PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

EXTENSIÓN DE UN GESTOR DE REDES BASADO EN SOFTWARE LIBRE PARA UN OPERADOR MÓVIL

Tesis para optar el Título de Ingeniero de las Telecomunicaciones, que presenta el bachiller:

HERNÁN ROMANO CURO

ASESOR: MG. ANTONIO OCAMPO ZÚÑIGA

Lima, julio de 2013

Resumen

En la presente tesis se desarrollará el diseño e implementación de la extensión de un gestor de redes basado en software libre de un operador nacional de telecomunicaciones móviles con el objetivo de monitorear parámetros de red que no son posibles conocer vía el protocolo SNMP. Para lo cual se hizo uso de protocolos de acceso remoto tales como el protocolo Telnet y SSH.

El procedimiento a seguir en el desarrollo de la presente tesis se realiza de la siguiente manera. En el capítulo 1, se identificará el actual problema suscitado en el operador móvil, además se analizará las necesidades de los operadores de red, asimismo se limitará el alcance y plantearan objetivos para la búsqueda de una solución concreta.

En el capítulo 2, se expondrá los temas necesarios a conocer para el desarrollo y buen entendimiento de las etapas de diseño e implementación realizadas en el capítulo 3. Para lo cual, se tendrá en consideración las características de los protocolos y software's, bajo licencias propietarias o licencias libres, que se encuentran actualmente disponibles en el mercado.

Finalmente, en el capítulo 4 se analizarán los resultados obtenidos a raíz de las distintas pruebas realizadas. Además, se realizará un análisis de costos del diseño e implementación del sistema y se plantearan mejoras para la realización de futuros trabajos.

Dedicatoria

A mis padres Juana y Juan

Agradecimientos

A mis padres quienes siempre me apoyaron y brindaron las herramientas necesarias para desarrollarme, tanto como persona y futuro profesional; además de preocuparse constantemente en toda esta etapa de mi vida.

A mis hermanos y abuelos por haberme brindado sus consejos y sabiduría.

A mi asesor de tesis, Mg. Antonio Ocampo Zúñiga, por haberme ayudado con la formación del tema de la presente tesis; así como su constante apoyo durante las largas jornadas de trabajo para alcanzar los objetivos planteados.

A todos mis amigos, con los cuales he compartido grandes momentos de mi vida y experiencias.

Y finalmente a Jessica, porque sin su apoyo y constantes consejos en los momentos críticos y complicados, no hubiera sido capaz de superarlos.

A cada uno de ustedes, un muy fuerte abrazo y ¡Muchas Gracias!

Índice

Lista de figuras

Lista de tablas

Introducción

La presente tesis propone una solución a la gestión de equipos que únicamente cuentan con una implementación básica o nula del protocolo de gestión SNMP. El diseño propuesto se implementó en la red de un proveedor de servicios nacional de telecomunicaciones móviles. Asimismo, las pruebas se extendieron en una ambiente de laboratorio controlado para el fin de demostrar el diseño planteado.

Cabe recalcar que el desarrolló se llevó a cabo en equipos de red GGSN que se encontraban en funcionamiento y brindando un servicio continuo a clientes corporativos del operador móvil; el cual permitía a los usuarios conectarse a la red de Internet móvil.

La implementación se llevó acabo utilizando protocolos de acceso remoto tales como Telnet y SSH, los cuales fueron utilizados para la extracción de la información de las entidades de la red que contaban con una implementación básica o nula del protocolo SNMP.

Finalmente, los resultados obtenidos en la implementación demostraron el correcto funcionamiento del diseño planteado en el presente trabajo; para lo cual se realizaron gráficas y tablas que lo demuestren.

Capítulo 1

Identificación del problema en el operador móvil

1.1 Introducción

En las telecomunicaciones, específicamente, el centro de gestión de red se considera un área sumamente crítica debido a que desde este punto se monitorea el estado de una red y se asegura un servicio continuo y de buena calidad.

