccionada. Es posible añadir, borrar o editar atributos.

   b. *Search Result Editor*, muestra el resultado de la búsqueda seleccionada más frecuente en una tabla.

   c. *Schema Browser*, válida para encontrar cualquier schema con sus definiciones de clases de objetos y atributos.

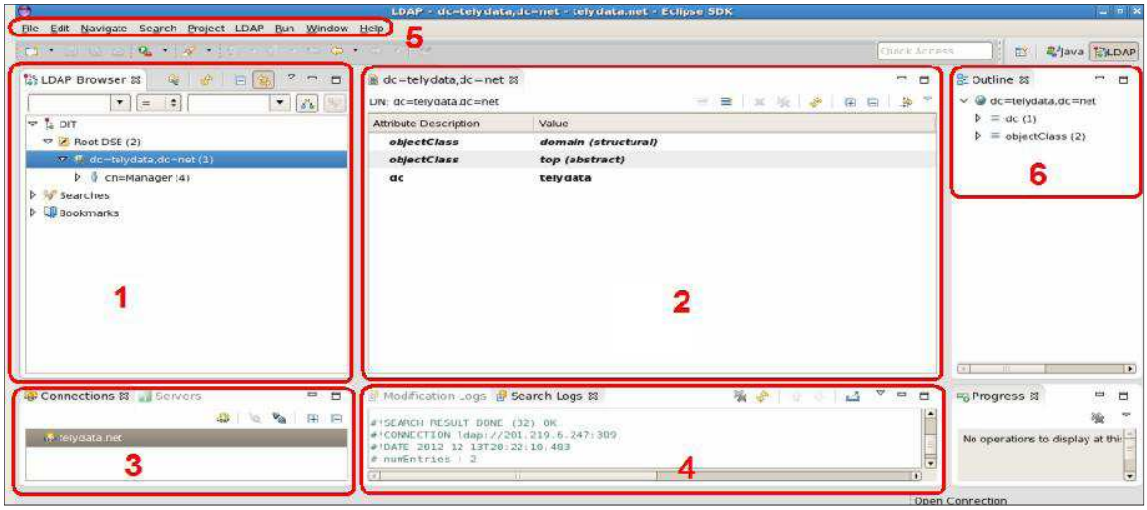



Figura 4. 23: Interfaz de Administración de Entradas

3. Connections: es usado para crear, editar, borrar, modificar, abrir o cerrar conexiones con servidores LDAP.

4. Logs: en esta área se indica, entre otros anuncios, si las modificaciones hechas fueron satisfactorias o fallidas, así como también los resultados de las peticiones de búsqueda.

5. Menu: es el menú principal, mediante el cual brinda diferentes utilidades para el trabajo con ApacheDS como guardar o abrir proyectos, abrir perspectivas LDAP, etc.

6. Outline view: muestra principalmente la estructura de clases de la entrada seleccionada en el LDAP Browser View.

### 4.3.3 CREACIÓN DE LAS ENTRADAS DE INFORMACIÓN PARA EL WISP TELYDATA CÍA. LTDA

Para describir un procedimiento general de cómo crear las entradas a través del interfaz de administración, se explica a manera de ejemplo la creación de la siguiente entrada: *ou=EntornoAplicacional,cn=Manager,dc=telydata,dc=net.*
Una vez creada la entrada principal *cn=Manager,dc=telydata,dc=net* en la sección 4.3.2.4, como paso inicial se añade una nueva entrada partiendo del nivel *cn=Manager*, tal como se indica en la Figura 4.24

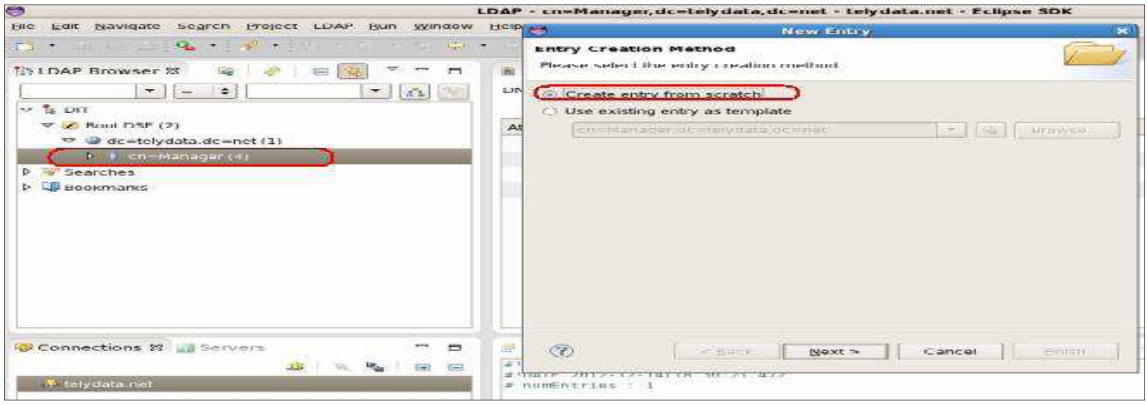



Figura 4. 24: Creación de entradas del DIT de Telydata

La opción "*create entry from scratch"* permite crear la entrada a partir de cn=Manager. Luego se seleccionan las Clases de Objetos a utilizar, en este caso se selecciona organizationalUnit que incluye a su clase superior (top), tal como se muestra en la Figura 4.25. A continuación, se ingresan los valores de todos los atributos requeridos por las clases de objetos anteriormente seleccionadas. En este punto también hay la posibilidad de usar atributos opcionales con sus respectivos valores. Para este caso se ingresa el atributo *ou* con su valor *Entorno Aplicacional,* tal como se muestra en la Figura 4.26. También el administrador gráfico indica cual es el DN padre y el DN que resultará luego de terminar con la entrada que se está realizando.

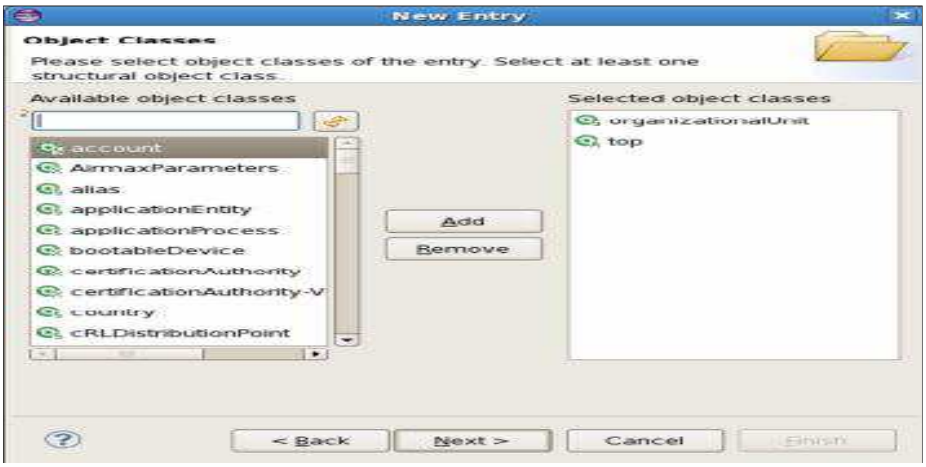



Figura 4. 25: Elección de Clases de Objetos

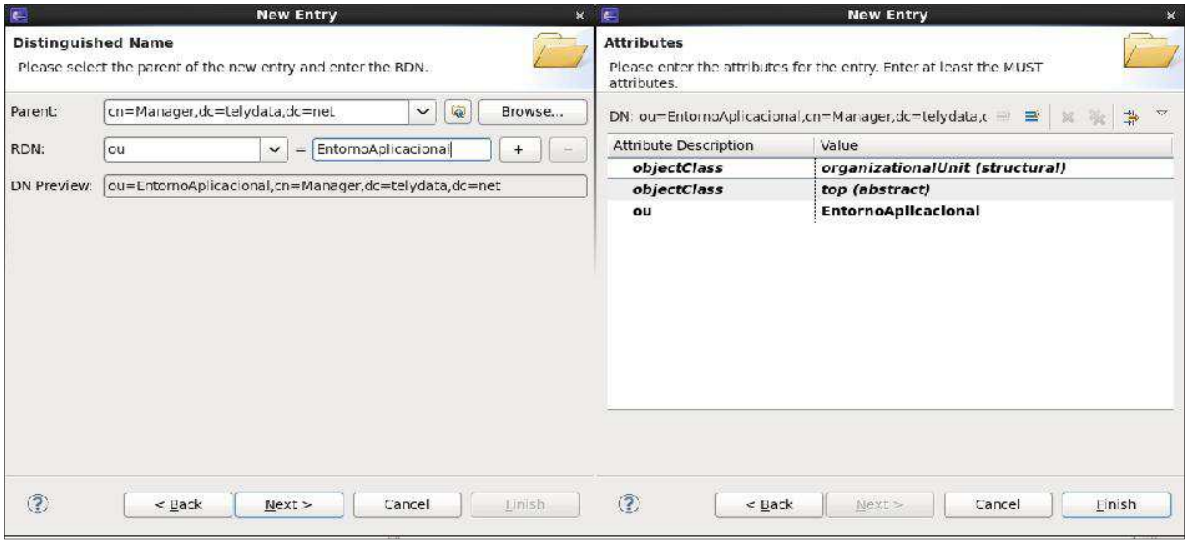



Figura 4. 26: Elección de Atributos necesarios

Finalmente, queda la entrada en el área LDAP Browser View, tal como se muestra en la Figura 4.27.

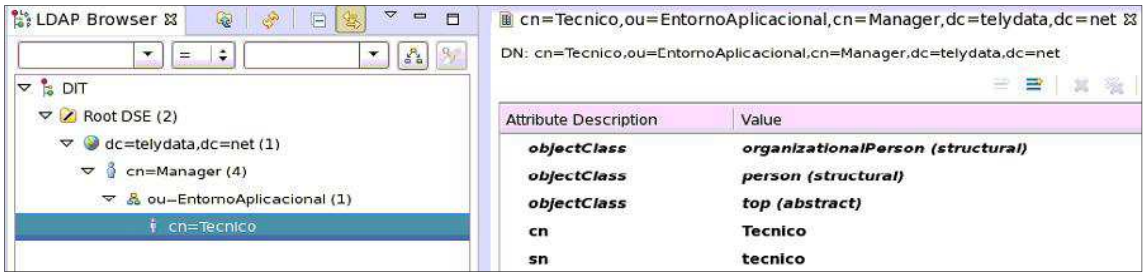



Figura 4. 27: Ejemplo de creación de entradas

### 4.3.3.1 Creación de las Entradas para la entidad Entorno Aplicacional

a) *pcelsRole=Conexiones,cn=TraficoCorporativo,cn=Tecnico,ou=Entorno Aplicacional, cn=Manager,dc=Telydata,dc=net*, se crea para establecer las conexiones con las direcciones IP, según la sección 2.6.2, tal como se muestra en la Figura 4.28.

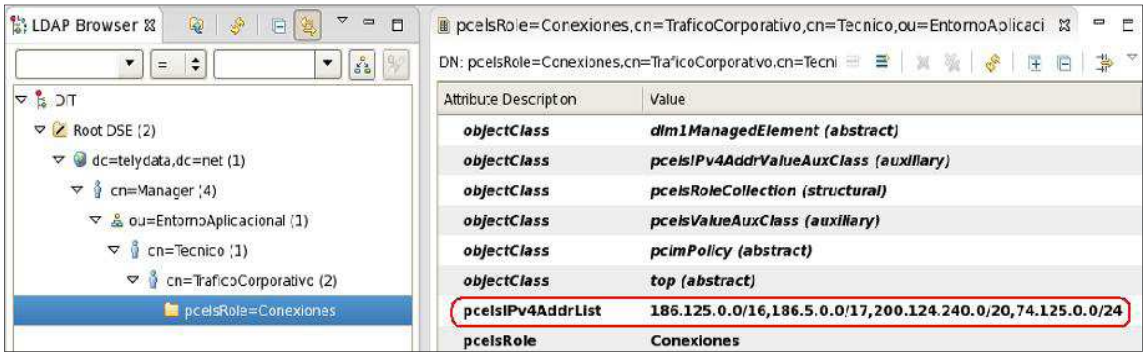



Figura 4. 28: Creación de la entrada *Conexiones*

b) *pcelsRole=Puertos,cn=TraficoCorporativo,cn=Tecnico,ou=EntornoAplicacional, cn=Manager,dc=telydata,dc=net*, se crea para la asignación de Puertos, tal como se muestra en la Figura 4.29.

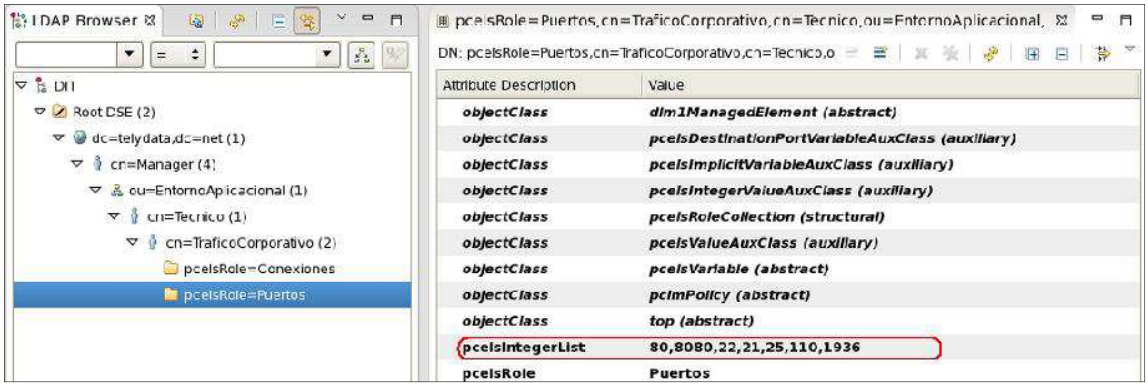



Figura 4. 29: Creación de la entrada *Puertos*

### 4.3.3.2 Creación de las entradas para la entidad Usuarios, sección Cliente

#### *4.3.3.2.1 Clientes del Sur: Ejemplo para Alexandra Parreño*

*ou=RIT_Sur,ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net*; se crea para almacenar todos los clientes del Sur, tal como se muestra en la Figura 4.30 para la cliente Alexandra Parreño.

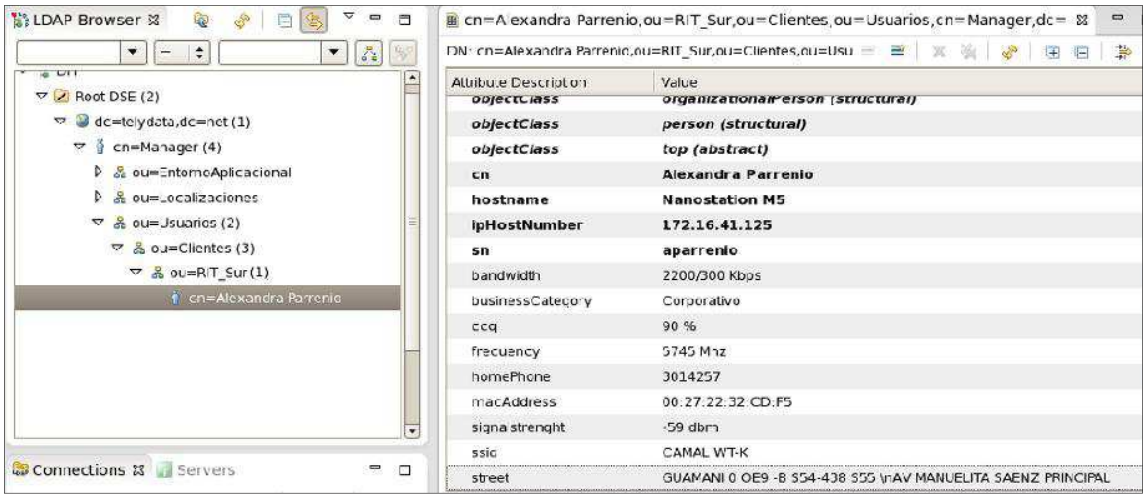



Figura 4. 30: Clientes del Sur. Ejemplo Alexandra Parreño

#### *4.3.3.2.2 Clientes del Centro: Ejemplo para David Cachahuasai*

*ou=RIT_Centro,ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net*; se crea para almacenar todos los clientes del Centro, tal como se muestra en la Figura 4.31 para el cliente David Cachahuasai.

*4.3.3.2.3 Clientes del Norte, Sección Planada: Ejemplo para José Montero*

*cn=Planada,ou=RIT_Norte,ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net* se crea para almacenar los clientes del norte pertenecientes al sector de la Planada, tal como se muestra en la Figura 4.32 para el cliente José Montero.

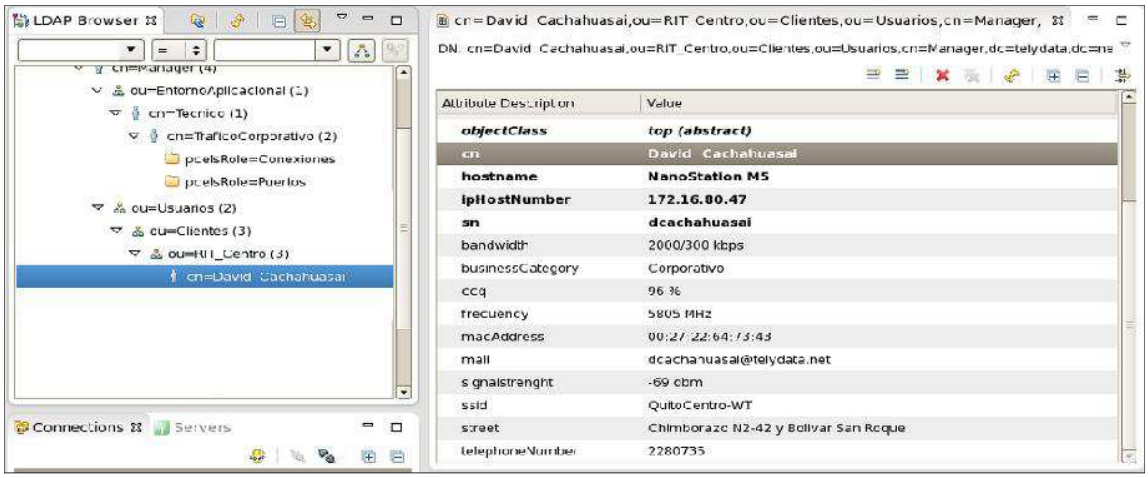



Figura 4. 31: Clientes del Centro. Ejemplo cliente David Cachahuasai

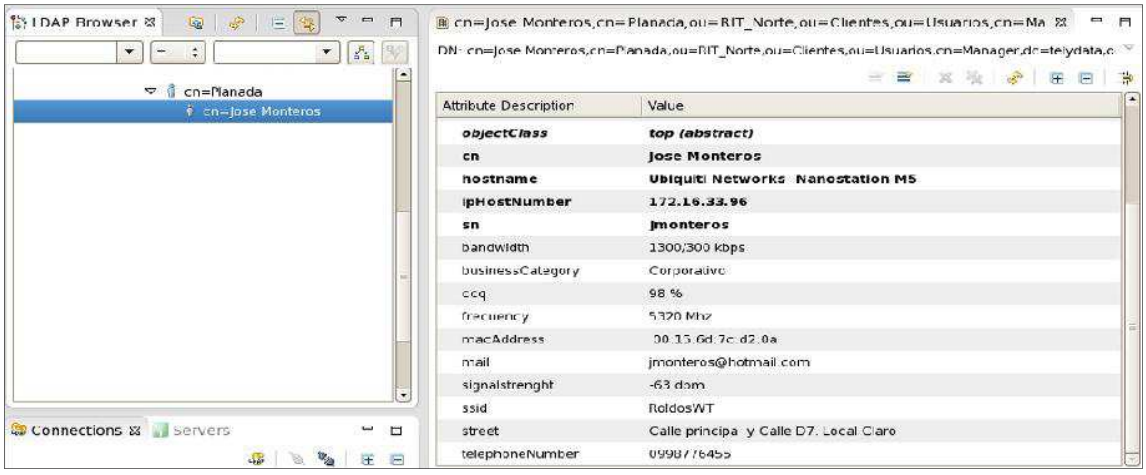



Figura 4. 32: Clientes del Norte, sección Planada. Ejemplo cliente José Montero

*4.3.3.2.4 Clientes del Norte Sección Roldós. Ejemplo para Sandra Gordillo*

*cn=Roldos,ou=RIT_Norte,ou=Clientes,ou=Usuarios,cn=Manager,dc=Telydata,dc=net*; se crea para almacenar los clientes del norte pertenecientes al sector de la Roldós, tal como se muestra en la Figura 4.33 para el cliente Sandra Gordillo.

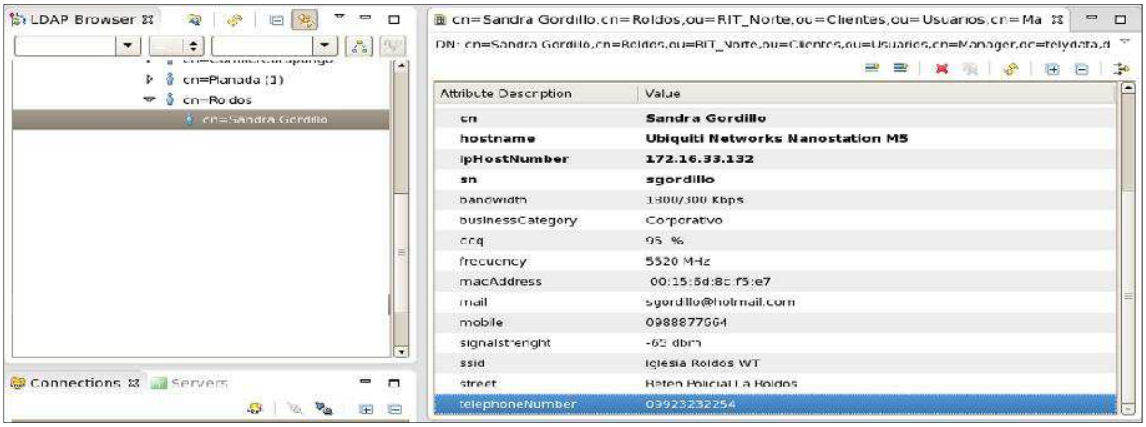



Figura 4. 33: Clientes del Norte Sección Roldós. Ejemplo para Sandra Gordillo

*4.3.3.2.5 Clientes del Norte Sección Comité/Carapungo. Ejemplo para Patricia Guacollante.*

*cn=Comite/Carapungo,ou=RIT_Norte,ou=Clientes,ou=Usuarios,cn=Manager, dc=Telydata, dc=net*; se crea para almacenar los clientes del norte pertenecientes al sector del Comité del Pueblo y Carapungo, tal como se muestra en la Figura 4.34 para la cliente Patricia Guacollante.

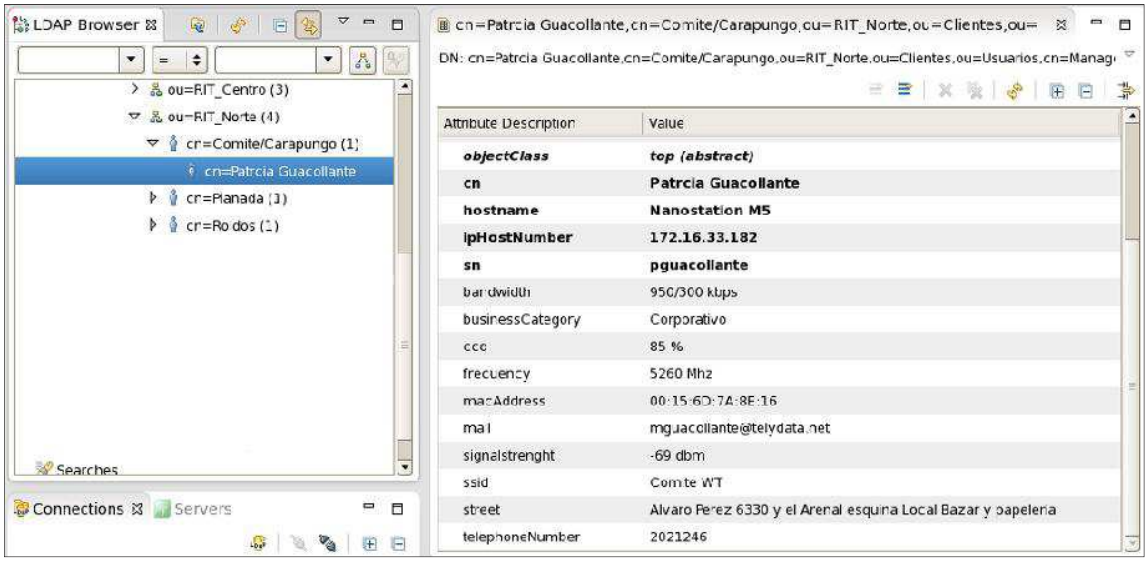



Figura 4. 34: Clientes del Norte, Sección Comité del Pueblo. Cliente ejemplo Patricia Guacollante

*4.3.3.2.6 Creación de las entradas para la entidad Usuarios, sección Técnicos*

*ou=Tecnicos,ou=Usuarios,cn=Manager,dc=Telydata,dc=net*; se crea para almacenar la información de los técnicos de Telydata, tal como se muestra en la Figura 4.35 para Alejandro Andrade.