Es por ello que las herramientas utilizadas por las personas que trabajan en el centro de red de un proveedor de servicios suelen manejar herramientas que les permiten gestionar los diferentes dispositivos de manera remota y; de esta forma, lograr resolver cualquier tipo de problemas en el menor tiempo posible; pero para ello es muy importante contar con un gestor de red que extraiga la mayor cantidad de información de los cientos de dispositivos que se encuentran en la red.

1.2 Estudio del escenario gestionado

1.2.1 Problema actual

En los proveedores de servicios de telecomunicaciones existe la necesidad de comprar distintos dispositivos con diferentes tecnologías; las cuales se puedan adaptar a las problemáticas o mejoras en calidad de servicio que se desee ofrecer. Para ello se realizan análisis de rentabilidad; los cuales se basan en adquirir la mejor opción entre la gran cantidad de ofertas provenientes de distintos fabricantes de tecnologías para la compra de equipos como los utilizados en networking.

El problema del uso de equipos, provenientes de diferentes fabricantes, se encuentra en que estos equipos son monitoreados con una herramienta de gestión de red propia del fabricante la cual analiza la información proveniente de los equipos tales como: el nivel de procesamiento del CPU, la capacidad de las interfaces, la cantidad de recursos disponibles, entre otros parámetros.

Esto implica que para cada dispositivo se necesite un software de gestión de red distinto para cada fabricante; debido a que algunos fabricantes implementan sus propios protocolos de gestión; lo cual exige que se utilice software propietario para el monitoreo de sus equipos.

Este problema ha afectado al operador de telecomunicaciones móviles sobre la cual se desarrolla la presente tesis; debido a que el margen de rentabilidad de los proyectos planteados, disminuye por los pagos de licencia necesarios por cada nuevo dispositivo agregado a la red del operador. Y además, el uso de distintas plataformas se ha vuelto tedioso para los operadores encargados de la red actual; provocando que su trabajo se vuelva ineficiente.

1.2.2 Necesidades del centro de gestión

Las necesidades identificadas en el centro de gestión y operaciones son:

- Obtener la evolución histórica con intervalos de 5 minutos de parámetros de la red de datos móviles (Packet Core) y la red móvil de voz, que actualmente no pueden ser monitoreados actualmente usando SNMP.
  Por ejemplo, actualmente existen equipos dentro de la red móvil que únicamente muestran estadísticas de los últimos 30 minutos acerca de los

recursos de red, y pasado este tiempo, se refrescan los contadores, haciendo así, imposible conocer el comportamiento de los usuarios y el correcto funcionamiento del equipo. Un ejemplo de ello son los equipos MSS de Nokia.

- Obtener estadísticas de los usuarios conectados a la red corporativa y evaluar sus comportamientos.

- Uso de una única plataforma de monitoreo de red, común para todos los fabricantes.

- Optimizar el análisis de la utilización de la red, de manera que se determine el estado y eficiencia de la misma; además de preparar los requerimientos de la red en un futuro.

1.3 Objetivos

Objetivo general:

- Extender las funcionalidades del actual gestor de red utilizado en el operador móvil mediante la integración de las distintas tecnologías, basándonos en software libre.

Objetivos específicos:

- Extraer datos esenciales de equipos, que trabajan bajo un protocolo de gestión propietario e integrarlos a la actual plataforma del operador móvil.

- Realizar la implementación haciendo uso de software libre.

Capítulo 2

Conceptos de gestión de redes

2.1 Definición

La gestión de red es un concepto general que implica el uso de distintas herramientas, técnicas y sistemas que le permiten a un operador administrar la gran gama de dispositivos y sistemas de forma más sencilla y; de esta manera, ejecutar las acciones pertinentes en caso ocurra algún suceso que afecte el funcionamiento de la red.

2.2 Antecedentes

En la actualidad existen distintos protocolos dedicados a la gestión de red los cuales se pueden diferenciar entre propietarios, como NBAR de Cisco Systems [1], protocolos de gestión de Huawei y Nokia; y libres como es el caso del SNMP.

Al no pagarse ninguna licencia por este último protocolo, este es implementado en todo tipo de equipos de una red. Por ejemplo, dentro de la red del operador móvil sobre la cual se trabaja en esta tesis, se hallan equipos tales como GGSNs y MSSs los cuales

desean ser monitoreados debido a la importancia de la información sobre el comportamiento de los usuarios con la que cuentan [2].

Trabajos relacionados, se pueden encontrar en [3]; en el cual se monitorea un servidor de telefonía de la RAAP (Red Académica Peruana) la cual no cuenta con soporte SNMP.

2.3 Modelo de gestión de redes

Se describirá un modelo realizado por la organización ISO [4] en la cual se determinan los objetivos de un administrador para gestionar de forma plena su red.

Este modelo es conocido como FCAPS y como todo modelo debe ser moldeado a la situación actual sobre la cual se aplicará de acuerdo a nuestras necesidades de gestión. Por ello solo será utilizado como referencia; debido a que no se implementarán todas sus áreas funcionales; las cuales por sus siglas en inglés hace referencia a Faul-Management, Configuration, Accounting, Performance, Security [5].

2.3.1 Gestión de fallas

Lo que se desea de una red es que al originarse un error, este pueda ser reconocido, aislado, corregido y registrar el suceso para que en otras oportunidades se tome una acción rápida basada en la solución tomada, o sirva para realizar un diagnóstico. De esta manera la red permanecerá siempre disponible.

2.3.2 Gestión de configuraciones

En las redes actuales existe todo un personal especializado; el cual se encarga de acceder remotamente a los equipos de una red para su debida configuración; para esto lo que se desea es llevar un control sobre los cambios que ocurran en la configuración de los equipos; además de tener información de cada entidad de la red como sus características y capacidades.

2.3.3 Gestión de contabilidad

Este punto hace referencia a asegurar que los recursos de red estén siendo utilizados de forma correcta y para ello también saber, quién está haciendo usos de los recursos de la red.

2.3.4 Gestión de performance

En esta sección de debe determinar el estado y eficiencia de la red, mediante la observación y constante análisis de la utilización de la red; para poder predecir tendencias y preparar la red para futuros proyectos que impliquen brindar nuevos servicios a los usuarios.

A parte de monitorear el uso del ancho de banda en una interfaz, también debe ser conveniente extraer información sobre los servicios que se estén brindando a los usuarios de la red.

2.3.5 Gestión de seguridad

Un punto crítico en la gestión de redes es la seguridad. Esta categoría está relacionada con el acceso de recursos y la protección frente ataques por parte de personas mal intencionadas, sin permiso a ingresar a la red o realizar cambios sobre las distintas entidades.

2.4 Protocolos de gestión de redes

2.4.1 Protocolos libre

2.4.1.1 SNMP

Simple Network Management Protocol (SNMP) es un protocolo que permite realizar la gestión remota de dispositivos pertenecientes a una red IP [6]. A diferencia de su antecesor, el SGMP (Simple Network Gateway Management Protocol); el cual únicamente administraba routers, el SNMP extendió sus funcionalidades a una mayor cantidad de dispositivos tales como sistemas Unix, sistemas Windows, impresoras, UPS, entre otros.

En general, cualquier dispositivo que soporte la implementación del protocolo; esto incluye servidores web y base de datos.

SNMP ha sufrido mejoras y variaciones desde sus inicios, es por ello que existen distintas versiones de SNMP, las cuales serán mencionadas en la siguiente tabla:

Tabla 1.1: Características principales de las versiones SNMP

Versión	Característica principal
SNMPv1	• Seguridad basada en comunidades. • Passwords sin encriptación.
SNMPv2p	• Passwords encriptados. • Complicado de implementar, es por ello que no es utilizado
SNMPv2c	• Seguridad basada en comunidades • Passwords encriptados. • Soportado por la mayoría de los fabricantes. • Mayor variedad en los tipos de datos.
SNMPv3	• Aumento de seguridad mediante el uso de comunidades privadas.