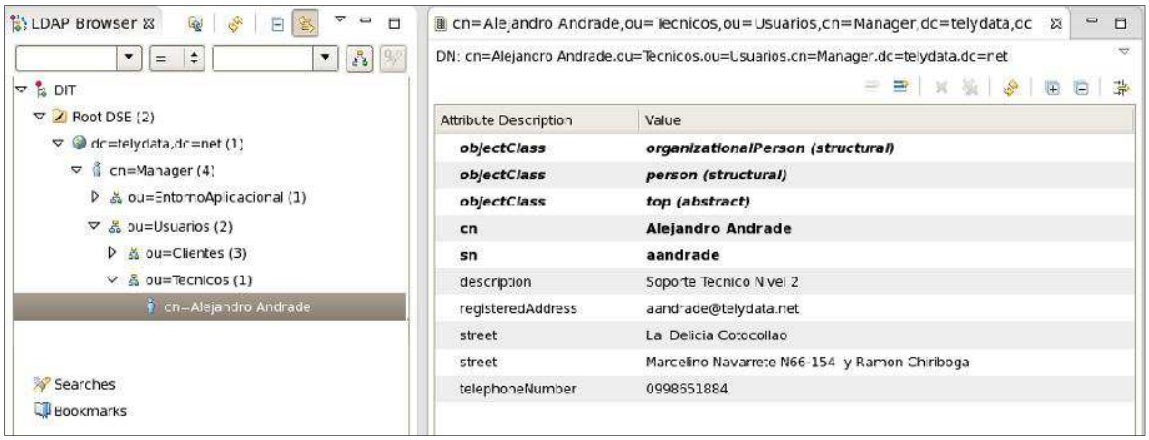



Figura 4. 35: Usuarios, sección Técnicos: Técnico ejemplo Alejandro Andrade

### 4.3.3.3 Creación de las entradas para la entidad Variables Temporales

a) *cn=VariableDiaSemana,ou=Variables Temporales,cn=Manager, dc=telydata, dc=net;* se crea para hacer referencia al día de la semana en que el equipo inalámbrico será reiniciado. En la Figura 4.36 se indica las clases y atributos respectivos.

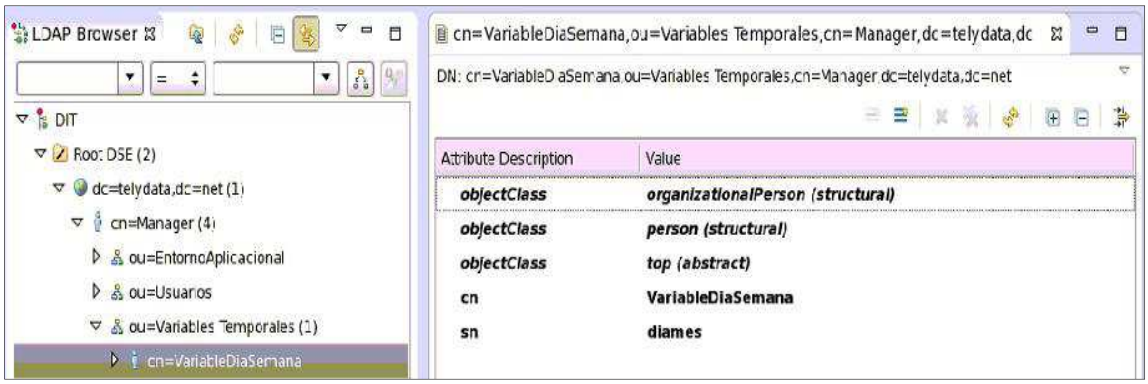



Figura 4. 36: Entidad Variables Temporales. Entrada VariableDiaSemana

b) *pcimTPCDayOfWeekMask='0001000', cn=VariableDiaSemana,ou=Variables Temporales,cn=Manager,dc=telydata,dc=net;* se crea para asignar una máscara que representará el día de la semana en que se reiniciará el equipo inalámbrico, tal como se muestra en la Figura 4.37.

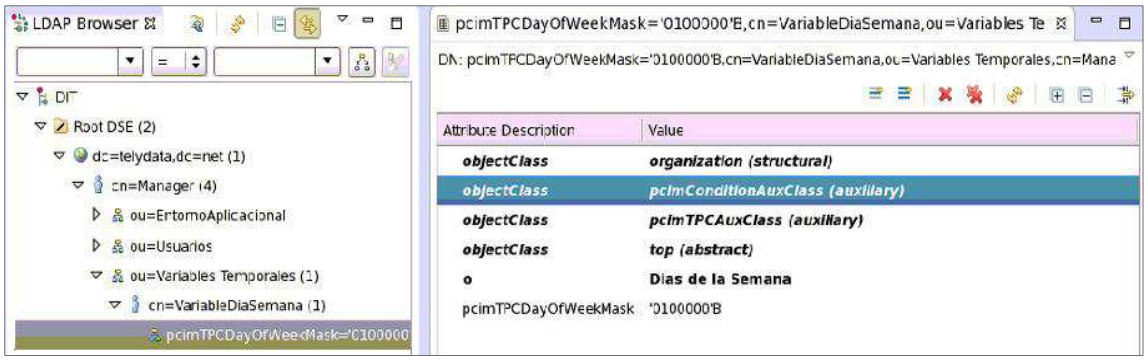



Figura 4. 37: Máscara Día de la Semana

### 4.3.3.4 Creación de las entradas para la entidad Localizaciones

a) *cn=TopologíaNorte,l=Quito_Norte,ou=Localizaciones,cn=Manager,dc=telydata, dc=net;* se crea para representar la ubicación del enlace Iñaquito Norte y almacenar su topología, tal como se muestra en la Figura 4.38.

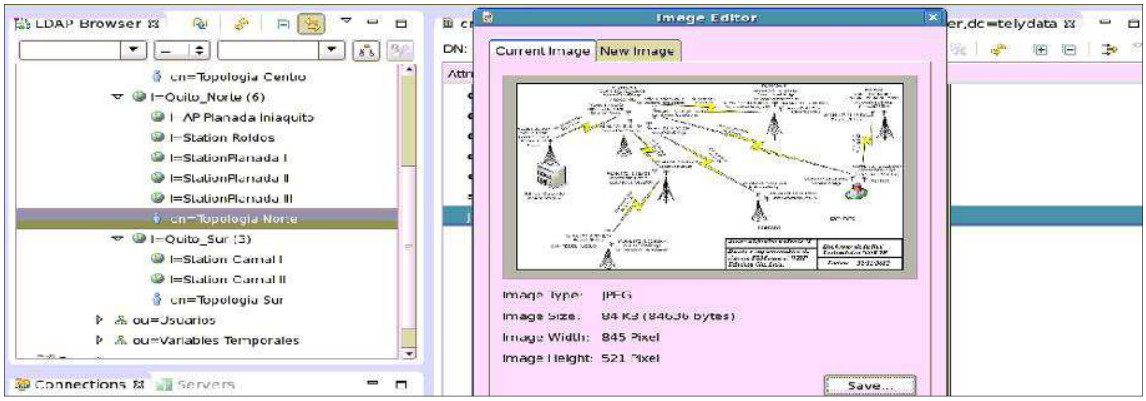



Figura 4. 38: Entidad Localizaciones, sección Topología Norte

b) *cn=Topología Sur,l=Quito_Sur,ou=Localizaciones, cn=Manager,dc=telydata, dc=net;* se crea para representar la ubicación del enlace Iñaquito Sur y almacenar su topología, tal como se muestra en la Figura 4.39.

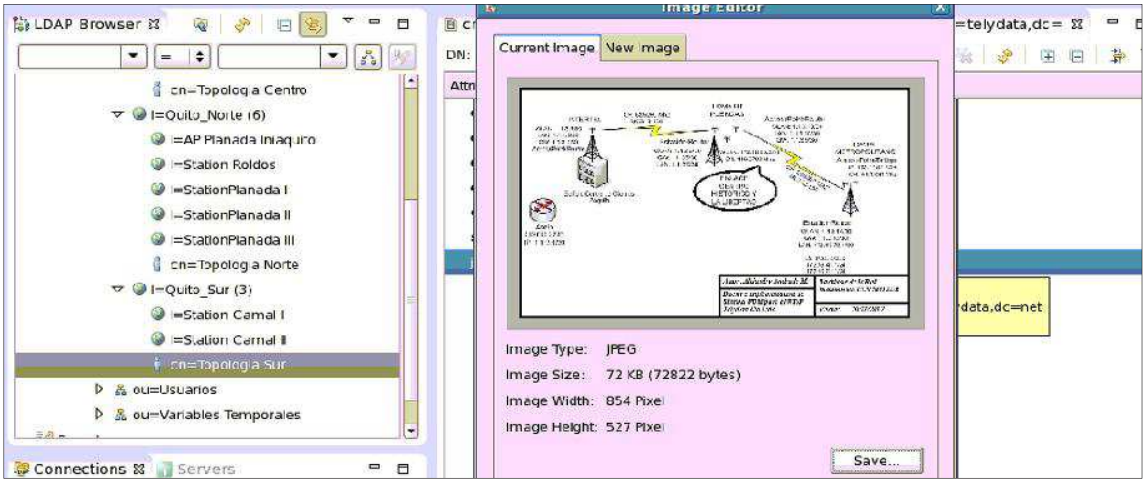



Figura 4. 39: Entidad Localizaciones, sección Topología Sur

Como se puede verificar en las Figuras 4.38 y 4.39, los gráficos de las topologías se pueden descargar tantas veces se requiera. El backbone del Sur distribuye la red del centro histórico, es por ello, que no es necesario diagramar el backbone del Centro. Una aclaración importante es que si el backbone cambia, los gráficos deberían ser también actualizados inmediatamente y almacenados en el servidor LDAP.

En la Figura 4.40 y 4.41 se indica la implementación de la topología de la Red Inalámbrica de Telydata Telecomunicaciones y Datos, tanto para el norte como para el sur de Quito.

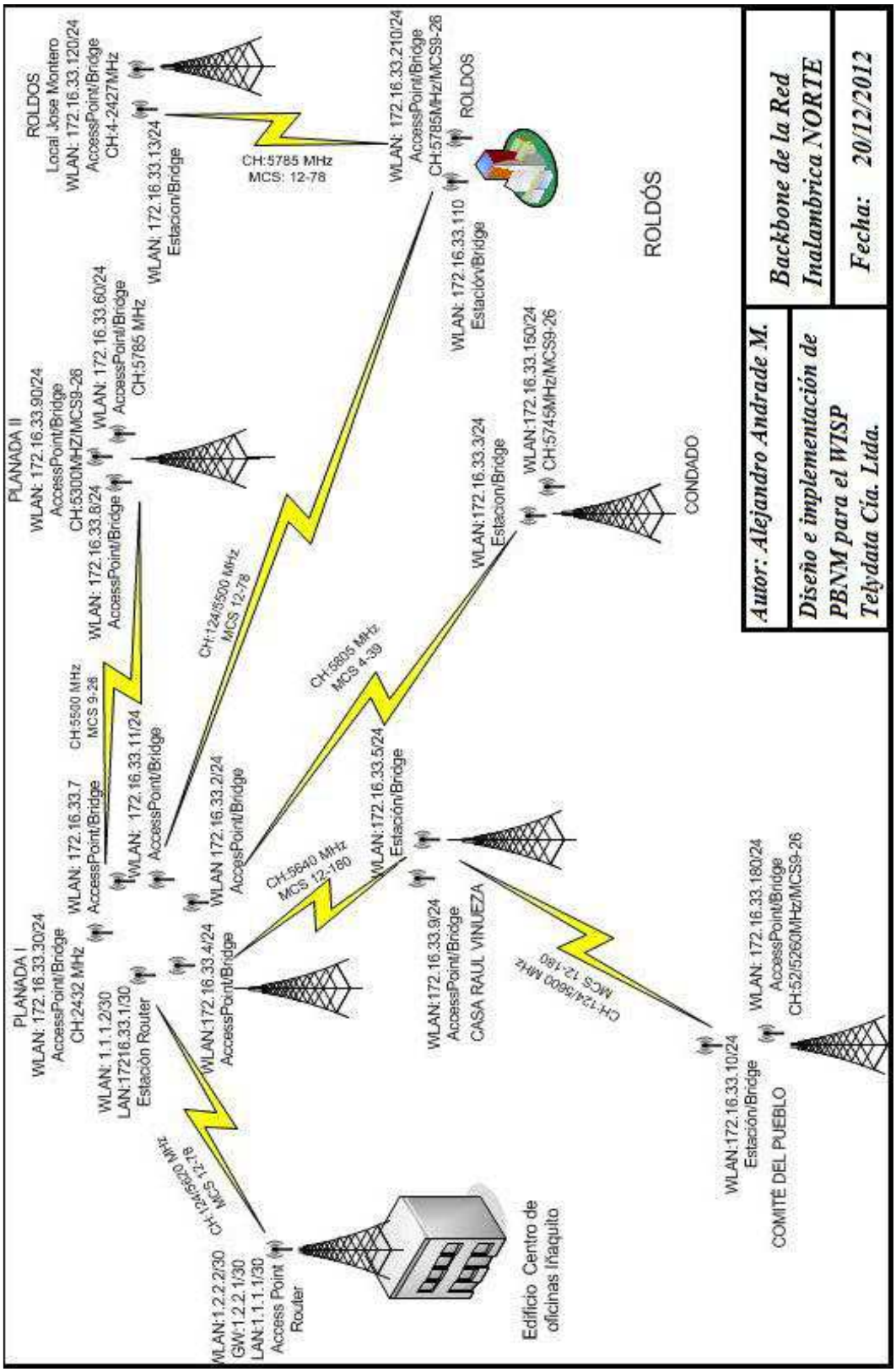



Figura 4. 40: Backbone RIT Norte

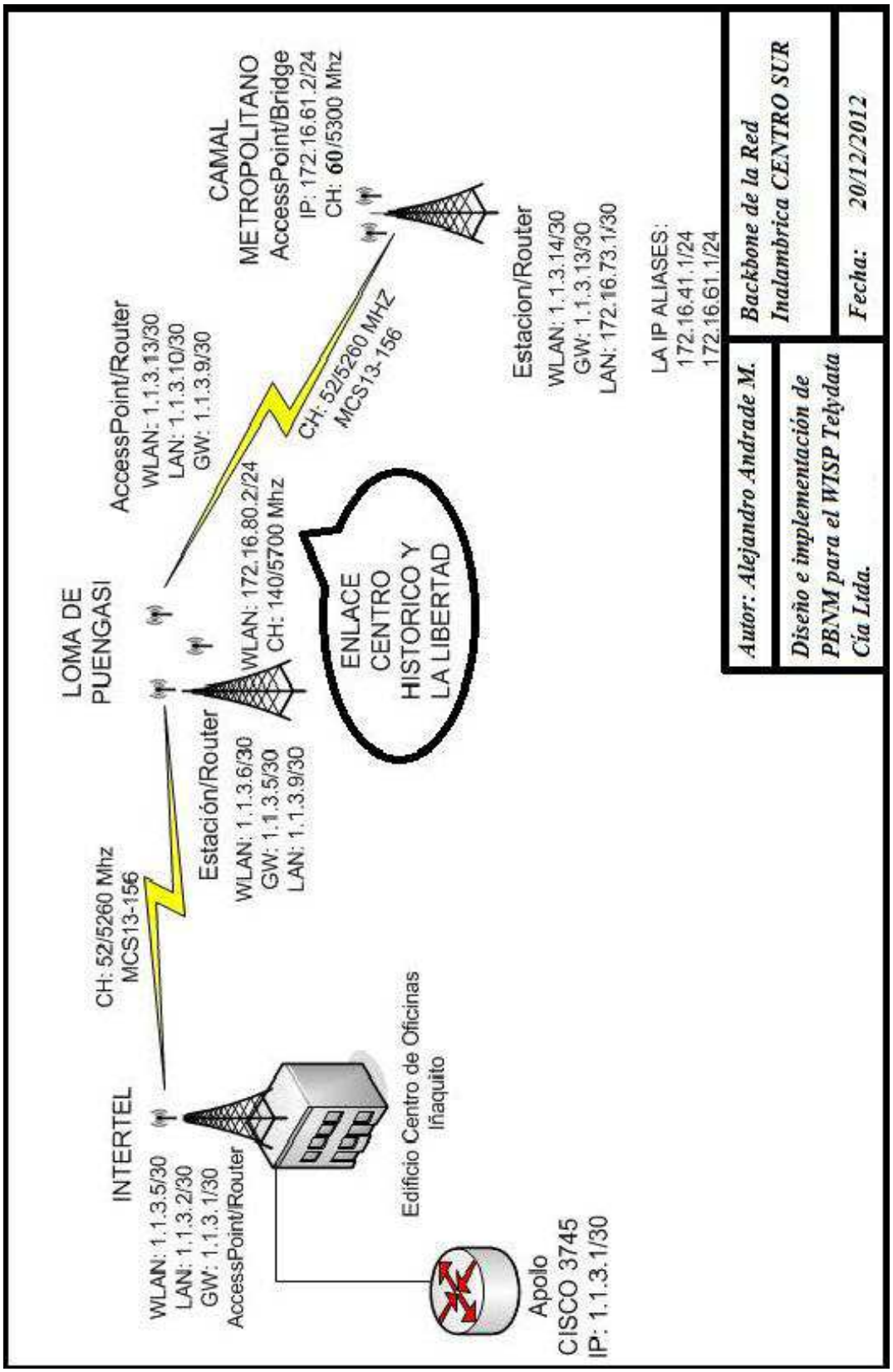



Figura 4. 41: Backbone RIT Sur

## 4.4 IMPLEMENTACIÓN DEL SCRIPT PARA EL MAPEO DE INFORMACIÓN. (SCRIPT *datosClientes*)

Se designa la ubicación del *script* en /etc/mapeo/datosClientes, almacenado en el servidor de Monitoreo. Este *script* permitirá generar los archivos requeridos especificados en la sección 3.3.2.2, para el funcionamiento del módulo ejecutor. La búsqueda de atributos y sus valores se logra a través de los comandos *ldapsearch* y el filtro *grep.* Se utiliza también el comando *gawk* para suprimir caracteres y ordenar los parámetros en los archivos de texto.

A) Generación del archivo controlador-velocidades.conf

Se filtran los atributos *ipHostNumber* y *bandwidth* de aquellos clientes que tienen la categoría del negocio como *corporativo*, tal como se muestra en la Figura 4.42.

```
##############################Filtrado de IP y ancho de banda de cada cliente#############################################
gawk 'BEGIN {print "##############Clientes Corporativos del Norte##########" }' | cat > /etc/gestor/controlador-velocidades.conf
ldapsearch -LLL -x -b "ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net" "(businessCategory=Corporativo)" | grep 'ipHostNumber' > datosClientesUno
ldapsearch -LLL -x -b "ou=Clientes,ou=Usuarios,cn=Manager,dc=telydata,dc=net" "(businessCategory=Corporativo)" | grep 'bandwidth' > datosClientesDos
gawk '{print $2}' FS=":" datosClientesUno | tr -d ' ' | cat > datosClientes1
gawk '{print $2}' FS=":" datosClientesDos | tr -d ' ' |tr -d '[a-z]' | tr -d '[A-Z]' | tr -d '=' | tr -d '_' | tr -d ',' | cat  > datosClientes2
```

Figura 4. 42: Filtrado de direcciones IP y valores de velocidad

Los datos resultantes de ese filtrado son almacenados en archivos temporales. Uno para las direcciones IP y otro para los correspondientes valores de velocidad. Luego, los datos de cada archivo temporal son leídos en un arreglo distinto: ip[n] y rate[k] respectivamente, tal como se muestra en la Figura 4.43.

```
while read line ;do
        set -- $line
        test -z "${1##\#*}" && continue
        test -z "$1"
        ip[n]=$1
        let n++
done <<EOF
$(< datosClientes1)
EOF
while read line ;do
        set -- $line
        test -z "${1##\#*}" && continue
        test -z "$1"
        rate[k]=$1
        let k++
done <<EOF
$(< datosClientes2)
EOF
```

Figura 4. 43: Almacenamiento de direcciones IP y velocidades

Uniendo los datos de cada arreglo y de acuerdo a la Tabla 3.4 y 3.5 del Capítulo 3, se genera el archivo *controlador-velocidades.conf*, el cual contendrá para cada cliente su valor de capacidad de acceso. Se usa el código que se presenta en la Figura 4.44

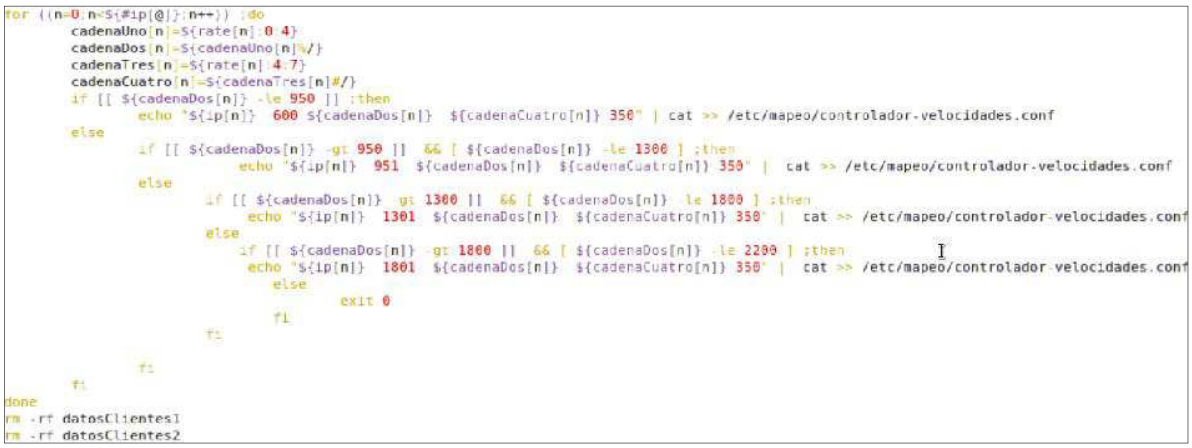



Figura 4. 44: Generación del archivo controlador-velocidades.conf

B) Generación del archivo infotemporal.txt

Para ello se usa el atributo pcimTPCDayOfWeekMask y se escoge el valor que corresponde únicamente a la parte binaria, el dato resultante se envía al archivo *Infotemporal.txt,* tal como se muestra en la Figura 4.45.


Figura 4. 45: Generación del archivo InfoTemporal.txt

C) Generación del archivo ConexionesIP.txt

Para ello se usa el atributo *pcelsIPv4AddrList*. Su valor es una lista de direcciones IP separados por comas (,) las cuales serán eliminadas para almacenar únicamente las direcciones IP individuales en un arreglo, tal como se muestra en la Figura 4.46.

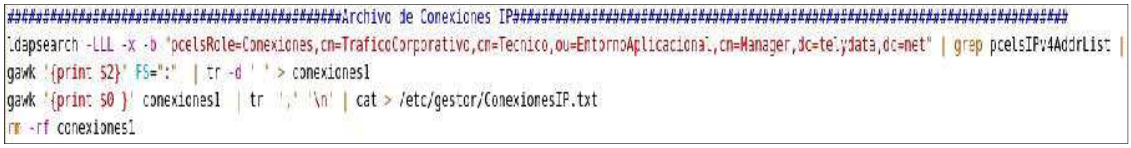



Figura 4. 46: Generación del archivo ConexionesIP.txt

D) Generación del archivo PuertosRed.txt

Para ello se utiliza la clase *pcelsDestinationPortVariableAuxClass*, la cual es usada con el atributo *pcelsIntegerList* para presentar una lista de números de puertos. El código se muestra en la Figura 4.47.


Figura 4. 47: Generación del archivo PuertosRed.txt

E) Generación del archivo Reinicio.txt

Este archivo almacenará las principales direcciones IP pertenecientes al backbone inalámbrico, por lo tanto se usa el atributo *ipHostNumber* de las entradas que se encuentran en la entidad Localizaciones. El código se muestra en la Figura 4.48.


Figura 4. 48: Generación del archivo Reinicio.txt

## 4.5 IMPLEMENTACIÓN DEL MÓDULO EJECUTOR

La implementación del módulo Ejecutor, está constituido por los *script*: tablasnat, controlador y reinicio; además de la implementación de una lista de acceso en el ruteador Apolo. Cada uno de estos componentes se desarrolla según el diseño planteado en el capítulo 3.