En la actualidad, la mayor cantidad de fabricantes tienen implementado la versión SNMPv2c y se están trasladando, progresivamente, de acuerdo a la seguridad que necesiten los clientes, a la versión SNMPv3. Por esta razón, se desarrollará la presente tesis en base a la versión SNMPv2c.

Arquitectura del sistema SNMP

El SNMP usa dos tipos de entidades para su correcta aplicación: Administrador y Agente. El administrador es un servidor que tiene implementado un sistema de software el cual se encarga de las tareas de gestión de una red. El sistema que realiza la gestión es llamado NMS (*Network Management System*). Por el otro lado, el agente es un programa que se encuentra alojado en un dispositivo de red; los cuales realizan las acciones que solicite el NMS.

Las acciones que permiten la extracción de datos y notificación de sucesos de un agente, se muestra en la figura 2.1. El NMS se encarga de enviar *queries* al agente, el cual responde con la información solicitada con un mensaje *response*. El *trap* hace referencia a los mensajes que envía el agente al NMS por la ocurrencia de algún suceso inesperado.

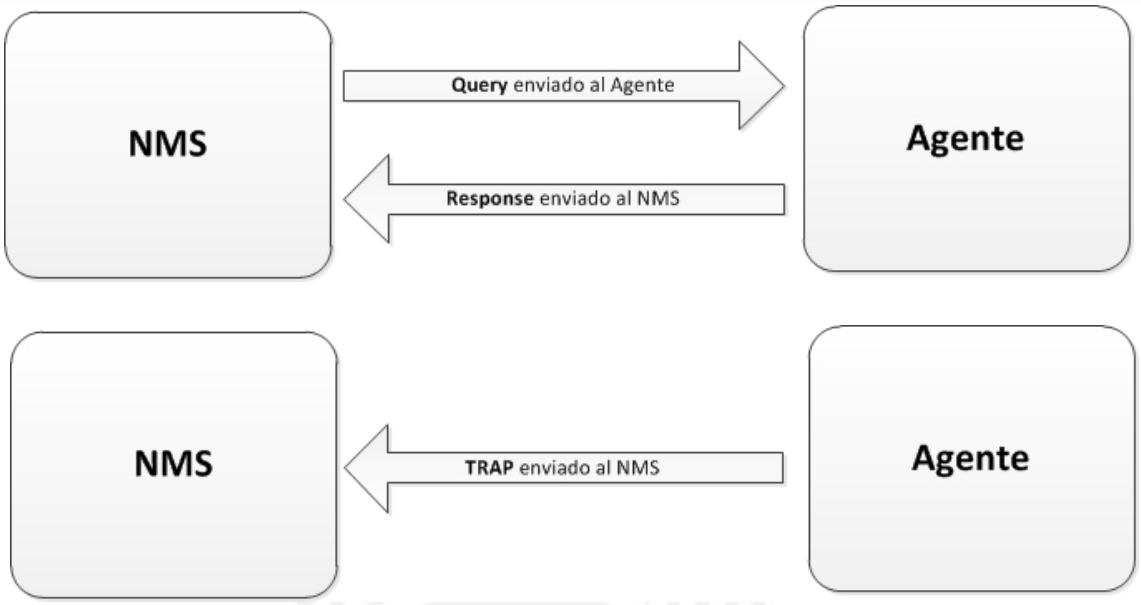

Figura 2.1: Mensajes entre NMS y un Agente

La combinación de los mensajes *query y response*, conforman el denominado *pooling* de datos; el cual se caracteriza por ser periódico, a diferencia de los *traps* que son asíncronos.