### 4.5.1 REQUERIMIENTOS DE HARDWARE DE POLICY SERVER

Las características están directamente relacionadas con los requerimientos físicos del sistema operativo Linux, con la distribución CentOS 6.3 [66]. En la Tabla 4.1 se indican las especificaciones generales.

| Arquitectura | Mínimo memoria RAM | Mínimo espacio en Disco | Versión kernel |
|---|---|---|---|
| i686 Intel 32 bit | 392MB CLI + 512MB GUI, 1GB por CPU lógico | 2 GB | 2.6.32 |

Tabla 4. 1: Requerimientos de Hardware CentOS 6

Por lo tanto, se elige instalar el servidor con las siguientes características de hardware:

- Procesador: Intel Core i3-3220, CPU de 3.30 GHz
- Almacenamiento en Disco: 512 GB
- Memoria RAM de 4GB.
- 1 tarjeta de red adicional D LINK Gigabit Ethernet DGE-5278T

Se procede con la instalación del sistema operativo CentOS 6.3 arquitectura i686, y se emplean las siguientes configuraciones de red:

- Hostname: policy
- Interfaz de red externa eth0: 201.219.5.188/28, Gateway: 201.216.5.185/28
- Interfaz de red interna eth1: 63.173.9.75/28
- DNS primario: 201.219.6.242/28
- DNS secundario: 201.219.5.186/28

### 4.5.2 IMPLEMENTACIÓN DEL COMPONENTE NAT. (Script *tablasnat*)

Se crea un archivo denominado *tablasnat*, en el directorio /etc/rc.d/init.d/ y a continuación se establecen los permisos de ejecución. El contenido del archivo se

divide de dos partes principales. En la primera parte se establece la configuración básica de reenvío de paquetes del y configuraciones básicas de seguridad. En la segunda parte se realiza la traducción de direcciones origen (SNAT). En la Figura 4.49 se indica la primera parte del *script tablasnat*.

```
#Autor: Alejando Andrade Mafla
#Proyecto: Disenio e implementacion de PBNM para el WISP Telydata. Componente NAT
#Fecha: Diciembre del 2012
echo 1 > /proc/sys/net/ipv4/ip_forward
cat /proc/sys/net/ipv4/ip_forward
# Descartar mensajes ICMP echo-request enviados a direcciones Broadcast
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
# Descartar paquetes con enganio ruta origen
echo 0 > /proc/sys/net/ipv4/conf/all/accept_source_route
# Habilitar proteccion contra ataques  TCP SYN
echo 1 > /proc/sys/net/ipv4/tcp_syncookies
# No aceptar mensajes ICMP redirect
echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects
# No enviar mensajes  ICMP redirect
echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
# Habilitar protecion contra envios de direcciones de origen falsas (spoofing)
echo 1 > /proc/sys/net/ipv4/conf/all/rp_filter
# Logear paquetes con direccion origen no valida
echo 0 > /proc/sys/net/ipv4/conf/all/log_martians
#No guardar mensajes en Kernel debido a tener el filesystem lleno
echo 1 > /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses
#Tiempo en el que linux tratara de finalizar una conexion
echo 90 > /proc/sys/net/ipv4/tcp_fin_timeout
#Tiempo para finalizar una conexio no activa
echo 1800 > /proc/sys/net/ipv4/tcp_keepalive_time
```

Figura 4. 49: Configuración básica de Firewall

Como se puede verificar en las dos primeras líneas de código de la Figura 4.49 se habilita en el kernel la funcionalidad de reenvío de paquetes. Con las líneas de código que se encuentran a continuación se pretende cumplir con las siguientes funcionalidades:

- Descartar mensajes ICMP echo-request, usado para que el equipo no responda a mensajes ICMP enviados a direcciones broadcast.

- Descartar paquetes source-routed: usado para que un paquete IP no engañe una ruta, haciéndola creer que estuvo prevista y es de confianza.

- Habilitar protección TCP SYN, evitaría denegación de servicio por ataques de inundación de paquetes TCP con la bandera SYN.

- Permite evitar que un atacante obtenga el tráfico e información del servidor, al redireccionarlo por una vía en concreto por donde este tenga acceso.

- Habilitar protección contra spoofing, esto permite descartar paquetes cuya dirección origen no coinciden con la tabla de enrutamiento del kernel.

- Descartar y logear la dirección IP origen no valida, destino y MAC de ciertos paquetes enviados desde la red. (*log_martians*).

- No guardar mensajes en el kernel debido a que puede llenar el sistema de archivos correspondiente y saturarlo.

- Tiempo en el que Linux tratará de finalizar una conexión cerrando los sockets (en 90 segundos.).

- Tiempo en el que se intenta verificar la inactividad de un cliente luego de haber ejecutado un proceso (en 1800 segundos).

Finalmente se habilita la funcionalidad NAT, a través del enmascaramiento de los paquetes que salen por la interfaz externa eth0. También se habilita fowarding desde la interfaz interna eth1, tal como se muestra en la Figura 4.50.

```
############################COMPONENTE NAT##############################
iptables -A FORWARD -s 0.0.0.0/0 -p tcp -d 0.0.0.0/0 --dport 25 -j REJECT
iptables --table nat --append POSTROUTING --out-interface eth0 -j MASQUERADE  #enmascaramiento de paquetes por la interfaz externa
iptables --append FORWARD --in-interface eth1 -j ACCEPT                       #habilitar reenvio de paquetes por que ingresan por eth1
```

Figura 4. 50: Enmascaramiento de paquetes

#### 4.5.2.1 Configuración de rutas para las redes

Como se verificó en el estado actual de la Red Inalámbrica de Telydata, las redes usadas son:

- 172.16.33.0/24 para el norte (Condado, Planada y Roldós)
- 172.16.41.0/24, 172.16.61.0/24 para el sur (Camal, Ciudadela Ibarra)
- 172.16.80.0/24, para el centro (Centro Histórico, La Libertad)

Por lo tanto se asigna a todas las redes con la puerta de enlace al ruteador Apolo con dirección IP 63.173.96.73, tal como se muestra en la Figura 4.51.

```
[root@policy ~]#
[root@policy ~]# route  add -net 172.16.41.0 netmask 255.255.255.0 gw 63.173.96.73
[root@policy ~]# route  add -net 172.16.33.0 netmask 255.255.255.0 gw 63.173.96.73
[root@policy ~]# route  add -net 172.16.33.0 netmask 255.255.255.0 gw 63.173.96.73
[root@policy ~]# route  add -net 172.16.80.0 netmask 255.255.255.0 gw 63.173.96.73
```

Figura 4. 51: Rutas para las Redes Locales

Tanto el *script* tablasnat como las rutas para las redes locales se incluyen en el archivo /etc/rc.d/rc.local, para que se puedan ejecutar cuando el servidor se reinicie.

```
[root@policy ~]# cat /etc/rc.d/rc.local
#!/bin/sh
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
touch /var/lock/subsys/local
###Rutas Adicionales####
route add -net 172.16.116.0 netmask 255.255.255.0 gw 63.173.96.65
route add -net 172.16.41.0 netmask 255.255.255.0 gw 63.173.96.73
route add -net 172.16.61.0 netmask 255.255.255.0 gw 63.173.96.73
route add -net 172.16.33.0 netmask  255.255.255.0 gw 63.173.96.73
route add -net 172.16.80.0 netmask 255.255.255.0 gw 63.173.96.73
/etc/rc.d/init.d/tablasnat
```

Figura 4. 52: Rutas y tablasnat para la ejecución de inicio del sistema

### 4.5.3 IMPLEMENTACIÓN DEL COMPONENTE CONTROLADOR DE LA CAPACIDAD DE ACCESO. (SCRIPT *controlador*)

Se crea el archivo denominado *controlador.conf,* en el directorio /etc/controlador/, en el cual se guarda las variables de configuración, entre las principales se tienen: interfaces del servidor, velocidad total de descarga y carga del enlace, valor MTU (Máxima Unidad de Transmisión) y asignación de valores a las clases padre. También se realiza la adaptación de los comandos netfilter. Teniendo toda esa información previa, se crea el *script* denominado *controlador* en el directorio: /usr/bin/, mediante el cual se implementan todas las funciones establecidas en la sección 3.4.1.2.3 (Diseño del *script* bash *controlador)*.

En las líneas de código de la Figura 4.53, se llama al archivo de configuración */etc/controlador/controlador.conf* y prueba su existencia. En las siguientes líneas de código se realiza las adaptaciones de funciones netfilter para los comandos iptables, iptables-save, iptables-restore y tc.

```
#!/bin/bash
#Script: Controlador de Ancho de Banda para la RIT
#Proyecto: Diseño e implementacion de un sistema PBNM para el WISP  Telydata.
#Autor:Alejandro Andrade Mafla
#Fecha: Diciembre del 2012
source "/etc/controlador/controlador.conf"
test -r $controlador_velocidades_conf || { echo "archivo de configuracion de velocidades no encontrado: $controlador_velocidades_conf" && exit 1; }

#adaptacion de funciones netfilter
function iptables () { ${iptables_comando:="/sbin/iptables"} "$@"; }
function iptables-save () { ${iptables_save_comando:="/sbin/iptables-save"} "$@"; }
function iptables-restore () { ${iptables_restore_comando:="/sbin/iptables-restore"} "$@"; }
function tc () { ${tc_comando:="/sbin/tc"} "$@"; }
```

Figura 4. 53: Llamada al archivo de configuración

### 4.5.3.1 Función cargar_conf ()

Es la función que carga las variables de direcciones IP, valores de velocidad y valores de las clases en variables globales, para ello se utilizan arreglos. Los valores que almacenará cada arreglo servirán para las siguientes funciones. En la Figura 4.54 se indica el código.

### 4.5.3.2 Función cargar_conexionIP ()  y  cargar_Puerto ()

Estas funciones cargan las direcciones IP y los puertos a los cuales se requieren dar prioridad para su uso posterior. En la Figura 4.55 se indica el código para almacenamiento de direcciones IP en un arreglo denominado *iprio [ ]*. De igual manera se almacenan los puertos en un arreglo denominado *puertoprio [ ].*

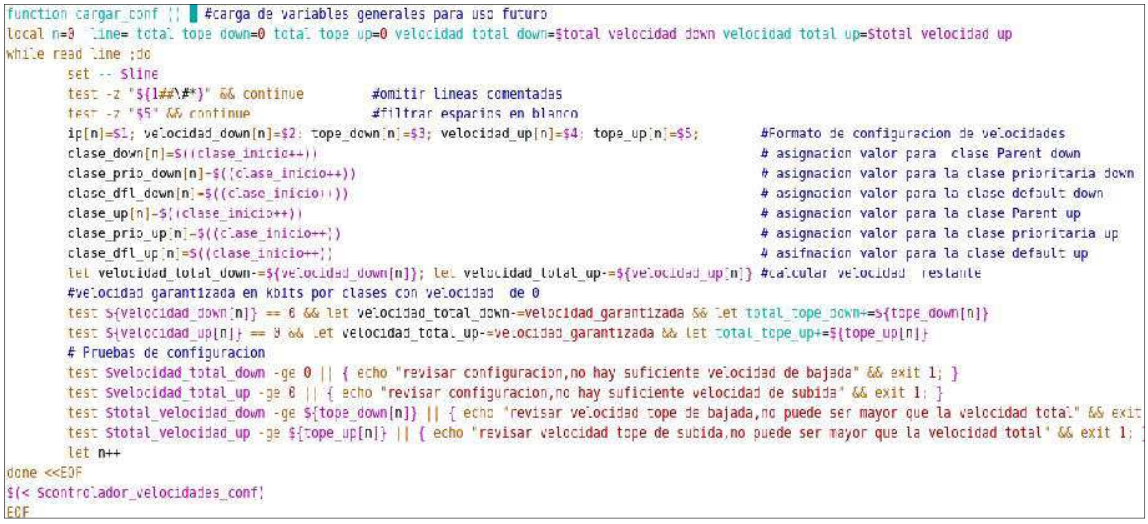
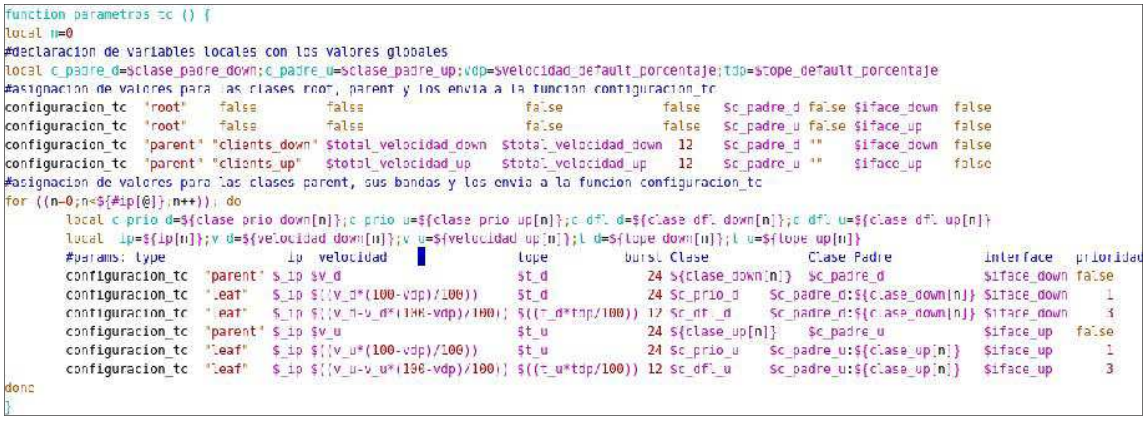



Figura 4. 54: Función cargar_conf()


Figura 4. 55: Función cargar_conexionIP ()

### 4.5.3.3 Función parametros_tc ()

Esta función se encarga de asignar valores para el esquema de configuración HTB de acuerdo al árbol propuesto en la Tabla 3.14. Para cada clase del árbol se definen diez valores en el siguiente orden: nombre del tipo de clase (root, parent o leaf), dirección IP (para el caso de clases parent), valor de velocidad, valor de velocidad tope (para el caso de las clases parent), burst, valor numérico referencial para la clase, valor numérico referencial para la clase padre, interfaz (eth0 ó eth1) y valor de prioridad (para el caso de las clases leaf). Todos estos valores son pasados como parámetros a la función configuración_tc (). En la Figura 4.56 se indica el código.



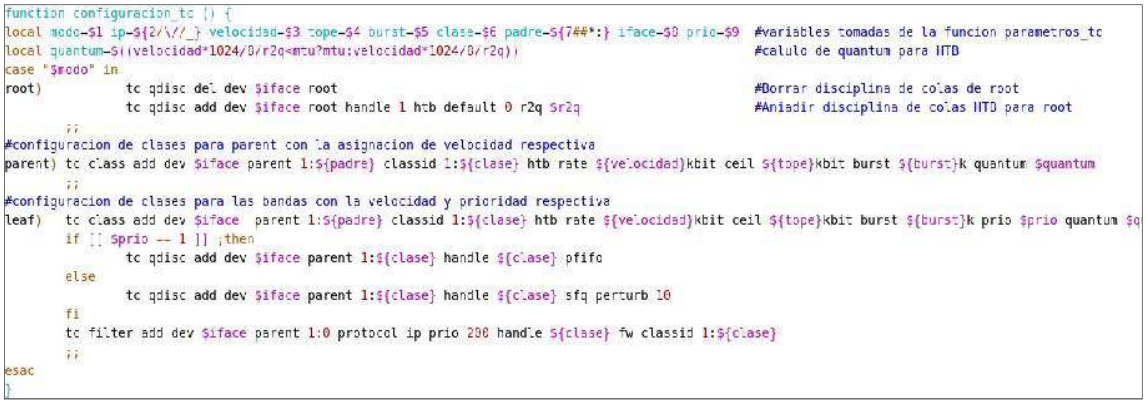
```
function parametros_tc () {
local n=0
#declaracion de variables locales con los valores globales
local c_padre_d=$clase_padre_down;c_padre_u=$clase_padre_up;vdp=$velocidad_default_porcentaje;tdp=$tope_default_porcentaje
#asignacion de valores para las clases root, parent y los envia a la funcion configuracion_tc
configuracion_tc  "root"    false         false                  false           false   $c_padre_d false $iface_down  false
configuracion_tc  "root"    false         false                  false           false   $c_padre_u false $iface_up    false
configuracion_tc  "parent" "clients_down" $total_velocidad_down  $total_velocidad_down  12    $c_padre_d ""    $iface_down  false
configuracion_tc  "parent" "clients_up"   $total_velocidad_up    $total_velocidad_up    12    $c_padre_u ""    $iface_up    false
#asignacion de valores para las clases parent, sus bandas y los envia a la funcion configuracion_tc
for ((n=0;n<${#ip[@]};n++)); do
        local c_prio_d=${clase_prio_down[n]};c_prio_u=${clase_prio_up[n]};c_df_d=${clase_df_down[n]};c_df_u=${clase_df_up[n]}
        local  ip=${ip[n]};v_d=${velocidad_down[n]};v_u=${velocidad_up[n]};t_d=${tope_down[n]};t_u=${tope_up[n]}
        #params: type              ip  velocidad                 tope          burst Clase                Clase Padre              interface    prioridad
        configuracion_tc  "parent" $_ip $v_d                     $t_d            24 ${clase_down[n]}      $c_padre_d               $iface_down false
        configuracion_tc  "leaf"   $_ip $[(v_d*(100-vdp)/100))   $t_d            24 $c_prio_d     $c_padre_d:${clase_down[n]} $iface_down     1
        configuracion_tc  "leaf"   $_ip $[(v_d-v_d*(100-vdp)/100)] $((t_d*tdp/100)) 12 $c_df_d     $c_padre_d:${clase_down[n]} $iface_down     3
        configuracion_tc  "parent" $_ip $v_u                     $t_u            24 ${clase_up[n]}      $c_padre_u               $iface_up   false
        configuracion_tc  "leaf"   $_ip $[(v_u*(100-vdp)/100))   $t_u            24 $c_prio_u     $c_padre_u:${clase_up[n]}   $iface_up       1
        configuracion_tc  "leaf"   $_ip $[(v_u-v_u*(100-vdp)/100)] $((t_u*tdp/100)) 12 $c_df_u     $c_padre_u:${clase_up[n]}   $iface_up       3
done
}
```

Figura 4. 56: Función parametros_tc ()

### 4.5.3.4 Función configuracion_tc ()

Con los valores provistos por la función parametros_tc, se realiza la configuración de clasificado y filtrado de acuerdo a la disciplina de colas HTB.

Se toma en cuenta el mecanismo de diseño de la sección 3.4.1.2.1, en donde se consideran tres clases: root, parent y leaf; para el control y priorización de tráfico en cada dirección IP de cada cliente. La implementación se muestra en la Figura 4.57.



```
function configuracion_tc () {
local modo=$1 ip=${2/\//_} velocidad=$3 tope=$4 burst=$5 clase=$6 padre=${7##*:} iface=$8 prio=$9  #variables tomadas de la funcion parametros_tc
local quantum=$((velocidad*1024/8/r2q<mtu?mtu:velocidad*1024/8/r2q))                               #calulo de quantum para HTB
case "$modo" in
root)           tc qdisc del dev $iface root                                                       #Borrar disciplina de colas de root
                tc qdisc add dev $iface root handle 1 htb default 0 r2q $r2q                       #Aniadir disciplina de colas HTB para root
        ;;
#configuracion de clases para parent con la asignacion de velocidad respectiva
parent) tc class add dev $iface parent 1:${padre} classid 1:${clase} htb rate ${velocidad}kbit ceil ${tope}kbit burst ${burst}k quantum $quantum
        ;;
#configuracion de clases para las bandas con la velocidad y prioridad respectiva
leaf)   tc class add dev $iface  parent 1:${padre} classid 1:${clase} htb rate ${velocidad}kbit ceil ${tope}kbit burst ${burst}k prio $prio quantum $q
        if [[ $prio -- 1 ]] ;then
                tc qdisc add dev $iface parent 1:${clase} handle ${clase} pfifo
        else
                tc qdisc add dev $iface parent 1:${clase} handle ${clase} sfq perturb 10
        fi
        tc filter add dev $iface parent 1:0 protocol ip prio 200 handle ${clase} fw classid 1:${clase}
        ;;
esac
}
```

Figura 4. 57: Función configuracion_tc ()

### 4.5.3.5 Función configuracion_iptables ()

Se encarga de escribir las reglas de marcado por puertos y conexiones para cada dirección IP con la herramienta iptables. Esta función se divide en dos: iptables_alfa() e iptables_beta().

La función iptables_alfa() realiza el nombramiento de tablas generales y envío de parámetros para la configuración de la función iptables_beta(). El código se muestra en la Figura 4.58.

```
# nombramiento de tablas mangle generales
iptables -t mangle -N con.down
iptables -t mangle -A FORWARD -o ${iface_down} -j con.down
iptables -t mangle -A OUTPUT -o ${iface_down} -j con.down
iptables -t mangle -N con.up
iptables -t mangle -A POSTROUTING -o ${iface_up} -j con.up
iptables -t mangle -A FORWARD -o ${iface_up} -j con.up
#Envio de parametros de direcciones IP y clases de las bandas
for ((n=0;n<${#ip[@]};n++)); do
        iptables_beta "down" ${ip[n]} ${clase_prio_down[n]} ${clase_dfl_down[n]}
        iptables_beta "up"   ${ip[n]} ${clase_prio_up[n]}   ${clase_dfl_up[n]}
done
}
```

Figura 4. 58: Función iptables_alfa ()

La función iptables_beta() se encarga de realizar las reglas de marcado de los paquetes cuyos puertos y direcciones IP son obtenidos del directorio LDAP y tendrán prioridad. La implementación se indica en la Figura 4.59.

```
function iptables_beta () {
local n=0
local dir=$1 ip=$2 clase_prio=$3 clase_dfl=$4 host_dir= sentido=
case "$dir" in down) host_dir=d;sentido=s ;; up) host_dir=s;sentido=d ;;esac
iptables -t mangle -N con.${dir}-${ip}
iptables -t mangle -A con.${dir} -${host_dir} ${ip} -j con.${dir}-${ip}
iptables -t mangle -A con.${dir}-${ip} -m mark --mark 0 -m length --length 0:100 -j MARK --set-mark ${clase_prio}
iptables -t mangle -A con.${dir}-${ip} -m mark --mark 0 -p udp -j MARK --set-mark ${clase_prio}
iptables -t mangle -A con.${dir}-${ip} -m mark --mark 0 -p tcp -m multiport --${sentido}ports $cadena -j MARK --set-mark ${clase_prio}
for ((n=0;n<${#iprio[@]};n++)); do
        iptables -t mangle -A con.${dir}-${ip} -m mark --mark 0 -${sentido} ${iprio[n]} -j MARK --set-mark ${clase_prio}
done
iptables -t mangle -A con.${dir}-${ip} -m mark --mark 0 -m helper --helper ftp -j MARK --set-mark ${clase_prio}
iptables -t mangle -A con.${dir}-${ip} -m mark --mark 0 -j MARK --set-mark ${clase_dfl}
iptables -t mangle -A con.${dir}-${ip} -j ACCEPT
}
```

Figura 4. 59: Función iptables_beta()

### 4.5.3.6 Uso de parámetros del *script controlador*

El *script* general se ejecutará de acuerdo a tres parámetros de entrada, los mismos que permitirán ejecutar las reglas tc, reglas iptables o imprimir en pantalla las reglas generadas por el *script* general, tal como se indica en la Figura 4.60.

```
#Ejecucion de los parametros de Entrada
case "$1" in
ejecutar|listar_tc)
        cargar_conf                                             #cargar variables generales
        cargar_conexionIP                                       #cargar direcciones IP destino
        cargar_Puerto                                           #cargar Puertos
        test "$1" == "listar_tc" && tc () { echo tc "$@"; }     #Listar configuracion TC
        test "$1" == "ejecutar" && configuracion_iptables
        parametros_tc
        ;;
listar_iptables)
        cargar_conf
        cargar_conexionIP
        cargar_Puerto
        configuracion_iptables "listar"
        ;;
loadvars)       cargar_conf
                ;;
*)      cat <<-EOF
        Usage:
        $0 ejecutar         #ejecuta las reglas tc e iptables
        $0 listar_tc        #imprime en pantalla las reglas tc
        $0 listar_iptables #imprime en pantalla las reglas iptables
```

Figura 4. 60: Uso de parámetros para la ejecución del *script* controlador

### 4.5.4 IMPLEMENTACIÓN DEL COMPONENTE REINICIOS. (SCRIPT *reinicio*)

Se crea el *script* denominado *reinicio*, en el directorio: /etc/ejecutor/, mediante el cual se realizan los reinicios de las principales antenas del backbone inalámbrico. En la Figura 4.61 se indica la primera parte del *script* en donde se almacena las direcciones IP en el arreglo ip[ ] que se posteriormente se reiniciarán. Además se indica el código para reiniciar el primer AP que servirá como ejemplo para todos los AP del backbone de la RIT.

```
#!/bin/bash
n=0
while read line ;do
        set -- $line
        test -z "${1##\#*}" && continue
        test -z "$1"
        ip[n]=$1
        let n++
done <<EOF
$(< /etc/ejecutor/Reinicio.txt)
EOF

HOST="${ip[0]}"
USER="ubnt"
PASS="wlely2012data"
VAR=$(expect -c "
spawn ssh -o StrictHostKeyChecking=no $USER@$HOST reboot
match_max 100000
expect \"*?assword:*\"
send -- \"$PASS\r\"
send -- \"\r\"
expect eof
")
echo "reiniciando....................."
```

Figura 4. 61: *Script* para el reinicio de antenas

Se utiliza el comando *ssh* junto con las direcciones IP de las antenas de las antenas obtenidas del archivo Reinicio.txt.

Para el funcionamiento del *script* se declaran tres variables principales:

- HOST, que contiene la dirección IP de la antena que se requiere reiniciar.

- USUARIO, contiene el nombre de usuario para el ingreso a las antenas vía SSH. Se tienen dos nombres posibles: *ubnt* o *root.*

- PASS, que contiene la contraseña de ingreso a la antena. Generalmente es la misma para todas las antenas del backbone. Una vez ingresada a la antena vía SSH, se ejecuta el comando *reboot* que permitirá el reinicio.