Datagrama SNMP

SNMP usa el protocolo UDP (*User Datagram Protocol*) para permitir el transporte de los paquetes de información entre el agente y NMS. El puerto UDP 161 para los mensajes *query/response* y el puerto UDP 162 para los *traps*.

El uso de este protocolo de transporte trae consigo una menor carga de tráfico en la red y que las transmisiones sean rápidas; debido a que el protocolo UDP no establece una sesión entre el NMS y el agente.

Al hacer uso de este protocolo de transporte, el SNMP se encarga de disminuir las probabilidades de pérdida paquetes, mediante la implementación de temporizadores y retransmisiones para saber si un paquete, fue o no transmitido.

Lo ineficiente de este tipo de transmisión es que, en caso un agente envíe un *trap* y se pierda, este no será notificado al NMS.

ASN.1

Abstract Syntax Notation One (ASN.1) es un estándar de la ISO y la ITU-T que permite especificar cómo la información es representada y transmitida entre el NMS y el agente, dentro del contexto del SNMP. La ventaja de ASN.1 es la independencia de su sintaxis frente a los distintos dispositivos que lo utilicen.

SMI

El SMI, o Structure Management Information por sus siglas en inglés, define los objetos administrados, sus comportamientos y específica los tipos de datos asociados a ellos.

La definición de un objeto administrado constituye de 3 atributos principales:

- Nombre: Hace referencia al nombre, o object identifier (OID), del objeto. Este puede ser alfanumérico.
- Tipo y sintaxis: El tipo de dato de un objeto administrado es definida usando el ASN.1
- Codificación: El objeto es codificado y decodificado por el Basic Encoding Rules (BER) y de esta manera es transmitido sin problemas.

OID

Un objeto hace referencia a una variable propia del equipo. El valor del objeto está identificado por un OID (Object Identifier), que es el ID, al cual hace referencia el NMS al momento de enviar una consulta al agente.

La respuesta a este query, usando el OID, me indica el valor de la variable, a la cual se está apuntando, en ese momento.

Una característica importante de los objetos administrados es que estos se encuentran organizados en un árbol jerárquico [Ver figura 2.2]. Esta estructura es la básica que utiliza el SNMP para declarar a sus objetos. Un OID está representado por una serie de números enteros basados en los nodos del árbol, separados por (.) o también por nombres que representan a un objeto contenido en un nodo.