Esta forma de reinicio es posible gracias a que el sistema operativo AirOS perteneciente a la infraestructura Ubiquiti Networks tiene embebida una solución Ubuntu Linux 6.10, lo cual posibilita el uso de CLI desde cualquier host que posea alguna distribución Linux. El reinicio es independiente para todas las antenas, ya que no todas tienen el mismo usuario o contraseña. Además, en el *script* aparecen los siguientes comandos importantes:

- Expect: permite ejecutar comandos y esperar una salida determinada por stdout[75]. Con la opción –c, se establece que los comandos sean ejecutados en el orden que se establecen en el *script*.

- Send: envía comandos por stdin[76] con el proceso, en este caso se envía el comando reboot.

Otro de los aspectos que hay que recalcar es que cuando se realiza una conexión con SSH con las antenas, se elimina la opción de fingerprint[77], a través de la opción StrictHostKeyChecking=no, ya que esto es un problema al momento de automatizar las conexiones.

[75] Stdout(standard output): representa el descriptor de salida resultante del comando
[76] Stdin(standard input): descriptor donde el comando espera encontrar su entrada
[77] Fingerprint: representa una huella digital única que identifica un servidor en específico

### 4.5.5 CONFIGURACIÓN DE LA LISTA DE ACCESO ENTRE EL RUTEADOR CISCO 3745 Y POLICY SERVER

Se revisa la configuración de la interfaz VLAN 3, ya que es en esta interfaz donde se tiene conectada la red inalámbrica de Telydata.

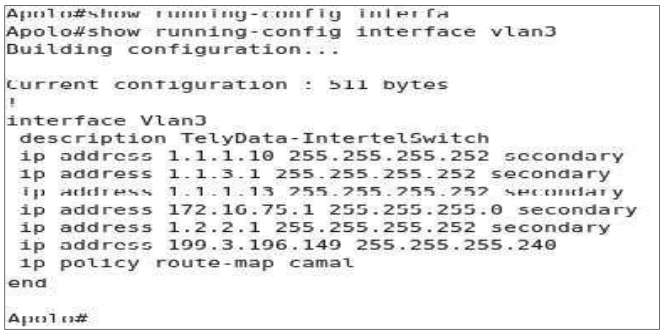



Figura 4. 62: Interfaz VLAN3 del Router Cisco 3745

Como se puede verificar en la Figura 4.62, la interfaz tiene varias direcciones IP secundarias especificadas, de las cuales: 1.1.3.1/30 pertenece a la RIT Sur, y la dirección IP 1.2.2.1/30, pertenece a la RIT Norte. La interfaz cuenta con el método de ruteo para redistribución de tráfico denominado *route-map.* A través de este se puede separar el tráfico de clientes que ocupan la misma infraestructura de la RIT por diferentes proxies. El *route-map* preestablecido se denomina *camal,* el cual mantiene la tabla de las diferentes rutas de la red inalámbrica de Telydata. (Ver Figura 4.63.)

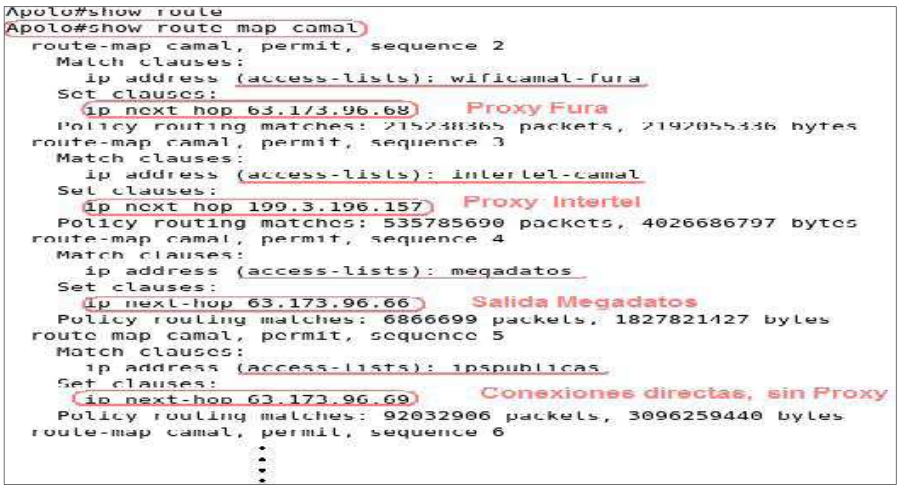



Figura 4. 63: Datos del *route-map camal*

Como se puede verificar en la Figura 4.63, la configuración del *route-map* denominado *camal* tiene para cada enlace, los siguientes datos:

- Secuencia de ejecución (*sequence*), la cual establece el orden o prioridad en el que se ejecuta la ruta en la tabla. En este caso el primer valor de secuencia empieza desde el valor 2.

- Lista de Acceso (*Access-List*), es el nombre de una lista, en la cual se pretende incluir todos los host que pasarán por un determinado proxy. También contiene nombres de listas de acceso que tienen salida directa al Internet, sin proxy.

- Próximo Salto (*ip next-hop*): para que el paquete continúe hacia alguno de los proxies de la red de core del WISP. En caso de que el tráfico de un host tenga salida directa, se especifica la dirección IP del switch (capa 3) o ruteador correspondiente.

#### 4.5.5.1 Creación de la lista de acceso

A continuación se crea una lista de acceso extendida denominada *policy-server* y posteriormente se incluye a los clientes de la RIT que van a usar el sistema de gestión presentado en el presente proyecto. A manera de ejemplo, en la Figura 4.64 se indica la inclusión a la lista *policy-server*, a un host de la red Iñaquito Sur, con la dirección IP 172.16.41.124/24, perteneciente al cliente Pilar Rodríguez.

```
Apolo#configure term
Apolo#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Apolo(config)#ip access
Apolo(config)#ip access-list exten
Apolo(config)#ip access-list extended policy-server
Apolo(config-ext-nacl)#permit ip host 172.16.41.124 any
Apolo(config-ext-nacl)#
Apolo#
```

Figura 4. 64: Creación de la lista de acceso denominada como *policy-server*

### 4.5.5.2 Asignación de la ruta por defecto

Según la Figura 4.63 (sección 4.5.5), el *route-map camal* no ocupa la secuencia 1, al momento se encuentra libre, por lo tanto con esta secuencia se procede a configurar la lista de acceso con ruta hacia Policy Server, tal como se muestra en la Figura 4.65.

```
Apolo#conf
Apolo#configure ter
Apolo#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Apolo(config)#route
Apolo(config)#route-
Apolo(config)#route-map camal permit 1
Apolo(config-route-map)#match ip addre
Apolo(config-route-map)#match ip address  policy-server
Apolo(config-route-map)#set ip nex
Apolo(config-route-map)#set ip next-hop  63.173.96.75
Apolo(config-route-map)#
Apolo#
Apolo#
```

Figura 4. 65: Configuración de la lista de acceso

Posteriormente se comprueba si se encuentran coincidencias (*matches*) del host que se incluyó anteriormente (Figura 4.64) en la lista de acceso *policy-server*.

```
Apolo#show ip access
Apolo#show ip access-lists  policy-server
Extended IP access list policy-server
    10 permit ip host 172.16.41.15 any (11329212 matches)
    20 permit ip host 72.16.33.211 any
    40 permit ip host 172.16.41.125 any (91738170 matches)
    50 permit ip host 172.16.41.50 any (8900174 matches)
    70 permit ip host 172.16.33.225 any (8471735 matches)
    90 permit ip host 172.16.41.124 any (31741079 matches)
    100 permit ip host 172.16.80.47 any (35607817 matches)
    110 permit ip host 172.16.41.155 any (1521864 matches)
    120 permit ip host 172.16.33.185 any (212939 matches)
Apolo#
```

Figura 4. 66: Tráfico por la lista de acceso

La Figura 4.66 indica las coincidencias (matches) de los host incluidos en la lista de acceso denominada *policy-server*, por lo tanto, es una prueba de que la lista de acceso se encuentra configurada correctamente. Hay que aclarar que en la lista de acceso la inclusión es por host, más no una inclusión por red, debido a que por política del gerente técnico, la lista de acceso por defecto para la RIT se denomina *prometeo*, y es ahí donde se incluyen las redes.

## 4.6 CONFIGURACIÓN DE NFS (NETWORK FILE SYSTEM) PARA EL ENVÍO DE INFORMACIÓN GESTOR-EJECUTOR

El servidor NFS se lo instala en la Monitoreo Server, es ahí donde se verifica que existan los siguientes servicios: *rpc.mountd*, *nfsd*, *portmap*, *nfslock*. El directorio /etc/gestor/, es el que se va a exportar ya que contiene los diferentes archivos implementados en la sección 4.4. Por lo tanto, se agrega la referencia de dicho directorio en el archivo /etc/exports con permisos de lectura y escritura, como se indica en la Figura 4.67.


Figura 4. 67: Exportación del archivo /etc/ exports

Finalmente se añaden las reglas respectivas para que en Monitoreo Server se acepten todo tipo de paquetes, ya que este también cuenta con un firewall por estar expuesto al Internet. El cliente NFS se lo realiza en Policy Server, para ello se crea el archivo /mnt/traduccion/, con los permisos requeridos para ser leído y modificado porque es en este directorio donde se montarán los archivos del directorio /etc/gestor de Monitoreo Server.

Finalmente se realiza el montaje simple del tipo *nfs* del directorio /etc/gestor a /mnt/traducción en el Policy Server. Ver Figura 4.68.

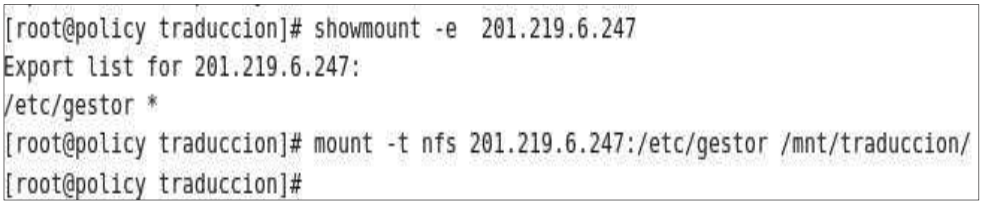



Figura 4. 68: Montaje de archivos en Policy Server

## 4.7. COSTOS REFERENCIALES DEL PROYECTO

### 4.7.1. HARDWARE DEL SERVIDOR DE MONITOREO

El Monitoreo Server está actualmente operativo en el NOC de Telydata. Se requirió instalar una memoria RAM, según los requerimientos establecidos en 4.2.1.1. Las características de la memoria se indican a continuación:

- Tipo: DDR2
- Módulo: DIMM
- Velocidad de procesamiento: 662 MHz
- Tamaño: 2 GB

La memoria tiene un costo de 30 dólares. Se la adquirió al proveedor Tecnomega C.A.

### 4.7.2. HARDWARE DEL SERVIDOR POLICY SERVER

En la Tabla 4.2 se indica el detalle y el costo de cada uno de los elementos que fueron adquiridos en diferentes centros de distribución como lo son: Compuaxir y Tecnomega C.A. Ver **Anexo 8**. El costo total es equivalente a 440,16 dólares.

| Detalle | Precio(dólares) |
|---|---|
| Procesador Intel Core i3 | 140 |
| MainBoard Intel Core i7 | 71 |
| Disco Duro Hitachi 500 GB 7200 rpm | 100 |
| Tarjeta de Red D-LINK | 24 |
| DVD ROM OLG | 42 |
| Case Nobutech | 16 |
| Subtotal | 393 |
| 12 % IVA | 47,17 |
| TOTAL | 440,16 |

Tabla 4. 2: Costo Hardware Policy Server

### 4.7.3 CABLEADO Y MONTAJE DE EQUIPOS

Debido a que el sistema se encuentra dentro de la red de core de red de Telydata, el cableado no fue necesario realizarlo, ni invertir en elementos pasivos para ello. El Monitoreo Server se ubica en el área de oficinas de la empresa. El servidor Policy Server fue montado sobre el rack principal de servidores del ISP en el área del nodo principal, sin ningún requerimiento adicional. Las áreas especificadas están en el quinto piso del Edificio Reinoso en la avenida Amazonas N69-159, en Quito.

Sin embargo, se estima costo por instalación de la memoria RAM en el servidor de Monitoreo, instalación de una tarjeta de red en Policy Server y las conexiones respectivas en la red por un valor de 150 dólares.

### 4.7.4. COSTO DE IMPLEMENTACIÓN DEL PROYECTO

Se considera la inversión en horas de trabajo por el autor del presente proyecto, en cuanto a la configuración, implementación del entorno de gestión. Si cada hora de trabajo cuesta 40 dólares, se realizan las Tablas 4.3, 4.4 y 4.5.

- Configuración de Monitoreo Server

| Detalle | No. HORAS | Precio(dólares) |
|---|---|---|
| Instalación de parches necesarios | 1 | 40 |
| Configuración de OpenLDAP y esquemas | 4 | 160 |
| Implementación del Directorio de Información | 6 | 240 |
| Implementación de interfaces gráficos de Gestión | 4 | 160 |
| Implementación del *script datosClientes* | 4 | 160 |
| Subtotal | | 760 |
| 12 % IVA | | 91,20 |
| TOTAL | | 851,20 |

Tabla 4. 3: Costo configuración Monitoreo server

- Instalación y configuración de Policy Server

| Detalle | No. HORAS | Precio(dólares) |
|---|---|---|
| Instalación de Linux y parches necesarios | 4 | 160 |
| Diseño e implementación del *script* controlador | 10 | 400 |
| Diseño e implementación de *script* reinicios | 7 | 280 |
| Implementación del componente NAT | 4 | 160 |
| Subtotal | | 1000 |
| 12 % IVA | | 120 |
| TOTAL | | 1120 |

Tabla 4. 4: Costo Instalación y configuración de Policy Server

- Configuración de equipos secundarios y pruebas:

| Detalle | No. HORAS | Precio(dólares) |
|---|---|---|
| Configuración Router Cisco Apolo | 2 | 80 |
| Configuración NFS | 4 | 160 |
| Configuración SNMP en la RIT | 6 | 240 |
| Pruebas | 4 | 160 |
| Subtotal | | 640 |
| 12 % IVA | | 76,80 |
| TOTAL | | 716,80 |

Tabla 4. 5: Costo configuración de equipos secundarios

El costo resultante al sumar los totales de las Tablas 4.3, Tabla 4.4 y Tabla 4.5 es 2688 dólares.

### 4.7.5 COSTO TOTAL

En la Tabla 4.6 se detalla el costo total resultante por la implementación del proyecto, el cual es equivalente a **3705, 14 dólares.**

| DESCRIPCION | SUBTOTAL(dólares) |
|---|---|
| Hardware servidor de Monitoreo | 30,00 |
| Hardware servidor Policy Server | 440,16 |
| Cableado y Montaje de Equipos | 150,00 |
| Costos de Operación | 2688,00 |
| Subtotal | 3308,16 |
| IVA (12 %) | 396,98 |
| **TOTAL** | **3705,14** |

Tabla 4. 6: Costo total del Proyecto

## 4.8 PRUEBAS DEL SISTEMA DE GESTIÓN

Luego de la implementación se espera que el departamento técnico pueda interactuar con el sistema de gestión, administrando obteniendo información y usuarios, controlando la capacidad de acceso al canal para cada cliente y ejecutando reinicios programados de las antenas del backbone. Se estableció un periodo de pruebas desde el lunes 24 de diciembre del 2012 hasta el viernes 4 de enero del 2013, durante horas laborales.

### 4.8.1 REPOSITORIO DE INFORMACIÓN Y GESTIÓN DE LA INFORMACIÓN

Principalmente se requiere demostrar consultas, modificación o eliminación de datos técnicos de clientes de la RIT, tanto desde Monitoreo Server, como desde cualquier parte del Internet. Esta prueba debería realizarla cualquier miembro del departamento Técnico.

Caso1: Búsqueda de datos del cliente David Castro perteneciente al enlace Planada de la RIT Norte, desde Monitoreo Server. Ver Figura 4.69.

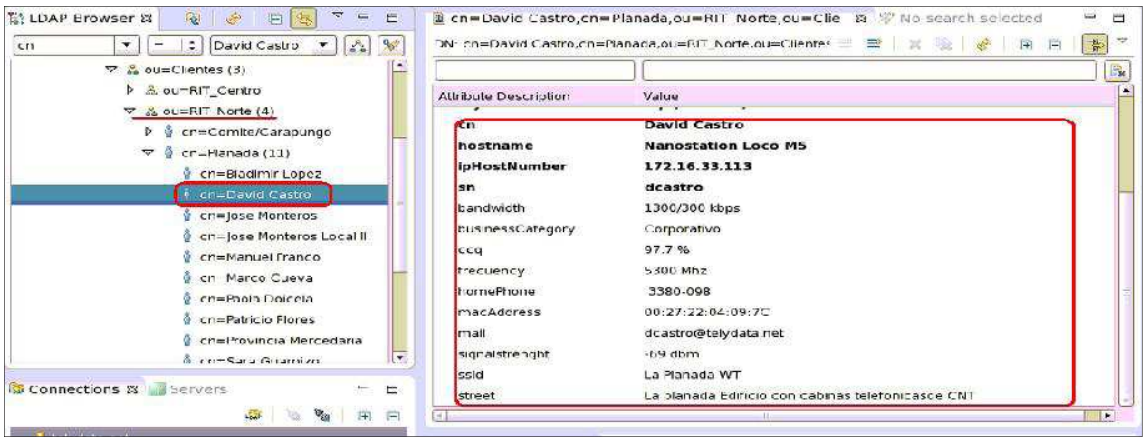



Figura 4. 69: Pruebas de Búsqueda y Modificación de datos en Monitoreo Server

Caso2: Búsqueda de datos del cliente David Castro desde cualquier parte del Internet. Para realizar esta prueba se utiliza la herramienta Softerra LDAP Browser, instalada en un computador con sistema operativo Windows. Ver Figura 4.70.

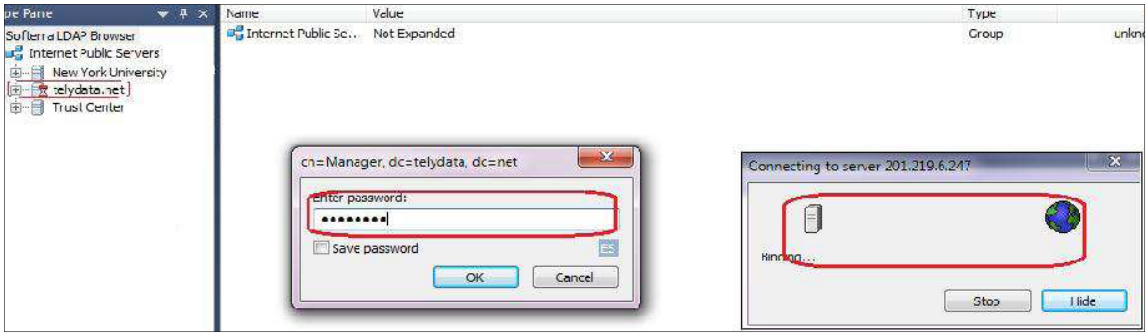



Figura 4. 70: Logín remoto al repositorio de información

La Figura 4.71, muestra que la búsqueda remota se la ha realizado desde el domicilio del Sr. Alejandro Andrade, a través de un computador conectado a un ruteador wireless, con dirección IP: 186.69.186.226/24.

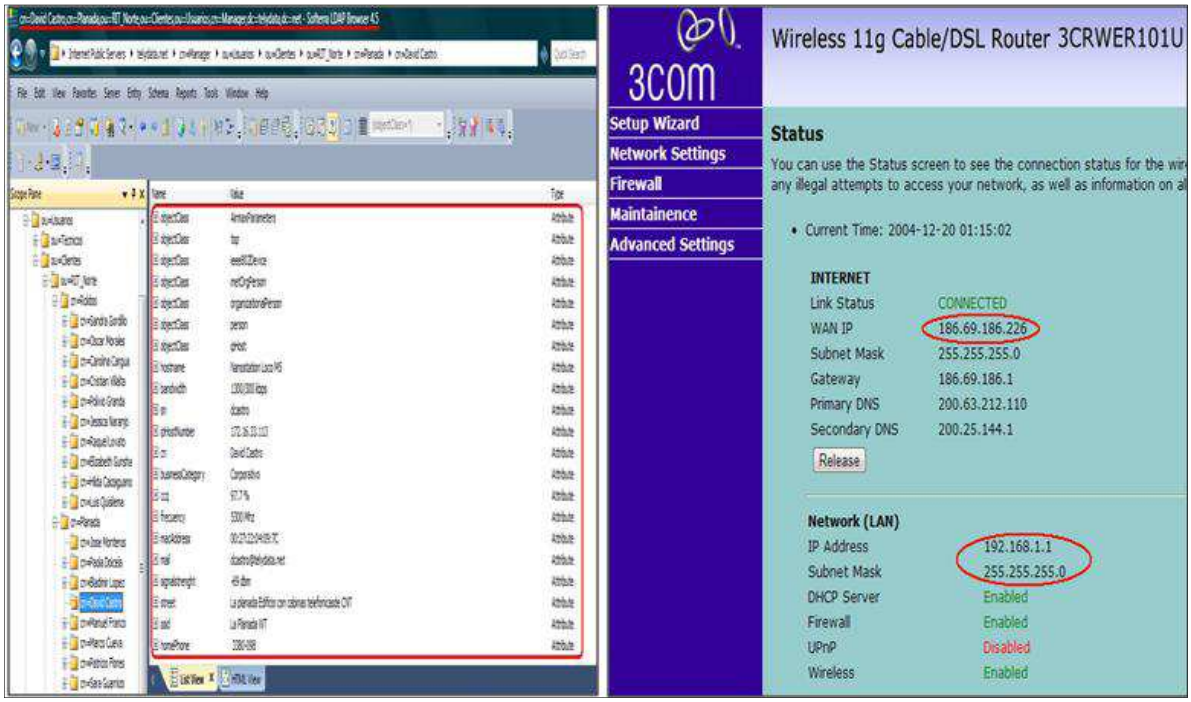



Figura 4. 71: Prueba de consulta de datos desde el Internet

En la Figura 4.71 también se puede verificar se tiene fácil acceso a los datos de David Castro, entre los datos más importantes tienen: dirección IP, bandwidth, ccq, signalstrength o MAC, es la información importante necesaria para brindar soporte técnico al cliente.

Modificación de Información

Para realizar esta prueba se realiza el cambio de valores como la dirección IP y capacidad del canal para el cliente Janeth Guananga perteneciente a la RIT Sur. En el Figura 4.72 se indica que es posible la edición de información de los clientes.

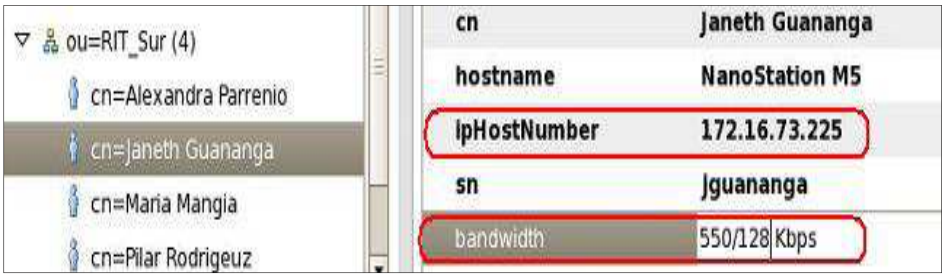



Figura 4. 72: Prueba de Modificación de datos de los clientes

Para probar que es posible agregar y eliminar clientes se añade y elimina a una nueva cliente denominada María Belén Mejía perteneciente a la RIT Norte, de la sección Comité/Carapungo, con todos los datos especificados en el modelo de información, como se indica en la Figura 4.73.

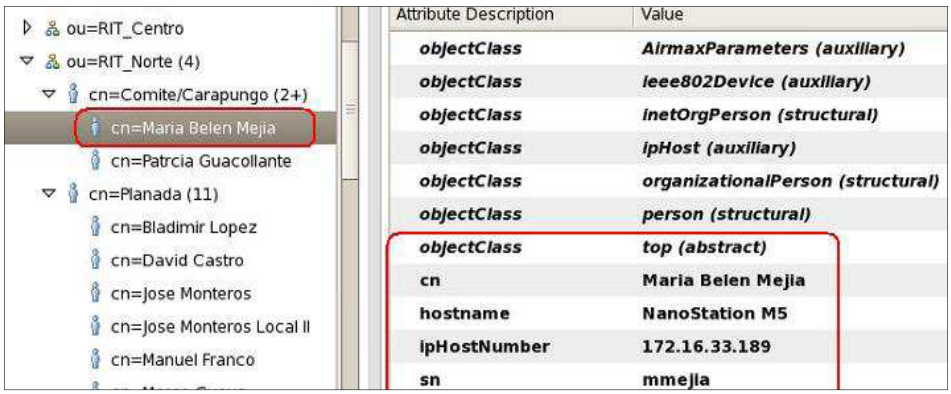



Figura 4. 73: Ingreso de clientes en el repositorio

### 4.8.2 CONTROL DE LA CAPACIDAD DE ACCESO POR CLIENTE

Para realizar esta prueba se toma el ejemplo de dos clientes: Alexandra Parreño de la RIT Sur y Paola Doicela de la RIT Norte

a. Caso Alexandra Parreño cuya dirección IP es 172.16.41.125/24, a quién se le aplico el control de capacidad, con su velocidad contratada 2200/350 Kbps el día 24 de diciembre del 2012 desde las 13h00. En la Figura 4.74 se indica el resultado.