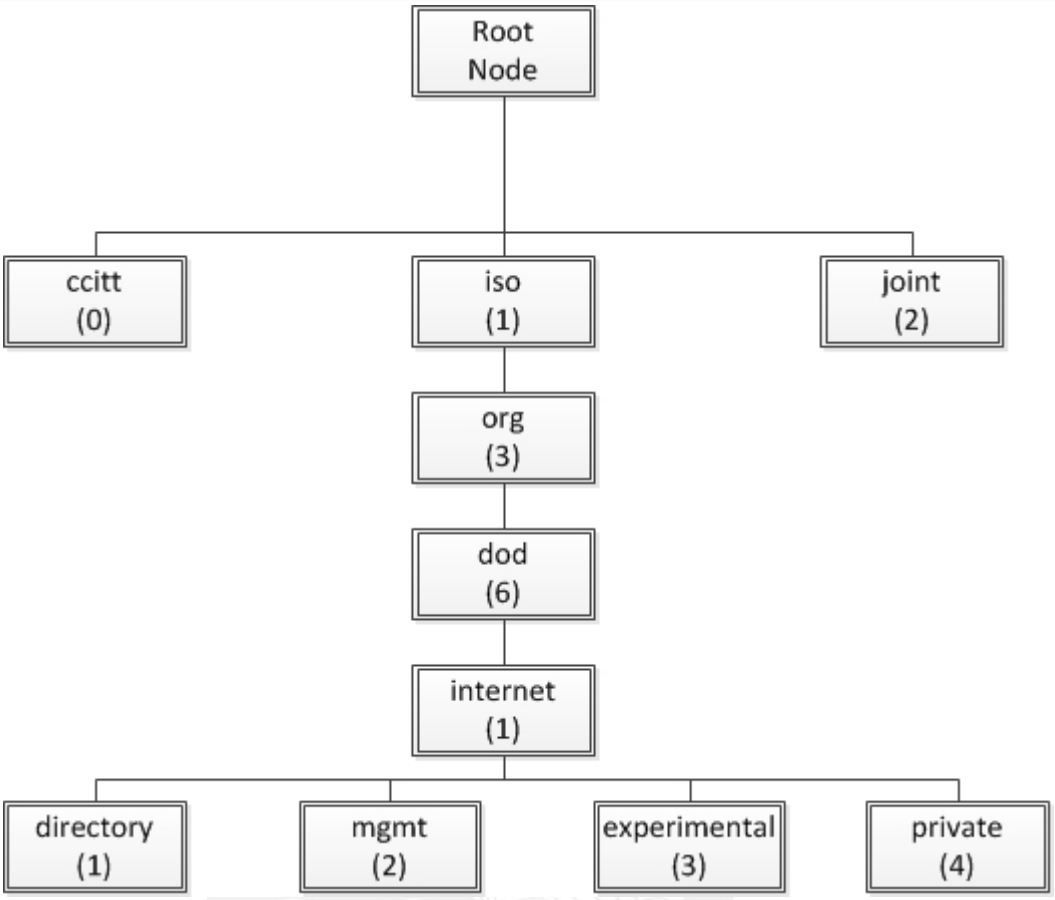

Figura 2.2: Estructura jerárquica de un objeto SMI

En la figura 2.2, se muestra un ejemplo de cómo se representa un objeto. En caso se quiera hacer referencia a un objeto dentro de la rama private. Se deberá utilizar el siguiente OID: 1.3.6.1.4.x, en donde “x” hace referencia al identificador del objeto.

MIB

Una MIB (Management Information Base, en sus siglas en inglés) es concebida como una base de datos jerarquizada de objetos; la cual es gestionada utilizando el SNMP y definida utilizando la SMI.