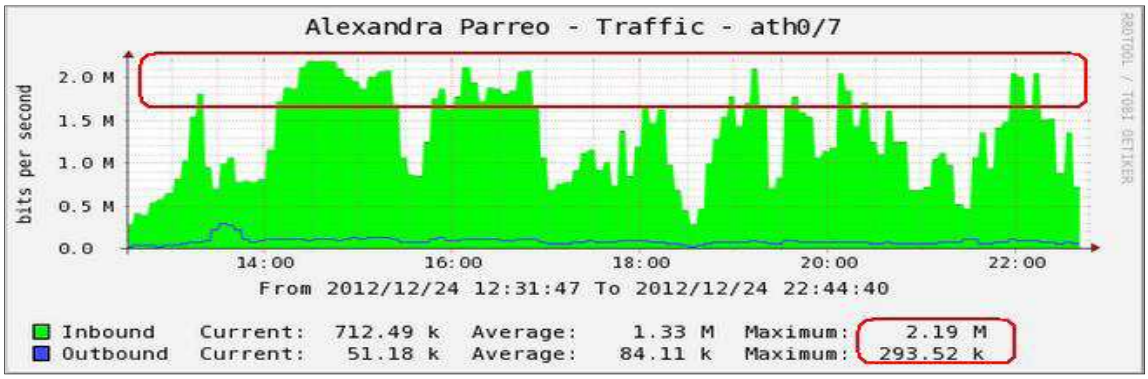



Figura 4. 74: Control de la capacidad de acceso sobre la cliente Alexandra Parreño

Como se puede verificar en la Figura 4.74, se cumple con el control de capacidad de acceso ya que se alcanza una velocidad máxima de descarga de 2190 Kbps. y 293.52 Kbps de carga. En el periodo desde las 14h25 a 14h45 se denota de mejor manera el control, al no permitir rebasar lo contratado.

b. Caso Paola Doicela: cuya dirección IP es 172.16.33.91/24 a quién se le aplicó el control de la capacidad de acceso, a su velocidad contratada de 1400/350 Kbps el día 24 de diciembre del 2012 desde las 13h00. En la Figura 4.75 se indica el resultado.

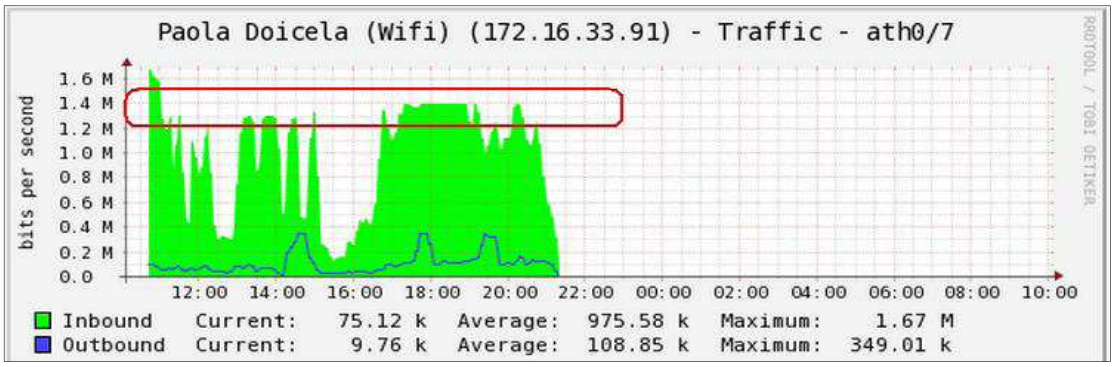



Figura 4. 75: Control de la capacidad de acceso sobre el cliente Paola Doicela

En la Figura 4.75, se indica que también cumple con el control de capacidad de acceso, ya que desde las 13h00 se alcanza una velocidad máxima de descarga de 1400 y 349.01 Kbps de carga. En el periodo de 17h00 a 19h00 se denota de mejor manera el control al no permitir rebasar lo contratado.

#### 4.8.2.1 Demostración del procedimiento de control de la capacidad de acceso.

Para hacer más descriptivo el procedimiento se limita la capacidad de acceso al cliente Paola Doicela a 950/350 Kbps durante dos horas (Ver Figura 4.76).

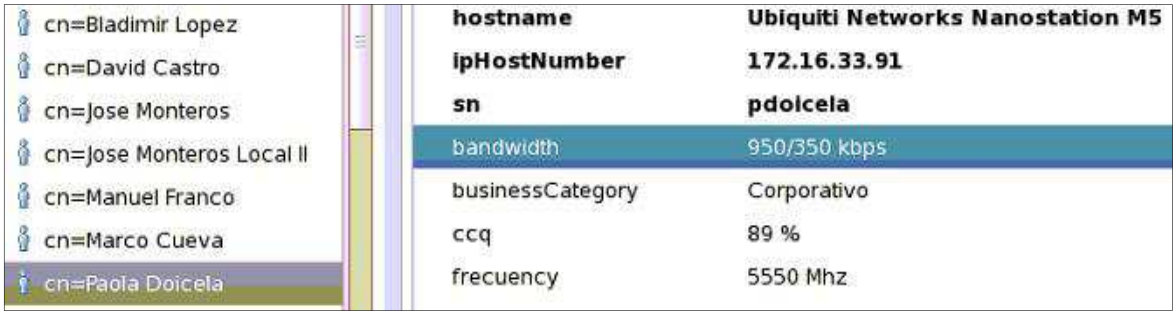



Figura 4. 76: Regulación de la capacidad de acceso. Cliente Paola Doicela

Por motivo de pruebas se establece que la ejecución de los *scripts* sea cada hora, por lo tanto se debería esperar que en el archivo controlador-velocidades.conf en Policy Server se actualice y tome efecto de dicha asignación de la capacidad de acceso para los clientes, luego de una hora. La prueba se inició a las 11h45 del día 25 de diciembre del 2012. (Ver Figura 4.77).
Además, en la Figura 4.77, también se verifica la dirección IP del equipo y la fecha en que se han realizado las pruebas.

Consecuentemente, desde las 12h50 se ejecuta el controlador de capacidad de acceso para los clientes. En la Figura 4.78, se verifica que la capacidad de acceso para el cliente Paola Doicela se fija en un valor máximo de 950/124 Kbps. (dato proporcionado por el servidor CACTI.)

**archivo: /etc/ejecutor/controlador-velocidades.conf**
**Hora: 11h45**

```
############Clientes Corporativos
172.16.33.96   950   1300 300 350
172.16.41.125 2000 2200  300 350
172.16.80.40   950   950 300 350
172.16.33.91 1200 1400  350 350
172.16.33.234 1600 1800  300 350
```

**archivo: /etc/ejecutor/controlador-velocidades.conf**
**luego de una hora. Hora: 12h50**

```
############Clientes Corporativos
172.16.33.96   950   1300 300 350
172.16.41.125 2000 2200  300 350
172.16.80.40   950   950 300 350
172.16.33.91   950   950 350 350
172.16.33.234 1600 1800  300 350
```

```
[root@policy ~]# date
Tue Dec 25 13:16:30 ECT 2012
[root@policy ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr E8:40:F2:ED:0F:F8
          inet addr:201.219.5.188  Bcast:201.219.5.191  Mask:255.255.255.240
          inet6 addr: fe80::ea40:f2ff:feed:ff8/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:100029371 errors:0 dropped:0 overruns:0 frame:0
```

Figura 4. 77: Actualización del Archivo /etc/ejecutor/controlador-velocidades.conf

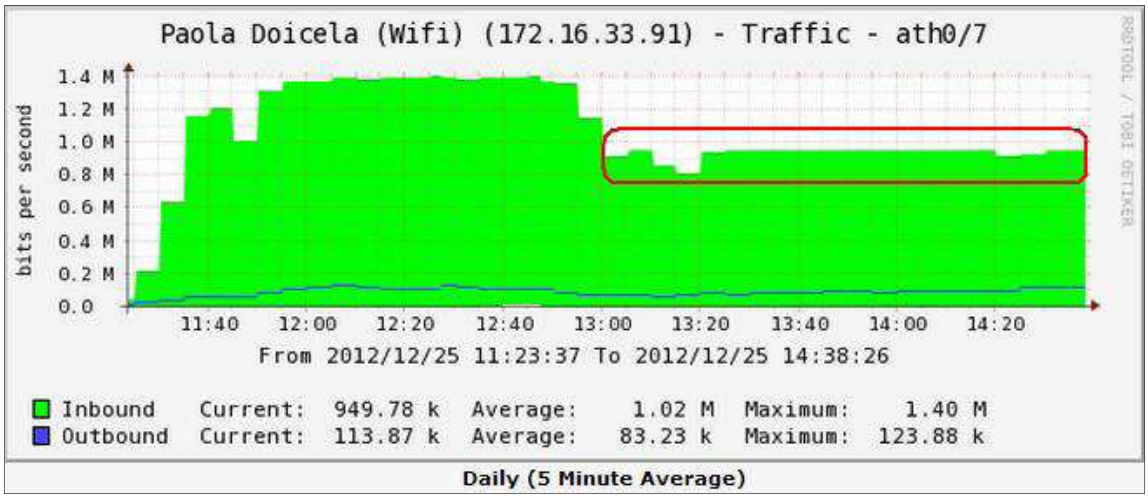



Figura 4. 78: Verificación de la regulación a la cliente Paola Doicela

En la Figura 4.79 se verifica la velocidad marcada en la antena perteneciente a la cliente Paola Doicela durante el horario de prueba, lo que confirma que el control de la capacidad de acceso es correcta.

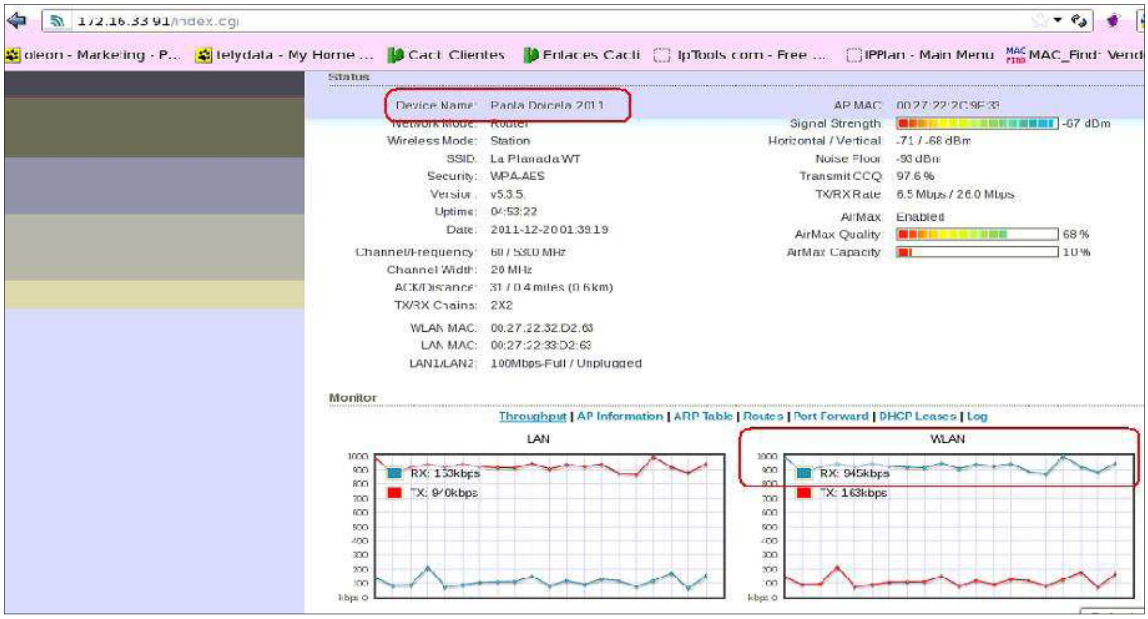



Figura 4. 79: Throughput marcado en la antena de la cliente Paola Doicela

En la Figura 4.80 se demuestra la formación del árbol diseñado en la sección 3.4.1.2.1 para permitir el control y priorización del tráfico para cada cliente, en este caso para la dirección IP del cliente Paola Doicela. Dicho árbol de clases es construido con el componente netfilter de Linux y las correspondientes propiedades de la interfaz.

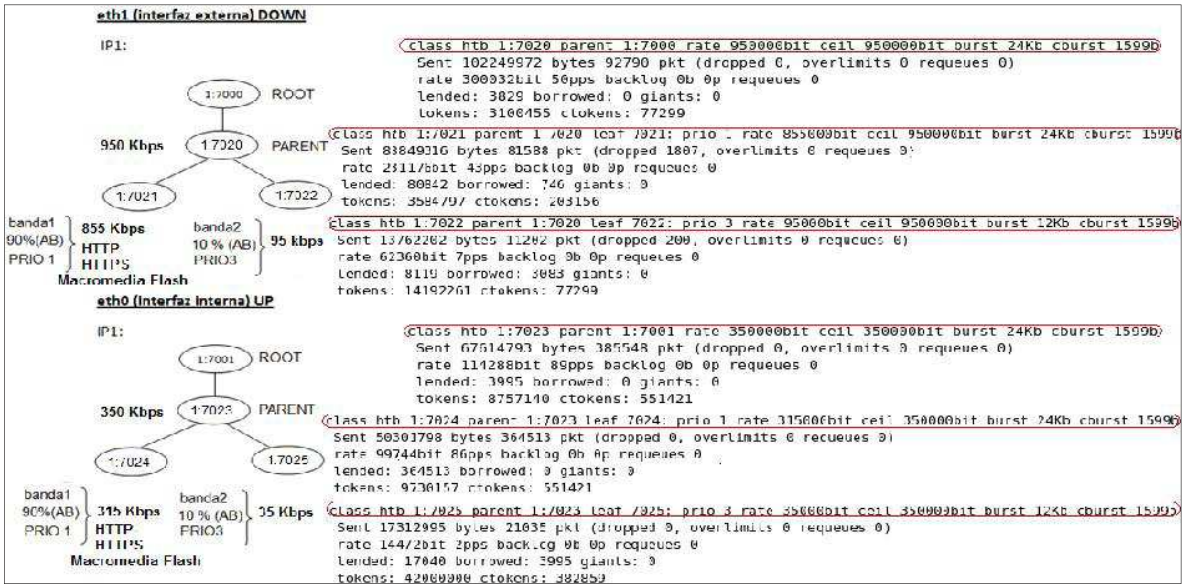



Figura 4. 80: Árbol de clases de la interfaz para el cliente Paola Doicela

### 4.8.3 REINICIOS PROGRAMADOS

Con el fin de comprobar la asignación del día en que se debe llevar a cabo los reinicios, se especifica un ejemplo para los días viernes, configurando la máscara pcimTPCDayOfWeekMask, tal como se muestra en la Figura 4.81.

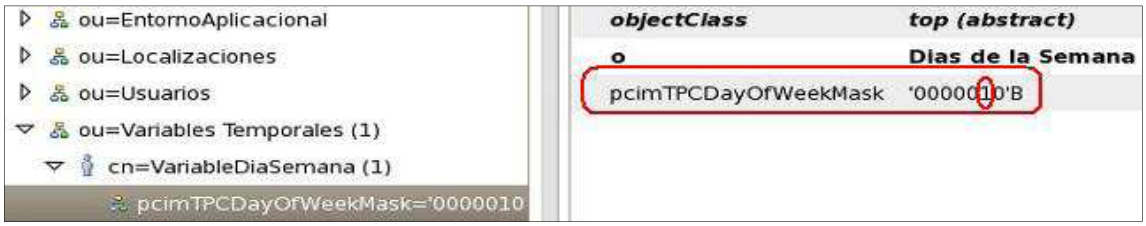



Figura 4. 81: Máscara de reinicios programados

Por lo tanto, El primer reinicio se lleva a cabo el día viernes 28 de diciembre del 2012, a las 4h00 según se especificó en el diseño. En la Figura 4.82 se verifica que el archivo /etc/crontab se actualizó correctamente en Policy Server.

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# For details see man 4 crontabs

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed
#*/1 * * * * root  /bin/echo "hola"
*/50  * *  * * root  /etc/copia
*/55  * *  * * root  /etc/ejecutor/crontabReinicio
0    4 *  * 5 root  /etc/ejecutor/reinicioScript
```

Figura 4. 82: Archivo /etc/crontab actualizado

Como se puede verificar la actualización del día para el reinicio de las antenas se ha realizado automáticamente. Con ello, se demuestra que existe sincronía entre el repositorio de información y el archivo crontab del Policy Server.

A las 8h00 del día viernes 28 de diciembre del 2012 se verifica si realmente se ha ejecutado el *script* para los reinicios de las antenas revisando el archivo */var/log/cron*, tal como se muestra en la Figura 4.83.

En la Figura 4.84, se indica el valor *Uptime* (tiempo) en que la antena (con IP 172.16.33.8) perteneciente al backbone ha permanecido encendida,

```
Dec 28 04:00:01 localhost CROND[12522]: (root) CMD ( /etc/ejecutor/crontabReinicio)
Dec 28 04:00:01 localhost CROND[12523]: (root) CMD (/usr/lib/sa/sa1 1 1)
Dec 28 04:00:01 localhost CROND[12524]: (root) CMD ( /etc/copia^I    )
Dec 28 04:00:01 localhost CROND[12525]: (root) CMD ( /etc/ejecutor/reinicioScript)
```

Figura 4. 83: Ejecución del *script* reinicio en Policy Server

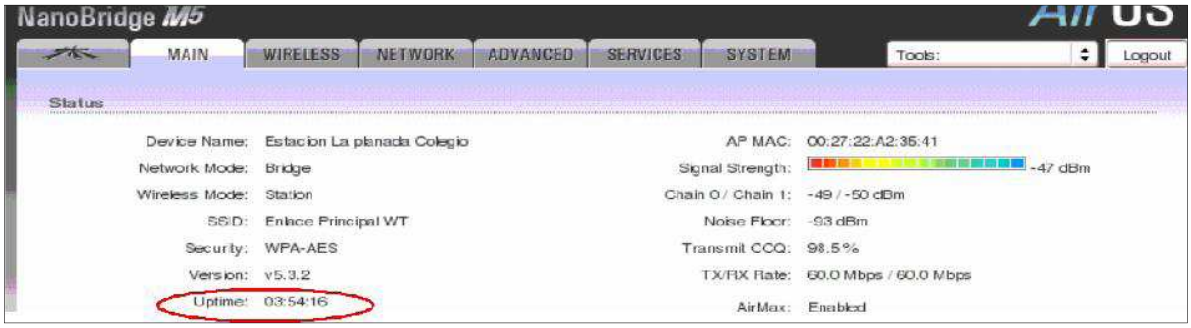



Figura 4. 84: Reinicio del AP 172.16.33.8

Como se puede verificar en la Figura 4.84, el AP tiene 4 horas de operatividad, lo que confirma que se ha reiniciado en la hora indicada.

# CAPÍTULO 5

# CONCLUSIONES Y RECOMENDACIONES

## 5.1 CONCLUSIONES

a) Gestión de red y usuarios

- ✓ La gestión de redes basada en políticas adoptada con el RFC 3198 permite usar directorios como repositorios de información para permitir la administración de entidades de red de acuerdo a los requerimientos de una empresa. Para Telydata se suplió la necesidad de almacenar algunas variables como puertos y direcciones IP con el fin de gestionar el tráfico perteneciente a sus enlaces inalámbricos.

- ✓ La flexibilidad que ofrece LDAP para la gestión de una red consiste en la edición y manipulación de esquemas que ayudan a manejar datos indispensables, eso de demostró con la creación del archivo *parameters.schema* adoptada a las necesidades del Departamento Técnico del WISP Telydata Cía. Ltda. En dicho esquema se definió la estructura de la clase *Airmax Parameters* de los dispositivos pertenecientes a la plataforma inalámbrica Ubiquiti Networks.

- ✓ El almacenamiento de los usuarios del WISP Telydata en un directorio ayudó en organización, disponibilidad y consistencia de su información de tal manera que facilita el trabajo al momento de realizar soporte técnico. Además, en el almacenamiento se han añadido tipos de datos técnicos como direcciones MAC, niveles de señal inalámbrica, CCQ (calidad del enlace) y frecuencia que la base de datos usada (CRM customer Relationship Management) de la empresa no los tenía.

- ✓ Un aspecto importante para la realización del soporte técnico y operación sobre la Red Inalámbrica de Telydata es tener la topología debidamente actualizada con los datos de direcciones IP de los Puntos de Acceso y Estaciones de cada enlace, canales de frecuencia y modulaciones digitales utilizadas. En caso de existir alguna incidencia referente a la infraestructura, una topología debidamente estructurada ayudará como una referencia para recuperación frente a fallos.

- ✓ El servidor OpenLDAP posee muchas ventajas en cuanto a la administración, ya que no solo se puede modificar el árbol DIT desde interfaces graficas LDAP como Apache LDAP Browser, sino que también existen comandos que permiten añadir, eliminar, modificar e incluso respaldar datos vía terminal de Linux.

- ✓ La herramienta de exploración Softerra LDAP Browser posee el componente denominado *LDAP SQL Query*, válido para realizar búsquedas avanzadas de clientes por su nombre, dirección IP o ubicación física, fijando sus resultados en una vista simple o una vista HTML para mayor comodidad del administrador de la Red. Esto ayuda que la búsqueda de usuarios se más eficiente.

- ✓ Una de las ventajas de haber utilizado un servidor OpenLDAP es que cumple con todos los requerimientos de la gestión planteada en el Capítulo 3, sin recurrir a procesamiento excesivo, costos por adquisición de software, ni a complejidad en el entorno de red.

b) Control de la capacidad de acceso de clientes en la RIT

- ✓ Realizar una especificación del nivel de servicio (SLS) respecto a la capacidad de acceso al canal resulta un proceso óptimo en la gestión de red en un proveedor de servicio de Internet, ya que permite organizar valores de

velocidades mínimas y máximas de los distintos clientes de acuerdo a lo contratado.

- ✓ Para lograr asignar a cada cliente un valor de velocidad independiente otros clientes fue primordial adoptar el algoritmo de control de tráfico HTB (*Hierarchical Tocken Bucket*) el cual permitió asignar a cada cliente una clase padre y estructura de clases jerárquica para el manejo del tráfico.

- ✓ Fue importante utilizar disciplinas de colas con clase sobre las interfaces de red de Policy Server, ya que así se administra la manera en cómo se transmiten datos, organizando y mejorando el uso de la capacidad del canal en base a los requerimientos planteados en el capítulo 2.

## 5.2 RECOMENDACIONES

- ✓ Se recomienda utilizar modelos de información como PCIM (*Policy Core Information Model*) y PCELS (*Policy Core Extension LDAP Schema*) para la gestión de usuarios, representación de entidades de red, control de recursos de red y aplicaciones. Estos modelos ayudan a organizar los datos de acuerdo a los requerimientos del administrador de Red y tienen una compatibilidad completa con el protocolo LDAP.

- ✓ Es recomendable que se realice cada semana un barrido de frecuencias para cada enlace de la RIT utilizando herramientas de Ubiquiti Networks, como por ejemplo la herramienta denominada *Site Survey,* para lograr que el enlace se encuentre libre de interferencias y en la máxima calidad, logrando así que la gestión de la capacidad del canal diseñado se ejecute sin problemas.

- ✓ Se recomienda realizar respaldos y copias de seguridad de la base de datos de OpenLDAP semanalmente usando la herramienta *slapcat* y con el servicio ldap detenido. Se puede realizar también con el servicio ldap encendido pero conlleva un proceso muy minucioso y al final se podrían encontrarse inconsistencias al momento de la restauración.

- ✓ Es importante asegurarse que las cadenas que permiten el marcado de paquetes se añadan sin reemplazar a las cadenas del componente NAT en la tabla del kernel Linux, esto evita que los paquetes provenientes de los usuarios no se pierdan en el servidor Linux y salgan al Internet.

- ✓ Se recomienda realizar una gestión de redes basada en políticas para empresas estatales, en donde se permita habilitar, controlar y priorizar las aplicaciones por departamentos, sean estos de contabilidad, de marketing, de talento humano durante determinadas horas del día.

- ✓ Se recomienda integrar la arquitectura Ubiquiti Networks con Linux, ya que permite la administración de una infraestructura completa no solamente vía interfaz gráfica, sino también vía interfaz de comandos (CLI). Puntualmente, los dispositivos soportan comandos básicos de ruteo, control de interfaces LAN, reinicio, apagado y verificación de parámetros de configuración.

- ✓ Es recomendable que para evitar que el kernel de Linux colapse al momento de restaurar las tablas generales del sistema, se definan las sintaxis de las reglas lo más cortas y precisas posibles, específicamente al momento de manipular reglas con la tabla mangle. Esto sucedió varias veces al probar el *script controlador*

# REFERENCIAS BIBLIOGRÁFICAS

## INTERNET

**[1]** ANÓNIMO. “Gestión deRed”
http://servi3.com/asesoria_consultoria_informatica.php
[Leído el 10 Febrero de 2012]

**[2]** HEWLETT-PACKARD COMPANY. “A primer on Policy-based Network Management”
http://pdf.aminer.org/000/291/591/an_analysis_of_policy_provisioning_complex ity_in_accordance_ with_the.pdf
[Leído el 23 Febrero de 2012]

**[3]** LYMBEROPOLOUS,L. “An Adaptative Policy Based Framework For Network Management”
http://pubs.doc.ic.ac.uk/adaptive-policy-network-managem/adaptive-policy-network-managem.pdf
[Leído el 3 de Marzo de 2012]

**[8]** ANÓNIMO. “Classful Queuing Disciplines (qdisc)”
http://linux-ip.net/articles/Traffic-Control-HOWTO/classful-qdiscs.html
[Leído el 2 de Abril de 2012]

**[10]** ANÓNIMO. “Mapping the Policy Core Information Model to a Directory”
http://www.docstoc.com/docs/100266698/Mapping-the-Policy-Core-Information-Model-to-a-Directory#
[Leído el 21 de Abril de 2012]

**[12]** REYES A, BARBA A. “Policy Core Extension Lightweight Directory Access Protocol Schema”
http://www.ietf.org/rfc/rfc4104.txt
[Leído el 22 de Abril de 2012]

**[14]** ANÓNIMO. “Simple Network Management Protocol”
http://www.asante.com/downloads/productdocuments/snmp.pdf
[Leído el 29 de Abril de 2012]

**[15]** BARBA, M. “Gestión de Red”
http://www4.ujaen.es/~mdmolina/grr/Tema%202.pdf
[Leído el 29 de Abril de 2012]

**[17]** PADILLA, D. “Visión Telydata Telecomunicaciones y Datos”
http://www.telydata.net/index.php/compania/mision-vision
[Leído 2 de Mayo de 2012]

**[18]** ANONIMO. “Technical Overview of MIMO”
http://www.home.agilent.com/agilent/editorial.jspx?cc=EC&lc=eng&ckey=1179977&nid=-34832.0.00&id=1179977
[Leído el 10 de Mayo de 2012]

**[19]** UBIQUITI NETWORKS. “Productos Airmax”
http://forum.ubnt.com/showthread.php?t=46005
[Leído el 10 de Mayo de 2012]

**[20]** UBIQUITI NETWORKS. “Airmax Hardware”
http://www.ubnt.com/airmax#airMaxHardware
[Leído el 10 de Mayo de 2012]

**[21]** UBIQUITI NETWORKS. “Manual de Usuario”
http://wiki.ubnt.com/AirOS_5_Spanish#AirOS_v5.0_Gu.C3.ADa_de_configuraci.C3.B3n
[Leído el 12 de Mayo de 2012]

**[22]** UBIQUITI NETWORS. “Manual de Usuario v 5.0”
http://wiki.ubnt.com/AirOS_5_Spanish
[Leído el 12 de Mayo de 2012]

**[25]** CACTI. “Documentación and howtos”
http://docs.cacti.net/
[Leído el 25 de Mayo de 2012]

**[26]** CACTI. “The Cacti Manual”
http:www.cacti.net/downloads/docs/html/install_unix.html
[Leído el 25 Mayo de 2012]

**[27]** ANÓNIMO. “Protocolo HTTP”
http://www.uhu.es/josel_alvarez/NvasTecnProg/recursos/ProtocoloHTTP.pdf
[Leído el 12 de Junio de 2012]

**[28]** ANÓNIMO. “HTTP over SSL”
http://www.rtfm.com/sslbook/chap9-sample.pdf
[Leído el 12 de Junio de 2012]

**[29]** ANÓNIMO. “Streaming Video with Flash Media Server”
http://www.mediacollege.com/adobe/flash/streaming/media-server.html
[Leído el 12 de Junio de 2012]

**[30]** ANÓNIMO. “Information About oemcacao-websvc Port 11175/tcp”
http://www.corrupteddatarecovery.com/Port/11175tcp-Port-Type-oemcacao-websvc-oemcacao-websvc.asp
[Leído el 20 de Junio de 2012]

**[31]** BORONAT, F. “Especificación y evaluación de un algoritmo de sincronización de grupo de flujos multimedia”
http://riunet.upv.es/bitstream/handle/10251/4781/tesisUPV1924.pdf
[Leído el 20 de Junio de 2012]

**[32]** ALOMO, S. “La mensajería instantánea, el protocolo messenger (msnp)”
http://tavmsn.sourceforge.net/pdf/presentacion%20el%20protocolo%20 messenger.pdf
[Leído el 20 de Junio de 2012]

**[33]** AKAMAI TECHNOLOGIES. “Customer Stories”
http://www.akamai.com/html/customers/index.html
[Leído el 20 de Julio de 2012]

**[34]** ANÓNIMO. “Ecuador Telecom S.A.”
http://es.wikipedia.org/wiki/EcuadorTelecom_S.A.
[Leído el 22 de Julio de 2012]

**[35]** ANÓNIMO. “What is 1e100.net ”
http://support.google.com/bin/answer.py?hl=en&answer=174717
[Leído el 22 de Julio de 2012]

**[36]** ANÓNIMO. “Facebook”
http://es.wikipedia.org/wiki/Facebook
[Leído el 22 de Julio de 2012]

**[37]** SOFTERRA, INC. “LDAP Browser, Schema Viewer”
http://www.ldapadministrator.com/resources/english/help/la20121/ch13s02.html
[Leído el 8 de Agosto de 2012]

**[38]** SCIBERRAS, A. “Lightweight Directory Access Protocol (LDAP): Schema for user Applications”
http://www.rfc-editor.org/rfc/rfc4519.txt
[Leído 9 de Agosto de 2012]

**[39]** SMITH, M. “Definition of the inetOrgPerson LDAP Object Class”
http://www.ietf.org/rfc/rfc2798.txt
[Leído 3 de Septiembre de 2012]

**[40]** HOWARD, L. “An Approach for Using LDAP as a Network Information Service”
http://www.ietf.org/rfc/rfc2307.txt
[Leído 3 de Septiembre de 2012]

**[41]** WAHL, M. “A Summary of the X.500. User Schema for use with LDAPv3”
http://www.ietf.org/rfc/rfc2256.txt
[Leído 3 de Septiembre de 2012]

**[42]** BARBA, A. BRUNNER, M. “Policy Core Extension Lightweight Directory Access Protocol Schema (PCELS)”
http://www.ietf.org/rfc/rfc4104.txt
[Leído 10 de Septiembre de 2012]

**[43]** STRASSNER, J. MOORE,B. “Policy Core Lightweight Directory Access Protocol (LDAP) Schema”
http://www.ietf.org/rfc/rfc3703.txt
[Leído el 12 de Septiembre de 2012]

**[44]** DEVERA, M. “HTB Linux queuing discipline manual – user guide”
http://luxik.cdi.cz/~devik/qos/htb/manual/userg.htm
[Leído 12 de Septiembre de 2012]

**[45]** SUAREZ, J. “Curso OpenLDAP”
http://www.redes-linux.com/manuales/openldap/curso_openldap.pdf
[Leído 14 de Septiembre de 2012]

**[46]** ANÓNIMO. “Lotus Domino”
http://www-10.lotus.com/ldd/dominowiki.nsf/xpViewCategories.xsp?lookupName=Lotus%20Domino
[Leído el 14 de Septiembre de 2012]

**[47]** ANÓNIMO. “Lotus Documentation”
http://www.ibm.com/developerworks/lotus/documentation/
[Leído el 14 de Septiembre de 2012]

**[48]** APACHE DIRECTORY. “Apache Directory Project”
http://directory.apache.org/
[Leído el 14 de Septiembre de 2012]

**[49]** ANÓNIMO. “Fedora Directory Server”
http://es.scribd.com/doc/12935981/Fedora-Directory-Server-FDS
[Leído el 20 de Septiembre de 2012]

**[50]** WINDOWS DEVELOPER CENTER. “So What is Active Directory”
http://msdn.microsoft.com/en-us/library/windows/desktop/aa746492(v=vs.85).aspx
[Leído el 20 de Septiembre de 2012]

**[51]** MICROSOFT TECHNET. “Guía detallada de administración de Active Directory”
http://www.microsoft.com/latam/technet/productos/windows/windowsserver2003/admng.mspx
[Leído el 20 de Septiembre de 2012]

**[52]** OPENLDAP PROJECT. “OpenLDAP Software 2.4 Administrator's Guide”
http://www.openldap.org/doc/admin24/
[Leído el 20 de Septiembre de 2012]

**[54]** ANÓNIMO. “JXplorer”
http://www.ecured.cu/index.php/JXplorer
[Leído el 20 de Septiembre de 2012]

**[55]** SOFTERRA, INC. “Softerra LDAP Administrator Features”
http://www.ldapbrowser.com/features_directory-browsing.htm
[Leído el 20 de Septiembre de 2012]

**[56]** APACHE DIRECTORY STUDIO. “The Eclipse based LDAP browser and directory client”
http://directory.apache.org/studio/
[Leído el 21 de Septiembre de 2012]

**[57]** ANÓNIMO. “Distribuciones de Linux”
http://www.linux-es.org/distribuciones
[Leído el 21 de Septiembre de 2012]

**[58]** ANÓNIMO. “Distribuciones principales de Linux”
http://linuxzone.es/distribuciones-principales/
[Leído el 22 de Septiembre de 2012]

**[60]** RedHat Enterprise Linux. “E.2.25. /proc/stat”
https://access.redhat.com/knowledge/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s2-proc-stat.html
[Leído el 23 de Septiembre de 2012]

**[61]** ANÓNIMO. “Obtener uso total del CPU”
http://www.espaciolinux.com/foros/hardware/como-obter-consumo-total-cpu-t45346.html
[Leído el 3 de Octubre de 2012]

**[62]** ANÓNIMO. “Calculo del uso del CPU”
http://quierolinux.blogspot.com/2008/10/clculo-del-uso-de-cpu-en-linux.html
[Leído el 3 de Octubre del 2012]

**[63]** OPENLDAP PROJECT. “Tunning”
http://www.openldap.org/doc/admin24/tuning.html#Memory
[Leído el 3 de Octubre de 2012]

**[64]** OPENLDAP PROJECT. “RAM SIZE VS. DB CACHE SIZE”
http://www.openldap.org/lists/openldap-software/200210/msg00477.html
[Leído el 3 de Octubre de 2012]

**[65]** ORACLE “Java SE Downloads”
http://www.oracle.com/technetwork/java/javase/downloads/index.html
[Leído el 7 de Noviembre de 2012]

**[66]** CentOS. Community ENTerprise Operating System. “CentOS Product Specifications”
http://wiki.centos.org/About/Product#fndef- 281c87f720ad19ead74bbcb7000 79336ab0643c5-8
[Leído el 7 de Noviembre de 2012]