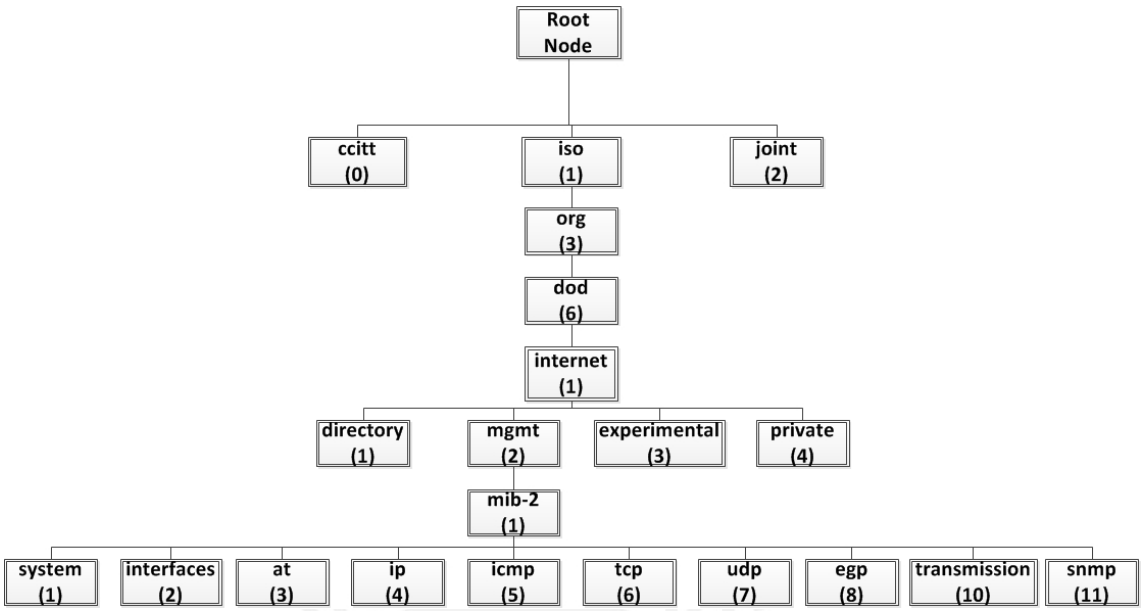

Figura 2.3: Estructura jerárquica de MIB-II

Existen extensiones de MIBs por cada entidad existente en una red que pueda ser gestionada bajo el protocolo SNMP. Así también, existen estructuras MIBs estándares; las cuales son implementadas por la mayoría de los fabricantes, ya que estas describen objetos básicos de un dispositivo a gestionar, como por ejemplo la MIB-II, como lo muestra figura 2.3; en la cual se puede apreciar los objetos de la MIB-II, los cuales se encuentran representados en forma de árbol.

2.4.2 Protocolos propietarios

2.4.2.1 NBAR

NBAR (Network Based Application Recognition) es una herramienta integrada en dispositivos del fabricante Cisco Systems tales como routers y switches; la cual se encarga de inspeccionar algunos paquetes de información de un flujo de datos que se transmiten; para determinar el protocolo y la aplicación que los produce a partir de los puertos de origen y destino de los paquetes.

La información extraída por NBAR es utilizada para aplicaciones tales como políticas de calidad de servicio, priorizando determinados flujos de datos y así mismo es usada para detectar malwares que traten de perjudicar el performance de la red. [1]

2.4.2.2 TL1

TL1 (Translation Language 1) es un protocolo de gestión creado por la empresa BellCore utilizada en redes SONET (Synchronous Optical Networking); la cual se encarga del formato de la información transmitida entre los elementos de red y el servidor de gestión. [7]

Los mensajes básicos que utiliza TL1 para el monitoreo de la red son:

- Mensajes *Input:* Es el mensaje enviado por el servidor de gestión hacia los elementos de red gestionados.
- Mensajes *Output/Response:* Respuesta enviada por los elementos de red gestionados en respuesta a los mensajes *Input.*
- Mensajes *Acknowledgment:* Utilizado como acuse de recibo a los mensajes *Output/Response.*
- Mensajes *Autonomous*: Mensaje enviado por el elemento de red gestionado a consecuencia de algún suceso inesperado ocurrido en el elemento de red.

Los mensajes presentados anteriormente, están compuestos por los siguientes elementos:

- Identificador del elemento y de la fuente (TID/SID por sus siglas en inglés): Nombre único asignado a cada elemento de red.
- Identificador de acceso (AID por sus siglas en inglés): Identificador de un elemento dentro de un elemento de red a gestionar.
- Etiquetas de correlación (CTAG/ATAG): Son números utilizados para correlacionar varios mensajes enviados al servidor de monitoreo.

2.5 Software de gestión

2.5.1 Software comercial

2.5.1.1 HP Openview

HP Openview es una línea de productos de la marca Hewlett Packard que consiste en la gestión de redes y sistemas. Este software, cuenta con una arquitectura modular pensada en satisfacer todas las necesidades que un operador pueda tener [8].

Dentro de la línea de productos se encuentra el Network Node Manager (NNM); el cual, mediante el uso del protocolo SNMP, realiza el monitoreo de los dispositivos de red; además de crear gráficas de la topología de la red a raíz del descubrimiento de los dispositivos a los cuales cuenta con acceso.

Esta solución cuenta con una licencia propietaria, lo cual implica el pago por el servicio que este ofrece. Lo cual implica que el costo de un proyecto aumente y en consecuencia, el margen de rentabilidad del proyecto también.

En la figura 2.4 se puede apreciar una gráfica de red y las características con la que esta cuenta.