## TESIS

**[4]** FLORES,M. “Implementación de un prototipo de una aplicación web para controlar el ancho de banda y brindar calidad de servicio por puertos para televigilancia basado en el sistema operativo Linux”. EPN. Quito, Ecuador. 2011

**[9]** TEXEIRA, G., CORDENA- Uma Plataforma para Gestão de Redes Baseada em Políticas. Arquitecturas e mecanismos de tradução de políticas. Universidad do Minho. Guimarães, Portugal. 2006.

**[53]** ROMERO, C. “Análisis comparativo entre productos que proveen servicio de directorio pertenecientes a tecnologías propietaria y de libre acceso, aplicado a laboratorios en ambientes educativos. Escuela Politécnica del Litoral. Guayaquil, Ecuador. 2008.

**[59]** JIMENEZ, G; PAZMIÑO, C. “Análisis, Implementación y evaluación de un prototipo router dual IPv4/IPv6 con soporte de QoS e IPSec, sobre Linux, usando AHP para la selección del hardware e IEEE830 para la selección del software”. EPN. Quito, Ecuador. 2010

## LIBROS Y DOCUMENTACIÓN TÉCNICA

**[5]** CUELLAR QUIÑONEZ, Juan. “Sheduling Algorithms in packet networks”. Universidad ISECI. Cali, Colombia. Agosto, 2009.

**[6]** Chuck Semeria. (2001, Diciembre). Supporting Differenciated Service Classes: QueueScheduling Disciplines [Online]. Disponible en: http://www.terabitsystems.com/juniper-docs/Supporting%20Differentiated%20Service%20Classes%20-%20Active%20Queue%20Memory%20Management.pdf

**[7]** Jussi Lemponen. (2001, Abril). Implementation of Differentiated Services Policy Information Based on Linux [Online]. Disponible en: http://www.atm.tut.fi/faster/qbone/linux-pep.pdf

**[11]** CARTER, Gerald. “LDAP System Administration”. 1era Edición. Editorial O'Reilly & Associates. California, Estados Unidos. Marzo, 2004.

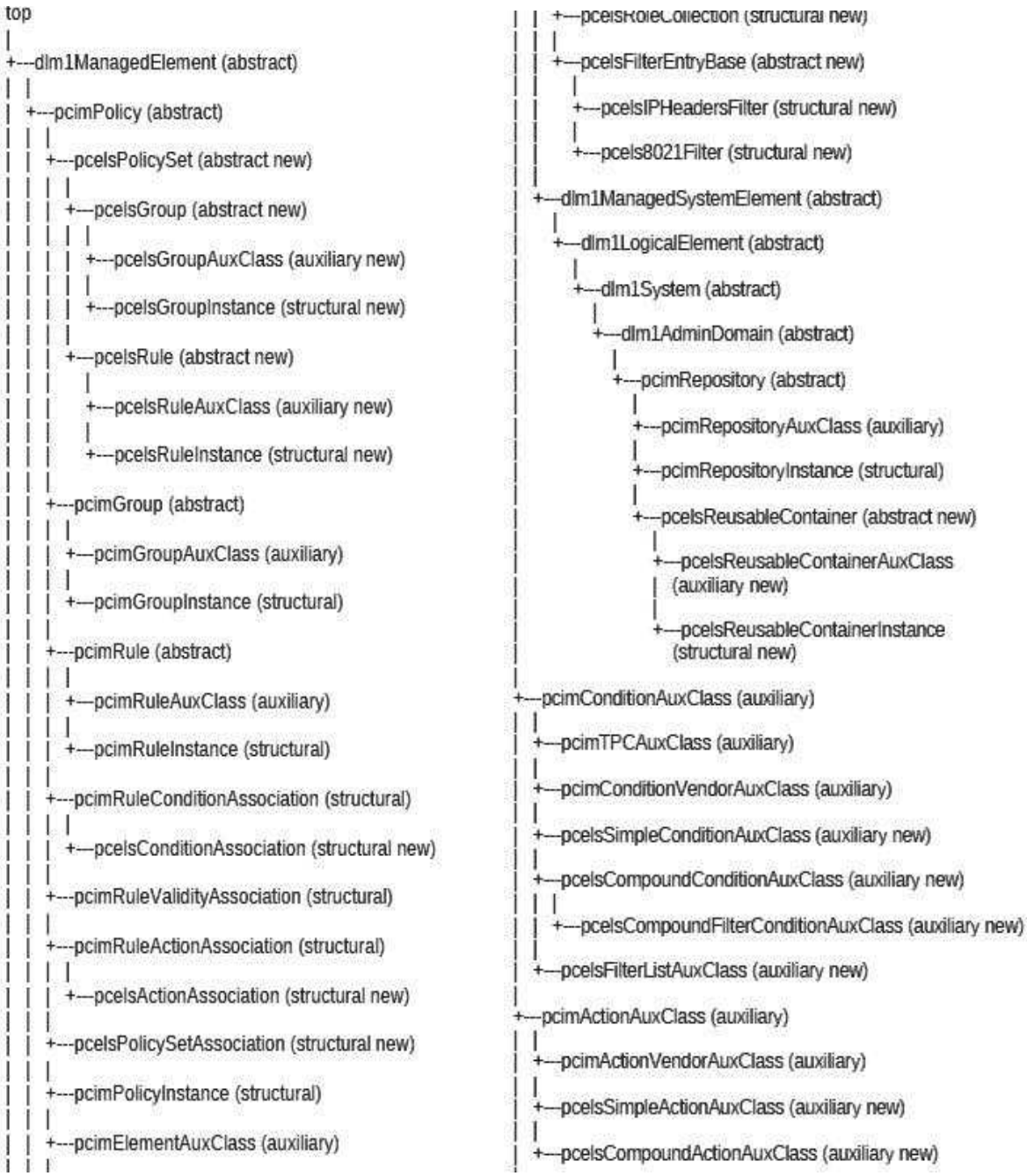
**[23]** SOLARWINDS. (2012, Febrero). SOLARWINDS Network Configuration Manager Administration Guide [Online]. Disponible en: http://www.solarwinds.com/documentation/orionNCM/docs/orionNCM AdministratorGuide.pdf

**[24]** CISCO SYSTEMS. (2009, Mayo). CISCO IOS NETFLOW CONFIGURATION GUIDE, CISCO IOS Release 12.2SX[Online]. Disponible en: http://www.cisco.com/en/US/docs/ios-xml/ios/netflow/configuration/12-2sx/nf-12-2sx-book.pdf

# ANEXOS

# ANEXO 1

# JERARQUÍA DE CLASES PCELS


top
dlm1ManagedElement (abstract)
pcimPolicy (abstract)
pcelsPolicySet (abstract new)
pcelsGroup (abstract new)
pcelsGroupAuxClass (auxiliary new)
pcelsGroupInstance (structural new)
pcelsRule (abstract new)
pcelsRuleAuxClass (auxiliary new)
pcelsRuleInstance (structural new)
pcimGroup (abstract)
pcimGroupAuxClass (auxiliary)
pcimGroupInstance (structural)
pcimRule (abstract)
pcimRuleAuxClass (auxiliary)
pcimRuleInstance (structural)
pcimRuleConditionAssociation (structural)
pcelsConditionAssociation (structural new)
pcimRuleValidityAssociation (structural)
pcimRuleActionAssociation (structural)
pcelsActionAssociation (structural new)
pcelsPolicySetAssociation (structural new)
pcimPolicyInstance (structural)
pcimElementAuxClass (auxiliary)
pcelsRoleCollection (structural new)
pcelsFilterEntryBase (abstract new)
pcelsIPHeadersFilter (structural new)
pcels8021Filter (structural new)
dlm1ManagedSystemElement (abstract)
dlm1LogicalElement (abstract)
dlm1System (abstract)
dlm1AdminDomain (abstract)
pcimRepository (abstract)
pcimRepositoryAuxClass (auxiliary)
pcimRepositoryInstance (structural)
pcelsReusableContainer (abstract new)
pcelsReusableContainerAuxClass (auxiliary new)
pcelsReusableContainerInstance (structural new)
pcimConditionAuxClass (auxiliary)
pcimTPCAuxClass (auxiliary)
pcimConditionVendorAuxClass (auxiliary)
pcelsSimpleConditionAuxClass (auxiliary new)
pcelsCompoundConditionAuxClass (auxiliary new)
pcelsCompoundFilterConditionAuxClass (auxiliary new)
pcelsFilterListAuxClass (auxiliary new)
pcimActionAuxClass (auxiliary)
pcimActionVendorAuxClass (auxiliary)
pcelsSimpleActionAuxClass (auxiliary new)
pcelsCompoundActionAuxClass (auxiliary new)

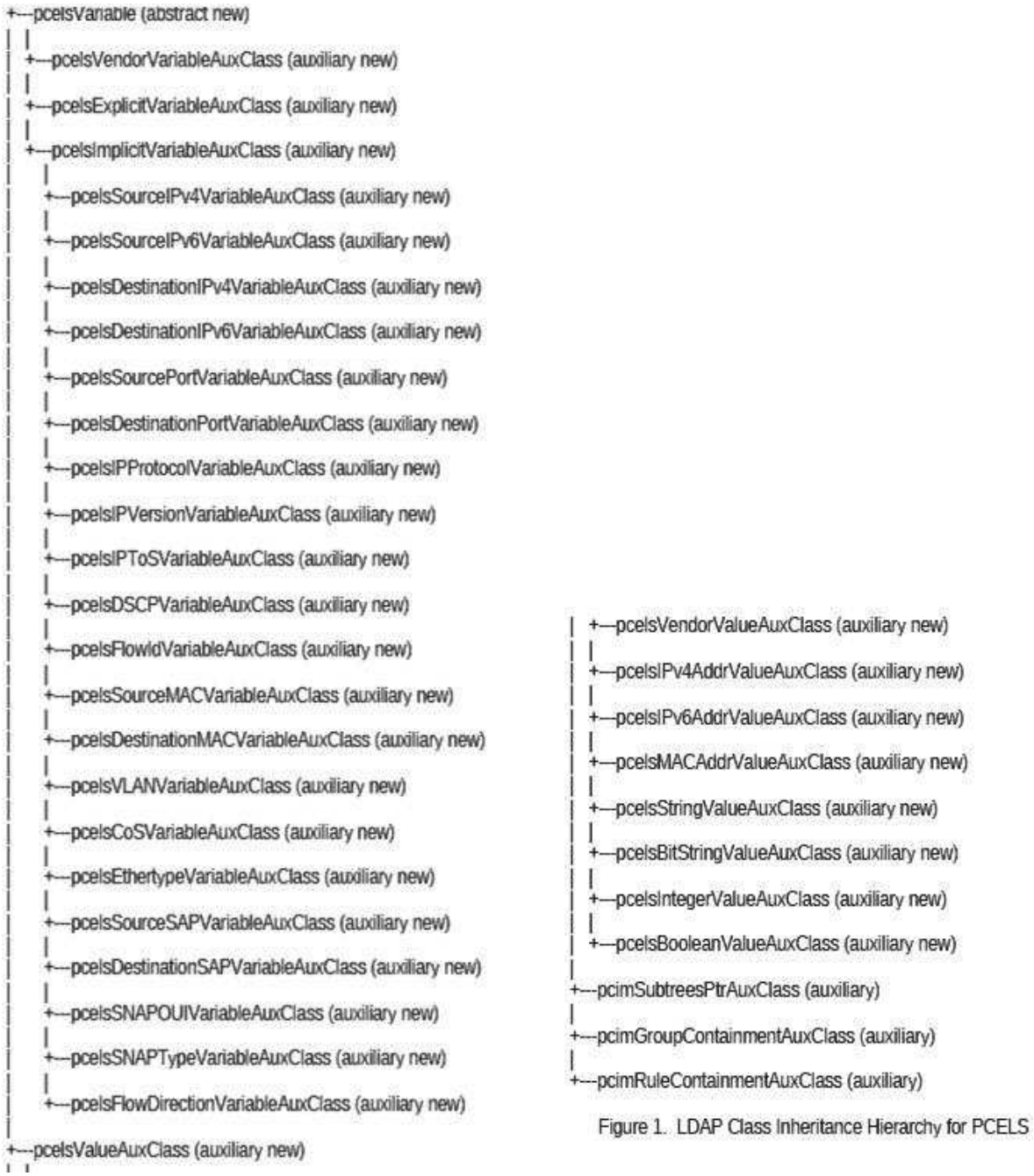



Figure 1. LDAP Class Inheritance Hierarchy for PCELS

# ANEXO 2

# TABLA DE VELOCIDADES ESTABLECIDA POR UBIQUITI NETWORK

**DATA RATES FOR MODULATION & CODING SCHEMES**
**FOR MULTIPLE CHANNEL WIDTHS – Mbps 12Jul10**

**IEEE 802.11n Ubiquiti AirOS 'M' series radios**

Rates are theoretical; not guaranteed

| MCS | MOD | Spatial Streams | 40Mhz channel | 20Mhz channel | 10MHz channel | 5Mhz channel |
|---|---|---|---|---|---|---|
| | | | 400ns GI | 800ns GI | 800ns GI | 800ns GI |
| 0 | BPSK | 1X1 | 15 | 6.5 | 3.25 | 1.625 |
| 1 | QPSK | 1X1 | 30 | 13 | 6.5 | 3.25 |
| 2 | QPSK | 1X1 | 45 | 19.5 | 9.75 | 4.875 |
| 3 | 16-QAM | 1X1 | 60 | 26 | 13 | 6.5 |
| 4 | 16-QAM | 1X1 | 90 | 39 | 19.5 | 9.75 |
| 5 | 64-QAM | 1X1 | 120 | 52 | 26 | 13 |
| 6 | 64-QAM | 1X1 | 135 | 58.5 | 29.25 | 14.625 |
| 7 | 64-QAM | 1X1 | 150 | 65 | 32.5 | 16.25 |
| 8 | BPSK | 2X2 | 30 | 13 | 6.5 | 3.25 |
| 9 | QPSK | 2X2 | 60 | 26 | 13 | 6.5 |
| 10 | QPSK | 2X2 | 90 | 39 | 19.5 | 9.75 |
| 11 | 16-QAM | 2X2 | 120 | 52 | 26 | 13 |
| 12 | 16-QAM | 2X2 | 180 | 78 | 39 | 19.5 |
| 13 | 64-QAM | 2X2 | 240 | 104 | 52 | 26 |
| 14 | 64-QAM | 2X2 | 270 | 117 | 58.5 | 29.25 |
| 15 | 64-QAM | 2X2 | 300 | 130 | 65 | 32.5 |

**MCS:** Índice usado por Ubiquiti Networks que determina el número de Flujos espaciales, la modulación, la velocidad de codificación y los valores de la velocidad de datos

**GI (Intervalo de Guardia):** Índice usado por Ubiquiti Networks para determinar un intervalo de guardia entre símbolos que ayuda a los receptores a superar los efectos de retardo por trayectoria múltiple. Ayuda también a contrarrestar el efecto por ISI (Interferencia Inter Símbolo).

# ANEXO 3

# TABLA Y CÁLCULO DEL TRÁFICO PROMEDIADO DIARIO POR CATEGORÍA

a) Tabla del Tráfico Diario. Semana del 16 de Abril del 2012 al 22 de Abril del 2012

| | 07/05/2012 | | | 08/05/2012 | | | 09/05/2012 | | | 10/05/2012 | | | 11/05/2012 | | | 12/05/2012 | | | 13/05/2012 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tráfico | TX(GB) | RX(GB) | Tráfico (Gbps) | TX(GB) | RX(GB) | Tráfico (Gbps) | TX(GB) | RX(GB) | Tráfico (Gbps) | TX(GB) | RX(GB) | Tráfico (Gbps) | TX(GB) | RX(GB) | Tráfico (Gbps) | TX(GB) | RX(GB) | Tráfico (Gbps) | TX(GB) | RX(GB) | Tráfico (Gbps) |
| HTTP | 31,90 | 119,90 | 0,014056 | 34,300 | 129,900 | 0,015204 | 38,900 | 150,200 | 0,017509 | 39,500 | 151,500 | 0,017685 | 40,000 | 160,000 | 0,018519 | 40,900 | 157,700 | 0,018389 | 36,500 | 141,500 | 0,016481 |
| HTTPS | 3,700 | 7,800 | 0,001065 | 3,400 | 9,300 | 0,001176 | 3,100 | 10,200 | 0,001231 | 3,100 | 9,600 | 0,001176 | 3,900 | 9,900 | 0,001278 | 3,900 | 9,500 | 0,001241 | 2,500 | 8,200 | 0,000991 |
| Macromedia Flash | 0,314 | 0,809 | 0,000104 | 0,049 | 0,781 | 0,000077 | 0,234 | 0,515 | 0,000069 | 0,856 | 0,542 | 0,000129 | 0,828 | 0,386 | 0,000112 | 0,432 | 0,351 | 0,000073 | 0,247 | 0,338 | 0,000054 |
| RTP | 0,224 | 0,232 | 0,000042 | 0,311 | 0,234 | 0,000050 | 0,374 | 0,422 | 0,000074 | 0,334 | 0,518 | 0,000079 | 0,167 | 0,231 | 0,000037 | 0,488 | 0,378 | 0,000080 | 0,478 | 0,475 | 0,000088 |
| MSNP | 0,026 | 0,041 | 0,000006 | 0,206 | 0,152 | 0,000033 | 0,256 | 0,186 | 0,000041 | 0,424 | 0,464 | 0,000082 | 0,447 | 0,553 | 0,000093 | 0,316 | 0,298 | 0,000057 | 0,847 | 0,627 | 0,000136 |
| Oem Web Service | 0,000 | 0,658 | 0,000061 | 0,000 | 0,663 | 0,000061 | 0,000 | 0,713 | 0,000066 | 0,000 | 0,710 | 0,000066 | 0,000 | 0,708 | 0,000066 | 0,000 | 0,706 | 0,000065 | 0,000 | 0,610 | 0,000056 |
| | | Suma | 0,015334 | | Suma | 0,016601 | | Suma | 0,018991 | | Suma | 0,019217 | | Suma | 0,020104 | | Suma | 0,019905 | | Suma | 0,017808 |

b) Tabla del Tráfico Diario. Semana del 23 de Abril del 2012 al 29 de Abril del 2012

| | 16/04/2012 | | | 17/04/2012 | | | 18/04/2012 | | | 19/04/2012 | | | 20/04/2012 | | | 21/04/2012 | | | 22/04/2012 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tráfico | TX (GB) | RX (GB) | Trafico Gbps | TX (GB) | RX (GB) | Trafico Gbps | TX (GB) | RX (GB) | Trafico Gbps | TX (GB) | RX (GB) | Trafico Gbps | TX (GB) | RX (GB) | Trafico Gbps | TX (GB) | RX (GB) | Trafico Gbps | TX (GB) | RX (GB) | Trafico Gbps |
| HTTP | 30,15 | 121,90 | 0,01407 | 33,260 | 128,80 | 0,015006 | 34,690 | 149,47 | 0,0170 | 38,65 | 153,30 | 0,01777 | 39,180 | 159,30 | 0,01837 | 42,000 | 158,1 | 0,01852 | 37,900 | 141,84 | 0,01664 |
| HTTPS | 3,100 | 6,200 | 0,00086 | 3,610 | 9,100 | 0,001177 | 2,960 | 9,960 | 0,00119 | 3,000 | 9,340 | 0,00114 | 3,170 | 9,750 | 0,001196 | 3,520 | 9,980 | 0,001250 | 2,567 | 8,232 | 0,00100 |
| Macromedia Flash | 0,205 | 0,706 | 0,00008 | 0,045 | 0,751 | 0,000074 | 0,306 | 0,491 | 0,00007 | 0,880 | 0,589 | 0,00013 | 0,759 | 0,344 | 0,00010 | 0,405 | 0,318 | 0,00006 | 0,289 | 0,381 | 0,00006 |
| RTP | 0,252 | 0,256 | 0,00004 | 0,262 | 0,236 | 0,000046 | 0,310 | 0,445 | 0,00007 | 0,351 | 0,519 | 0,00008 | 0,167 | 0,251 | 0,00003 | 0,451 | 0,313 | 0,00007 | 0,403 | 0,755 | 0,00010 |
| MSNP | 0,019 | 0,043 | 0,00000 | 0,179 | 0,112 | 0,000027 | 0,276 | 0,255 | 0,00004 | 0,398 | 0,401 | 0,00007 | 0,465 | 0,556 | 0,00009 | 0,258 | 0,278 | 0,00005 | 0,828 | 0,698 | 0,00014 |
| Oem Web Service | 0,000 | 0,561 | 0,00005 | 0,000 | 0,679 | 0,000063 | 0,000 | 0,721 | 0,00006 | 0,000 | 0,723 | 0,00006 | 0,000 | 0,712 | 0,00006 | 0,000 | 0,766 | 0,00007 | 0,000 | 0,515 | 0,00004 |
| | | Suma | 0,015129 | | Suma | 0,0163 | | Suma | 0,0185 | | Suma | 0,01927 | | Suma | 0,01987 | | Suma | 0,02003 | | Suma | 0,01800 |

a) Cálculo del tráfico promediado diario

1 día = 86400 segundos

$$\text{Tráfico}_{\text{Total}} : \frac{(Tx + Rx)\ GBytes}{1\ dia} * \frac{8\ bits}{1\ byte} * \frac{1\ dia}{86400\ seg.}$$

Ejemplo Tráfico Diario para HTTP. HTTP: $\frac{(31.90+119.90)GBytes}{1\ dia} * \frac{8\ bits}{1\ byte} * \frac{1\ dia}{86400\ seg.} =$

$$= 0.01405\ MBbps \approx 14.05\ Mbps$$

Ejemplo Tráfico Promediado Diario:

$T_{PROM.\ DIARIO.} = (T_{HTTP\ 1} + T_{HTTP\ 2} + \ldots\ldots\ldots + T_{HTTP\ 7)} / 7$

# ANEXO 4

# DIRECCIONES IP MÁS UTILIZADAS EN INTERNET

| Direccion IP | Asignación IANA | | Actividad relacionada Relacionada |
|---|---|---|---|
| | Rango Asignado | Nombre de la Organización | |
| 186.5.104.11<br>186.5.104.10<br>186.5.104.12<br>186.5.104.14<br>186.5.104.13 | 186.5.0.0/17 | Telconet S.A Guayaquil | Caches de Video. Ecuador |
| 74.125.229.40<br>74.125.229.41<br>74.125.45.117<br>74.125.229.134<br>74.125.229.45<br>74.125.229.231<br>74.125.229.<br>172.125.229.228<br>74.125.137.94<br>74.125.137.147<br>74.125.137.100<br>74.125.137.101<br>74.125.137.102<br>74.125.137.106<br>173.194.37.71<br>173.194.37.73<br>173.194.37.67<br>173.194.37.78<br>173.194.37.66<br>172.194.37.39<br>74.124.130.190 | 74.125.0.0 – 74.125.255.255 | GOOGLE | Videos Youtube |
| 194.190.77.178 | 194.190.76.0 - 194.190.77.255 | CJSC RuTube | Videos relacionados. Reproductor de videos on line Ruso. |
| 208.117.252.14 | 208.117.224.0 - 208.117.255.255 | YouTube, Inc. | Youtube, Repoductor de Video on line basado en adobe flash |
| 199.9.252.246 | 199.9.248.0 - 199.9.255.255 | Justin.tv, Inc. | Conjunto de sitios web relacionado al streaming de video (peliculas, canales en vivo,etc.) |
| 173.194.29.20<br>173.194.73.94<br>173.194.60.12 | 173.194.0.0- 173.194.255.255 | GOOGLE Buscador Avanzado de Google | Gestor de busquedas en internet a nivel simple o avanzado |

| | | | |
|---|---|---|---|
| 92.123.78.67<br>72.247.78.64 | 72.246.0.0-<br>72.247.255.255 | Akamai Technologies | Navegacion, Servicios Web |
| 63.130.161.106-161 | 63.130.161.0 - 63.130.161.255 | Akamai Technologies Inc | Navegacion y servicios web (Justin.tv) |
| | 72.246.0.0 - 72.247.255.255 | Akamai Technologies Inc | Navegacion y Servicios Web. |
| 77.67.28.72 | 77.67.28.0 - 77.67.28.127 | Akamai Technologies Inc | Navegacion y servicios Web |
| 69.171.247.16<br>69.171.247.80<br>69.171.247.48<br>69.171.242.70<br>66.220.153.74<br>66.220.158.11<br>69.171.227.50 | 69.171.224.0 - 69.171.255.255 | Facebook, Inc. | Redes Sociales |
| 65.54.50.103 | 65.52.0.0 - 65.55.255.255 | Microsoft Corp | Messenger,msn,chat. Correo electronico. |
| 200.124.254.209<br>200.124.254.208<br>200.124.254.216<br>200.124.254.215<br>200.124.254.214<br>200.124.254.212<br>200.124.254.211 | 200.124.240/20 | Ecuador Telecom S.A. | Ecutel, Claro servidores validos para navegacion web. Direcciones de servidores DNS |
| 201.218.56.213<br>201.218.56.212<br>201.218.56.218<br>201.218.56.219<br>201.218.56.210<br>201.218.56.217<br>201.218.56.214<br>201.218.56.215<br>201.218.56.209<br>201.218.56.216<br>201.218.56.208 | 201.218.32/19 | Telconet S.A. | Direciones de servidores DNS. Direcciones de navegadores web |
| 69.164.38.8<br>69.164.26.88 | 69.164.0.0 - 69.164.63.255 | Limelight Networks, Inc. | Red de distribución de contenido en línea líder de contenido digital multimedia (CDN) |

| | | | |
|---|---|---|---|
| 72.21.81.132 | 72.21.80.0 - 72.21.95.255 | EdgeCast Networks, Inc. | Servicios de Distribucion de Contenidos digitales multimedia a nivel global y de alta calidad |
| 195.39.12.39 | 195.39.12.0 - 195.39.12.255 | AVAST Software a.s. | Distribución del Software Avast |
| 95.211.143.77 | 95.211.142.144 - 95.211.159.255 | LEASEWEB | Relacionados a la distribución de contenidos digitales y descargas. |
| 199.80.55.158 | 199.80.52.0 - 199.80.55.255 | WZ Communications Inc. | Descarga de archivos desde varios almacenadores como Webzilla , 4shared |

# ANEXO 5

# DESGLOSE DE CARACTERÍSTICAS DE LAS DISTRIBUCIONES LINUX

- **Debian**

| Criterio | Detalles | Peso | Evaluación | Total |
|---|---|---|---|---|
| CONFIGURACION | | | | |
| Licencia **(Item1)** | Perteneciente al software libre GNU/GPL, sin embargo debián contiene repositorios no libres que no se incluyen por defecto, sino que hay que descargarlos manualmente | 5 | 3 | 15 |
| Manejo del Sistema basado en Gráfico | No tiene herramientas especiales, puede usar GNOME | 3 | 1 | 3 |
| Manejo del Sistema basado en consola | Posee un paquete robusto de configuración (debconf). Editor vi | 5 | 5 | 25 |
| SISTEMA DE PAQUETES | | | | |
| Cantidad de Paquetes | En este aspecto Debián es líder, ya que posee alrededor de 18 mil paquetes en el repositorio principal. | 4 | 5 | 20 |
| Gestión de Paquetes y Resolución Automática de Dependencias | Posee la herramienta apt, muy poderosa para gestionar paquetes. La instalación es fácil al resolver las dependencias automáticamente | 5 | 5 | 25 |
| EFICIENCIA | | | | |
| Velocidad del Sistema de Arranque **(Item4)** | El sistema de arranque es muy rápido. El uso de update-rc.d, herramienta para eliminar los archivos innecesarios colaboran para que el sistema operativo sea más rápido con un tiempo de 40 a 50 segundos en una instalación por defecto. | 5 | 5 | 25 |
| Velocidad de Respuesta del Sistema **(Item5)** | La velocidad de respuesta es de nivel medo. Existen programas que no han sido optimizados, sin embargo el uso de *script*s ayudan a mejorar las implementaciones de tipo servidor | 5 | 3 | 15 |
| ESTABILIDAD Y DISPONIBILIDAD | | | | |
| Centro de Seguridad **(Item3)** | Los paquetes de seguridad se actualizan periódicamente mediante apt-get. | 5 | 4 | 20 |

| | | | | |
|---|---|---|---|---|
| Estabilidad y Madurez **(Item2)** | Es una de las distribuciones más antiguas, y posee gran desarrollo por la comunidad | 5 | 5 | 25 |
| Documentación **(Item6)** | Existe documentación adecuada como: manuales, procedimientos, documentos y foros | 5 | 5 | 25 |

- **Slackware**

| Criterio | Detalles | Peso | Evaluación | Total |
|---|---|---|---|---|
| CONFIGURACION | | | | |
| Licencia **(Item1)** | Todos los componentes pertenecen a GNU/GPL | 5 | 5 | 25 |
| Manejo del Sistema basado en Grafico | No tiene herramientas graficas aunque puede utilizar GNOME, o KDE | 3 | 1 | 3 |
| Manejo del Sistema basado en consola | A través de editores básicos: vi, emacs. No tiene herramientas adicionales | 5 | 1 | 5 |
| SISTEMA DE PAQUETES | | | | |
| Cantidad de Paquetes | Pequeño número de paquetes oficiales (alrededor de 100 a 1500). Son importantes las utilidades LinuxPackages y slacky.it | 4 | 3 | 12 |
| Gestión de Paquetes y Resolución Automática de Dependencias | Se basa en simples paquetes tgz, sin embargo no contiene toda la información sobre dependencias. | 5 | 1 | 5 |
| EFICIENCIA | | | | |
| Velocidad del Sistema de Arranque **(Item 4)** | Aceptable, pero puede ser optimizada mediante *script*s. Promedio de 1 minuto | 5 | 5 | 25 |
| Velocidad de Respuesta del Sistema **(Item 5)** | Es muy buena debido a que no se habilitan muchos servicios. | 5 | 4 | 20 |
| ESTABILIDAD Y DISPONIBILIDAD | | | | |
| Centro de Seguridad **(Item3)** | De alto nivel, ya que posee herramientas para protección de acceso a nivel de núcleo, memoria y red | 5 | 5 | 25 |

| | | | | |
|---|---|---|---|---|
| Estabilidad y Madurez **(Item2)** | Fiable; casi comparable con debián ya que contiene un grupo selecto de paquetes | 5 | 5 | 25 |
| Documentación **(Item6)** | Difícil para usuarios novatos, las comunidades son escasas. | 5 | 3 | 15 |

- **Centos**

| Criterio | Detalles | Peso | Evaluación | Total |
|---|---|---|---|---|
| CONFIGURACION | | | | |
| Licencia **(Item1)** | Perteneciente a GNU/GPL en su totalidad | 5 | 5 | 25 |
| Manejo del Sistema basado en Grafico | Existen muchas herramientas basadas en GNOME. El sistema de operaciones en su mayoría se lo puede realizar sin necesidad de usar la ventana terminal | 3 | 5 | 15 |
| Manejo del Sistema basado en consola | Buena, ya que tiene varias herramientas para gestión del equipo de manera integral, tarjeta de red, audio, video, periféricos. | 5 | 5 | 25 |
| SISTEMA DE PAQUETES | | | | |
| Cantidad de Paquetes | Muy completo, El número de paquetes registrados con yum es aproximadamente 5785. Cuenta con repositorios activados como rpmforge, update, kbs-CentOS-Extras, addons. | 4 | 3 | 12 |
| Gestión de Paquetes y Resolución Automática de Dependencias | El gestor por defecto es yum, heredado de la fuente RHL[78] | 5 | 4 | 20 |
| EFICIENCIA | | | | |
| Velocidad del Sistema de Arranque **(Item4)** | Es dependiente de su configuración y servicios activados. Aproximadamente de 1 a 2 minutos en una instalación por defecto | 5 | 5 | 25 |

[78] Red Hat Linux: distribución de Linux basada en paquetes, orientada a comercialización en el ámbito GNU/LINUX e integración de nuevas tecnologías

| | | | | |
|---|---|---|---|---|
| Velocidad de Respuesta del Sistema **(Item5)** | Muy buena, con optimizaciones para uso de escritorio o servidor, los cuales deben ser configurados manualmente | 5 | 5 | 25 |
| ESTABILIDAD Y DISPONIBILIDAD | | | | |
| Centro de Seguridad **(Item3)** | Muy buena, ofrece seguridad a nivel chip y memoria, con la desventaja de que requiere parches para vulnerabilidades detectadas. | 5 | 5 | 25 |
| Estabilidad y Madurez **(Item2)** | Excelente, ya que se basa en paquetes de software bien probados y conservados que aseguran el funcionamiento adecuado. | 5 | 5 | 25 |
| Documentación **(Item6)** | Buena, ya que existe documentación variada, incluso proveniente de Red Hat Linux y desarrolladores de la comunidad | 5 | 4 | 20 |

- **Ubuntu**

| Criterio | Detalles | Peso | Evaluación | Total |
|---|---|---|---|---|
| CONFIGURACION | | | | |
| Licencia **(Item1)** | Los componentes básicos son GNU/GPL | 5 | 3 | 15 |
| Manejo del Sistema basado en Grafico | Cuenta con herramientas especificas por defecto: notificación de actualizaciones, gestor de instalaciones, sin embargo no tiene un panel de control específico de su actualización. | 3 | 4 | 12 |
| Manejo del Sistema basado en consola | Posee una herramienta de configuración denominada debconf, proviene del proyecto Debián y está disponible en las herramientas de configuración del estándar Debián. | 5 | 1 | 5 |
| SISTEMA DE PAQUETES | | | | |
| Cantidad de Paquetes | Puede existir alrededor de 10000 paquetes específicos para Ubuntu. | 4 | 3 | 12 |

| | | | | |
|---|---|---|---|---|
| Gestión de Paquetes y Resolución Automática de Dependencias | Permite encontrar paquetes certificados y no certificados, se los puede gestionar a través de APT, sin embargo cuenta con Smart que se considera superior a APT. | 5 | 1 | 5 |
| EFICIENCIA | | | | |
| Velocidad del Sistema de Arranque **(Item4)** | Con los servicios de configuración por defecto el arranque lleva un promedio de 1 minuto | 5 | 5 | 25 |
| Velocidad de Respuesta del Sistema **(Item5)** | Especialmente para estaciones de trabajo cuenta con configuraciones manuales para optimizar el sistema | 5 | 4 | 20 |
| ESTABILIDAD Y DISPONIBILIDAD | | | | |
| Centro de Seguridad **(Item3)** | Cuenta con paquetes de protección de memoria y núcleo, sin embargo no tiene herramientas propias de cortafuegos | 5 | 4 | 20 |
| Estabilidad y Madurez **(Ítem 2)** | Está basado en debián que es estable y maduro actualmente, pero al tener software adicional se pueden tener situaciones de inestabilidad | 5 | 4 | 20 |
| Documentación **(Item6)** | Permite obtenerla fácilmente desde la página oficial del proyecto, comunidades y manuales del sistema. | 5 | 3 | 15 |

# ANEXO 6

# VALORES DE PROCESAMIENTO TOMADOS DE MONITOREO SERVER

**VALORES (EN JIFFIES) DE PROCESAMIENTO 19 DE NOVIEMBRE DEL 2012**

| Hora de muestra | T1A | T1B | T1C | T1D | T0A | T0B | T0C | T0D | Carga CPU |
|---|---|---|---|---|---|---|---|---|---|
| 11h00 | 5171937 | 15548 | 493591 | 16263594 | 5171935 | 15548 | 493579 | 16263424 | 7, 608695 |
| 11h30 | 5184475 | 15549 | 495419 | 16339441 | 5184182 | 15549 | 495344 | 16338975 | 44,1247002 |
| 12h00 | 5224438 | 15549 | 499328 | 16428886 | 5224329 | 15549 | 499295 | 16428662 | 38,7978142 |
| 12h30 | 5232894 | 15549 | 500229 | 16443302 | 5232738 | 15549 | 500194 | 16442268 | 15,5918367 |
| 13h00 | 5241161 | 15549 | 500933 | 16453268 | 5240899 | 15549 | 500912 | 16453231 | 88,4375 |
| 13h30 | 5244618 | 15549 | 501340 | 16466593 | 5244614 | 15549 | 501336 | 16466391 | 3,80952381 |
| 14h00 | 5253755 | 15549 | 502022 | 16482977 | 5253717 | 15549 | 502016 | 16482682 | 12,979351 |
| 14h30 | 5257018 | 15549 | 502302 | 16499166 | 5257017 | 15549 | 502301 | 16498831 | 0,59347181 |
| 15h00 | 5260101 | 15549 | 502618 | 16517401 | 5260089 | 15549 | 502607 | 16517224 | 11,5 |
| 15h30 | 5264927 | 15549 | 503141 | 16554807 | 5264854 | 15549 | 503134 | 16554480 | 19,6560197 |
| 16h00 | 5271367 | 15549 | 503689 | 16600881 | 5271352 | 15549 | 503686 | 16600588 | 5,78778135 |
| 16h30 | 5277762 | 15549 | 504053 | 16612879 | 5277548 | 15549 | 504042 | 16612647 | 49,2341357 |
| 17h00 | 5280273 | 15549 | 504225 | 16624967 | 5280134 | 15549 | 504217 | 16624829 | 51,5789474 |
| 17h30 | 5284441 | 15549 | 504519 | 16642440 | 5284418 | 15549 | 504518 | 16642231 | 10,3004292 |
| 18h00 | 5292076 | 15549 | 505239 | 16661859 | 5292073 | 15549 | 505238 | 16661547 | 1,26582278 |
| 18h30 | 19833 | 714 | 4753 | 94970 | 19644 | 714 | 4737 | 94801 | 54,8128342 |
| 19h00 | 59608 | 714 | 7918 | 544714 | 59598 | 714 | 7917 | 544443 | 3,90070922 |
| Promedio de Carga del CPU: | | | | | | | | | 25,56 % |

## Según la ecuación:

$$\text{ocupación CPU} = 100 * \frac{(\text{carga cpu T1} - \text{carga cpu T0})}{(\text{uso total cpu T1} - \text{uso total cpu T0})}$$

*T1A y T0A*: Numero de Jiffies usados por el CPU en modo usuario para T=1 y T=0 respectivamente

*T1B y T0B*: Numero de Jiffies en modo usuario de baja prioridad para T=1 y T=0 respectivamente

*T1C y T0C*: Numero de jiffies usados por el sistema para T=1 y T=0 respectivamente

*T1D y T0C*: jiffies de Inactividad para T=1 y T=0 respectivamente

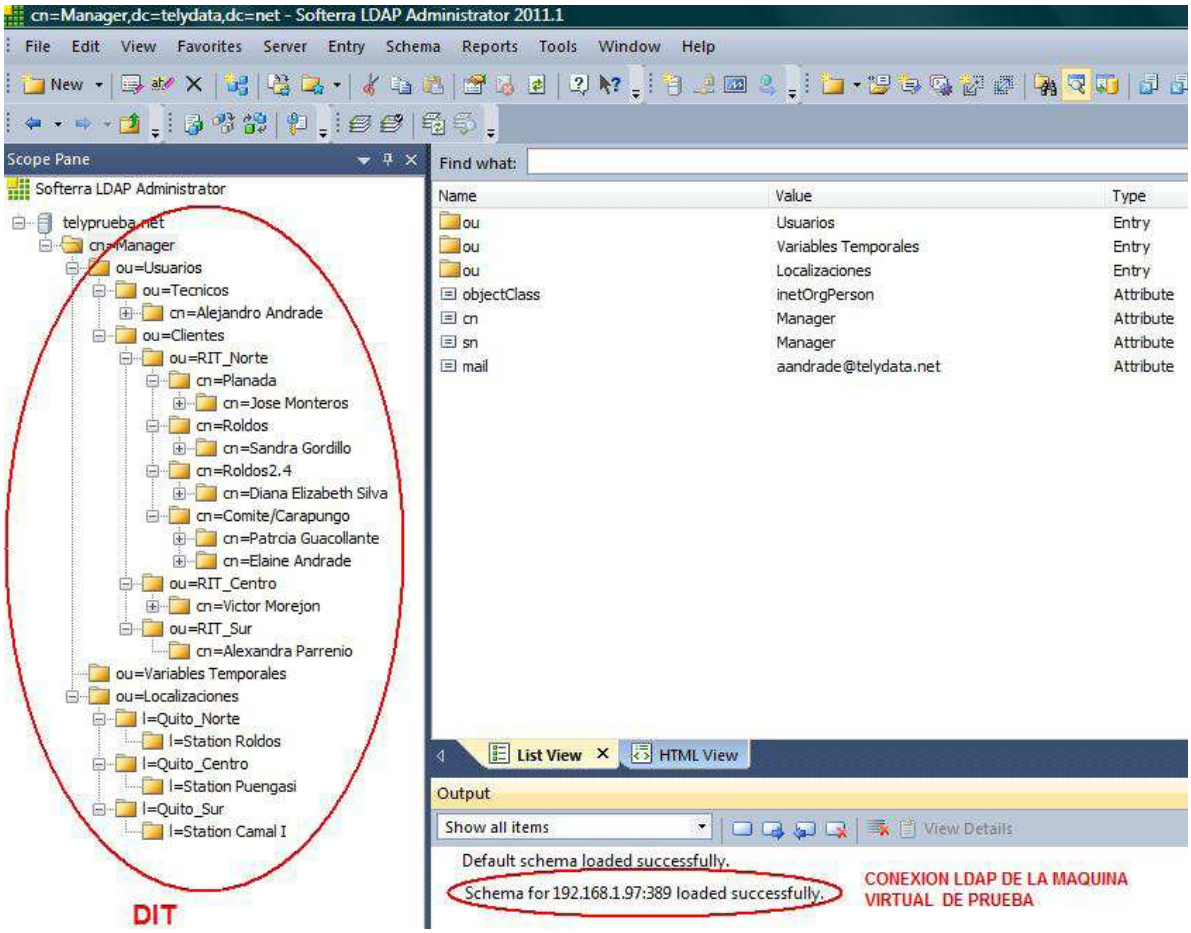
*carga cpu T1* = (T1A +T1B+T1C), *carga cpu T0* = (T0A+T0B+T0C)

*uso total cpu T1*= (*carga cpu T1) + T1D, uso total cpu T0= (carga cpu T0) + T0D*

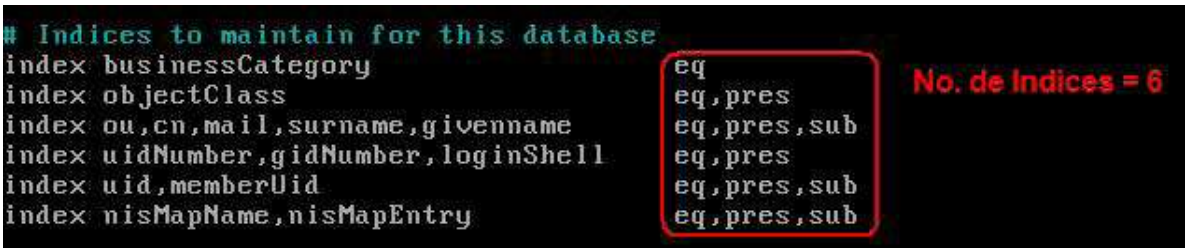
# ANEXO 7

# REPOSITORIO DE INFORMACIÓN DE PRUEBA

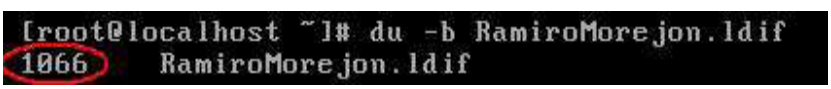
# REPOSITORIO DE PRUEBA



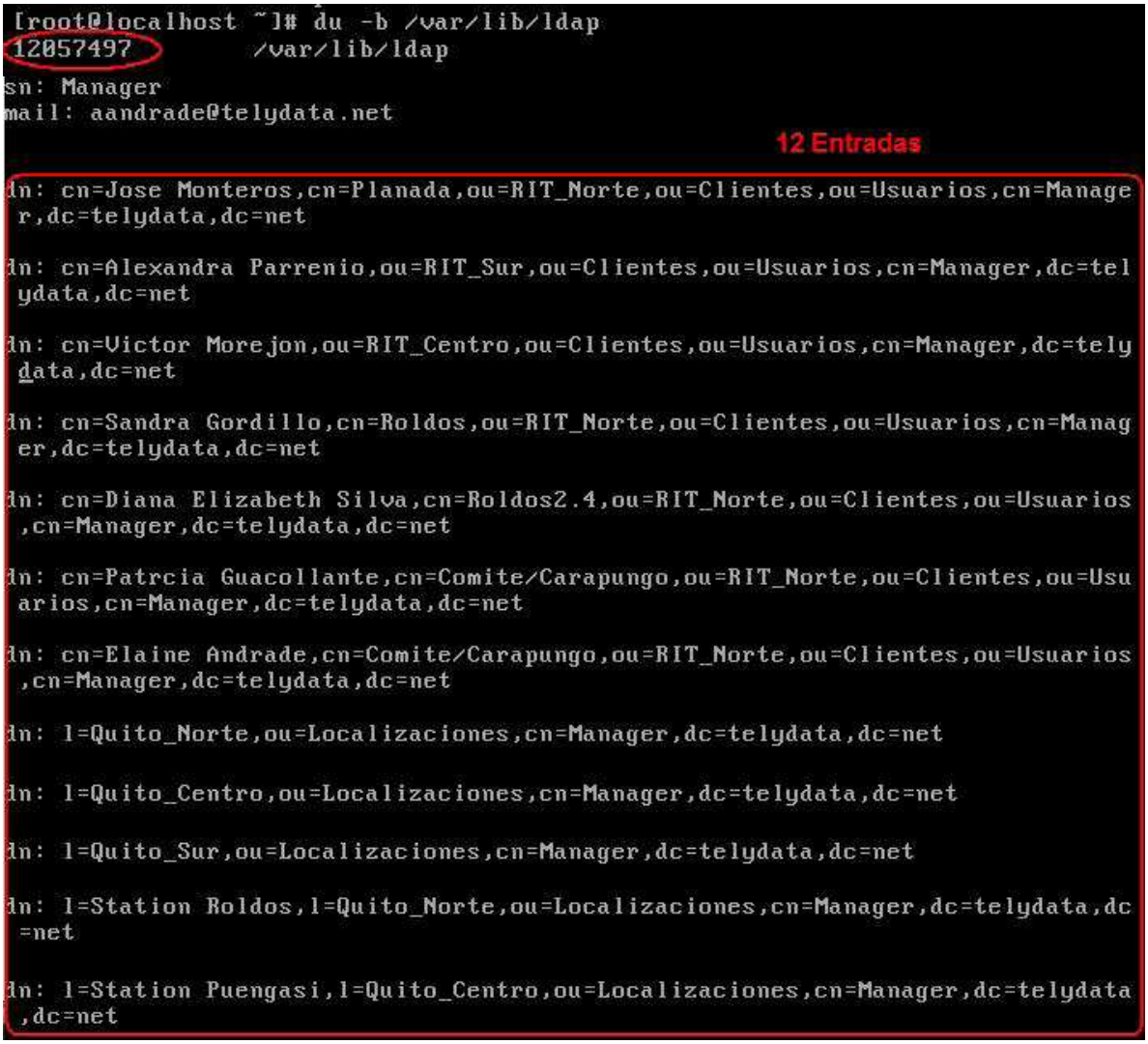

1. Numero de Índices:




2. Tamaño por entrada (Ejemplo Ramiro Morejon):



3. Número de Entradas:

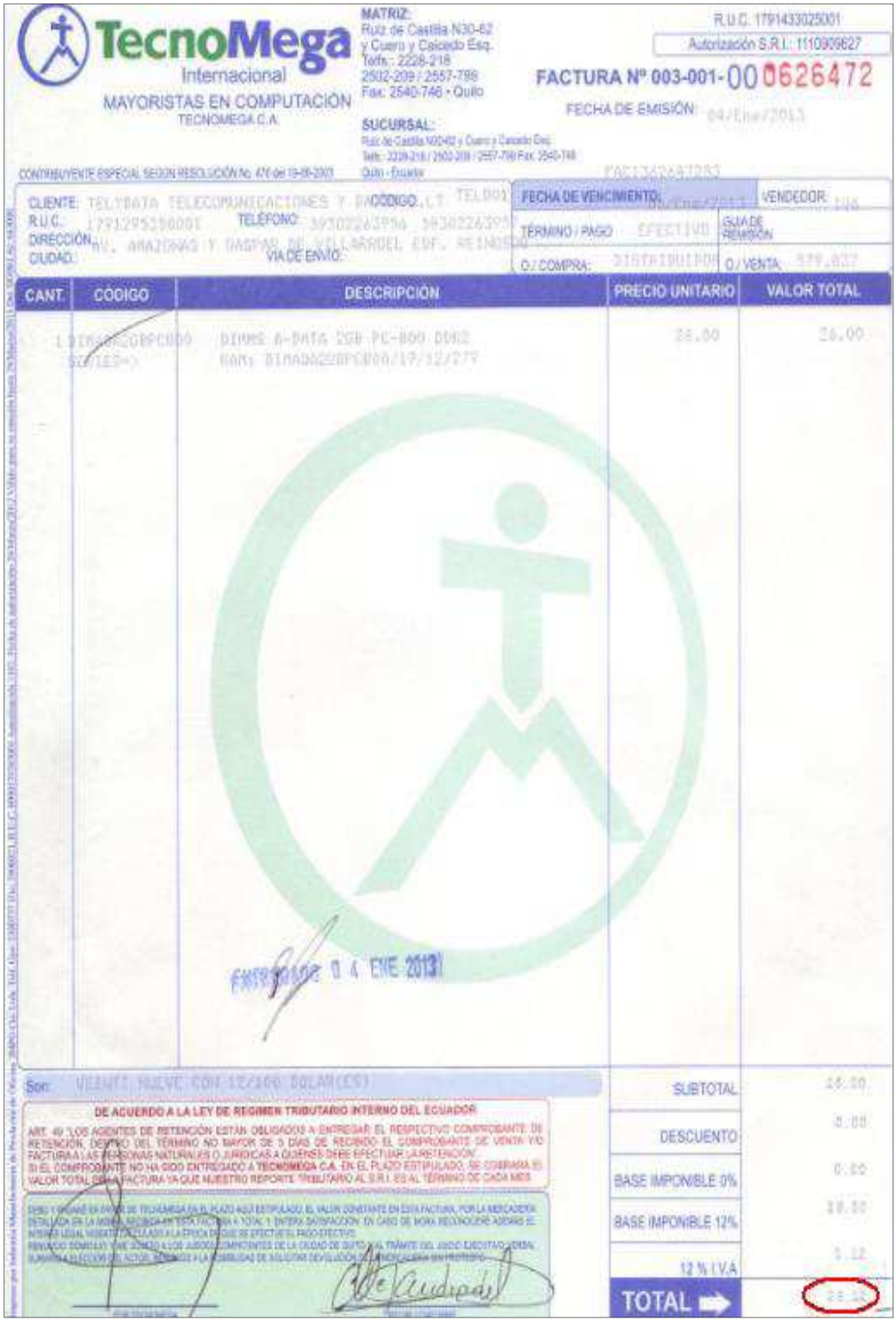



4. Tamaño del archivo de indexado

# ANEXO 8

# FACTURAS DE LOS DISPOSITIVOS ADQUIRIDOS PARA LA IMPLEMENTACIÓN DEL PROYECTO

- **Costo Memoria RAM**



TecnoMega
Internacional
MAYORISTAS EN COMPUTACIÓN
TECNOMEGA C.A.

MATRIZ:
Ruiz de Castilla N30-62
y Cuero y Caicedo Esq.
Telfs.: 2228-218
2502-209 / 2557-788
Fax: 2540-746 - Quito

SUCURSAL:
Ruiz de Castilla N30-62 y Cuero y Caicedo Esq.
Telfs.: 2228-218 / 2502-209 / 2557-788 Fax: 2540-746
Quito - Ecuador

R.U.C. 1791433025001
Autorización S.R.I.: 1110909627

FACTURA Nº 003-001-00 0626472

FECHA DE EMISIÓN: 04/Ene/2013

CONTRIBUYENTE ESPECIAL SEGÚN RESOLUCIÓN No. 470 del 15-08-2003

CLIENTE: TELYBATA TELECOMUNICACIONES Y [illegible] CODIGO: [illegible]
R.U.C.: 1791295336001 TELÉFONO: 3930226/3956 3930226/3956
DIRECCIÓN: AV. AMAZONAS Y GASPAR DE VILLARROEL ESQ. [illegible]
CIUDAD: VIA DE ENVIO:

FECHA DE VENCIMIENTO: [illegible] VENDEDOR: [illegible]
TÉRMINO / PAGO: EFECTIVO GUIA DE REMISION:
O./ COMPRA: 3197A1901P09 O./ VENTA: 539.837

| CANT | CODIGO | DESCRIPCION | PRECIO UNITARIO | VALOR TOTAL |
|---|---|---|---|---|
| 1 | [illegible] | DIMM A-DATA 2GB PC-800 DDR2 RAM: DIMADDR2PC800/1P/32/27Y | 26.00 | 26.00 |

ENTREGADO 04 ENE 2013

Son: VEINTE Y SEIS CON 12/100 DOLAR(ES)

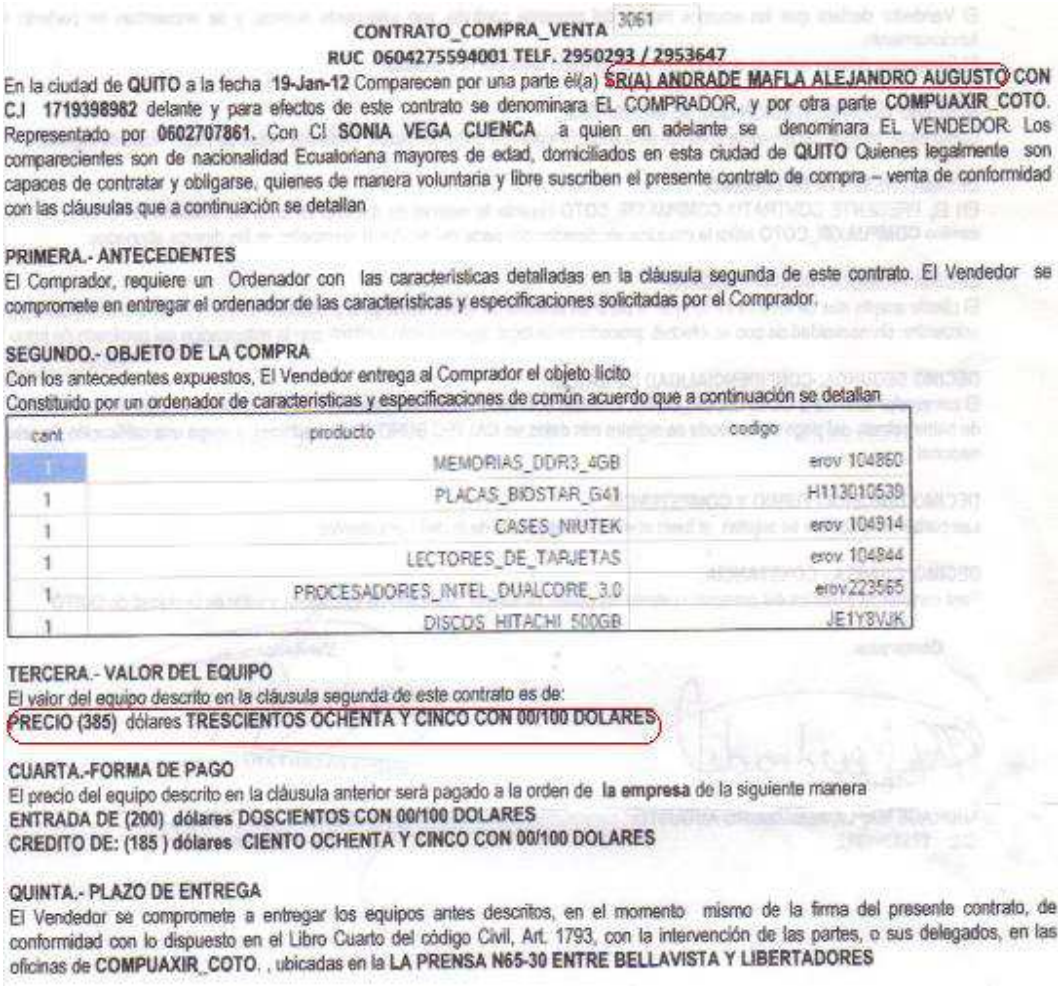
| | |
|---|---|
| SUBTOTAL | 26.00 |
| DESCUENTO | 0.00 |
| BASE IMPONIBLE 0% | 0.00 |
| BASE IMPONIBLE 12% | 26.00 |
| 12 % I.V.A | 3.12 |
| TOTAL | 29.12 |

DE ACUERDO A LA LEY DE REGIMEN TRIBUTARIO INTERNO DEL ECUADOR

ART. 49 "LOS AGENTES DE RETENCIÓN ESTÁN OBLIGADOS A ENTREGAR EL RESPECTIVO COMPROBANTE DE RETENCIÓN, DENTRO DEL TÉRMINO NO MAYOR DE 5 DÍAS DE RECIBIDO EL COMPROBANTE DE VENTA Y/O FACTURA A LAS PERSONAS NATURALES O JURÍDICAS A QUIENES DEBE EFECTUAR LA RETENCIÓN"
SI EL COMPROBANTE NO HA SIDO ENTREGADO A TECNOMEGA C.A. EN EL PLAZO ESTIPULADO, SE COBRARÁ EL VALOR TOTAL DE LA FACTURA YA QUE NUESTRO REPORTE TRIBUTARIO AL S.R.I. ES AL TÉRMINO DE CADA MES

[illegible]

- **Costo Policy Server Compuaxir**



CONTRATO_COMPRA_VENTA 3061

RUC 0604275594001 TELF. 2950293 / 2953647

En la ciudad de **QUITO** a la fecha **19-Jan-12** Comparecen por una parte él(a) **SR(A) ANDRADE MAFLA ALEJANDRO AUGUSTO** con **C.I 1719398982** delante y para efectos de este contrato se denominara EL COMPRADOR, y por otra parte **COMPUAXIR_COTO.** Representado por **0602707861.** Con CI **SONIA VEGA CUENCA** a quien en adelante se denominara EL VENDEDOR. Los comparecientes son de nacionalidad Ecuatoriana mayores de edad, domiciliados en esta ciudad de **QUITO** Quienes legalmente son capaces de contratar y obligarse, quienes de manera voluntaria y libre suscriben el presente contrato de compra – venta de conformidad con las cláusulas que a continuación se detallan

**PRIMERA.- ANTECEDENTES**

El Comprador, requiere un Ordenador con las caracteristicas detalladas en la cláusula segunda de este contrato. El Vendedor se compromete en entregar el ordenador de las caracteristicas y especificaciones solicitadas por el Comprador.

**SEGUNDO.- OBJETO DE LA COMPRA**

Con los antecedentes expuestos, El Vendedor entrega al Comprador el objeto licito

Constituido por un ordenador de caracteristicas y especificaciones de común acuerdo que a continuación se detallan

| cant | producto | codigo |
|---|---|---|
| 1 | MEMORIAS_DDR3_4GB | erov 104860 |
| 1 | PLACAS_BIOSTAR_G41 | H113010539 |
| 1 | CASES_NIUTEK | erov 104914 |
| 1 | LECTORES_DE_TARJETAS | erov 104944 |
| 1 | PROCESADORES_INTEL_DUALCORE_3.0 | erov223565 |
| 1 | DISCOS_HITACHI_500GB | JE1Y8VJK |

**TERCERA.- VALOR DEL EQUIPO**

El valor del equipo descrito en la cláusula segunda de este contrato es de:

**PRECIO (385) dólares TRESCIENTOS OCHENTA Y CINCO CON 00/100 DOLARES**

**CUARTA.-FORMA DE PAGO**

El precio del equipo descrito en la cláusula anterior será pagado a la orden de **la empresa** de la siguiente manera

**ENTRADA DE (200) dólares DOSCIENTOS CON 00/100 DOLARES**

**CREDITO DE: (185 ) dólares CIENTO OCHENTA Y CINCO CON 00/100 DOLARES**

**QUINTA.- PLAZO DE ENTREGA**

El Vendedor se compromete a entregar los equipos antes descritos, en el momento mismo de la firma del presente contrato, de conformidad con lo dispuesto en el Libro Cuarto del código Civil, Art. 1793, con la intervención de las partes, o sus delegados, en las oficinas de **COMPUAXIR_COTO.** , ubicadas en la **LA PRENSA N65-30 ENTRE BELLAVISTA Y LIBERTADORES**

A esta factura se restan los siguientes valores, puesto que son reemplazados con los costos de la Factura Tecnomega:

Costo Placa Biostar G41: 42.80 dólares + IVA

Costo Procesador Intel Dual core 3.0: 132.00 dólares + IVA

Fuente: Compuaxir.

- **Costo Policy Server Tecnomega**

TecnoMega Internacional
MAYORISTAS EN COMPUTACIÓN
TECNOMEGA C.A.

MATRIZ:
Ruiz de Castilla N30-62
y Cuero y Caicedo Esq.
Telfs.: 2228-218
2502-209 / 2557-709
Fax: 2540-746 • Quito

SUCURSAL:
Ruiz de Castilla N30-62 y Cuero y Caicedo Esq.
Telfs.: 2228-218 / 2502-209 / 2557-709 Fax: 2540-746
Quito - Ecuador

R.U.C. 1791433025001
Autorización S.R.I.: 1110909627

FACTURA Nº 003-001-000613285

FECHA DE EMISIÓN: 18/Oct/2012

CONTRIBUYENTE ESPECIAL SEGÚN RESOLUCIÓN No. 475 del 19-05-2003

FAC1261328563

CLIENTE: TELYDATA TELECOMUNICACIONES Y DATOS CIA LT
R.U.C.: 1791295358001
TELÉFONO: 59302263956 59302263957
DIRECCIÓN: AV. AMAZONAS Y GASPAR DE VILLAROEL EDF. REINOSO
CIUDAD:
VÍA DE ENVÍO:

CÓDIGO: TELD01
FECHA DE VENCIMIENTO: 18/Oct/2012
VENDEDOR: IVA
TÉRMINO / PAGO: CREDITO 8 DIAS
GUÍA DE REMISIÓN:
O/ COMPRA: DISTRIBUIDOR
O/ VENTA: 562.895

| CANT. | CÓDIGO | DESCRIPCIÓN | PRECIO UNITARIO | VALOR TOTAL |
|---|---|---|---|---|
| 1 | [illegible] | MBO INTEL DH61CR I7.LGA1155.DDR3.SN.VID.RE | 71.00 | 71.00 |
| 1 | [illegible] | PROC. INTEL CORE I3-3220 3.3Ghz 3MB | 140.00 | 140.00 |
| | SERIES=> | CPU: 2V224631A5434 MBO: BTCR22201MF3 | | |

ENTREGADO 18 OCT 2012

Son: DOSCIENTOS TREINTA Y SEIS CON 32/100 DOLARES

| | |
|---|---|
| SUBTOTAL | 211.00 |
| DESCUENTO | 0.00 |
| BASE IMPONIBLE 0% | 0.00 |
| BASE IMPONIBLE 12% | 211.00 |
| 12 % I.V.A. | 25.32 |
| TOTAL | 236.32 |

DE ACUERDO A LA LEY DE REGIMEN TRIBUTARIO INTERNO DEL ECUADOR

ART. 49 "LOS AGENTES DE RETENCIÓN ESTÁN OBLIGADOS A ENTREGAR EL RESPECTIVO COMPROBANTE DE RETENCIÓN DENTRO DEL TÉRMINO NO MAYOR DE 5 DÍAS DE RECIBIDO EL COMPROBANTE DE VENTA VIO FACTURA A LAS PERSONAS NATURALES O JURÍDICAS A QUIENES DEBE EFECTUAR LA RETENCIÓN"
SI EL COMPROBANTE NO HA SIDO ENTREGADO A TECNOMEGA C.A. EN EL PLAZO ESTIPULADO, SE COBRARÁ EL VALOR TOTAL DE LA FACTURA YA QUE NUESTRO REPORTE TRIBUTARIO AL S.R.I. ES AL TÉRMINO DE CADA MES.

DEBO Y PAGARÉ EN FAVOR DE TECNOMEGA EN EL PLAZO ESTIPULADO EL VALOR CONSTANTE EN ESTA FACTURA, POR CANCELACIÓN DETALLADA EN LA MISMA, RECIBIDA EN ESTA FACTURA A TOTAL Y ENTERA SATISFACCIÓN, EN CASO DE MORA RECONOCERÉ ADEMÁS EL INTERÉS LEGAL VIGENTE CALCULADO A LA TASA MÁS ALTA QUE ESTUVIERE EN EL MOMENTO DE EFECTUAR EL PAGO EFECTIVO, RENUNCIO DOMICILIO Y ME SUJETO A LOS JUECES COMPETENTES DE LA CIUDAD DE QUITO Y AL TRÁMITE DEL JUICIO EJECUTIVO VERBAL SUMARIO A ELECCIÓN DEL ACTOR. RENUNCIO A LA PROPIEDAD DE MI DERECHO DE RETENCIÓN Y ACEPTO EXPRESAMENTE

1719598982