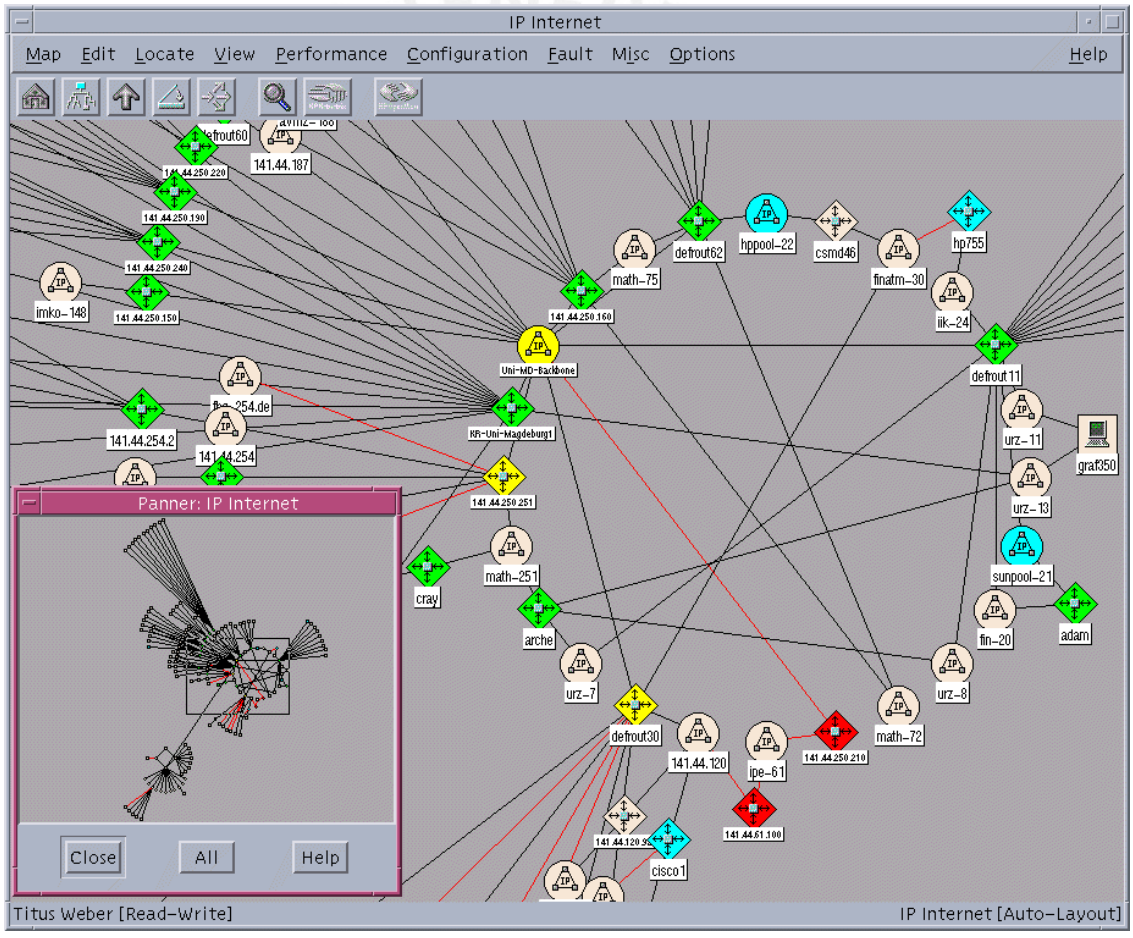

Figura 2.4: Ejemplo de topología de red creada en HP Openview [8]

2.5.1.2 iManager M2000

iManager M2000 es una plataforma de operación y mantenimiento exclusivo de la marca del fabricante de tecnologías, Huawei. Está plataforma provee a los operadores de redes móviles, la gestión centralizada de los equipos y realiza funciones básicas, tales como la gestión de configuración, desempeño, fallas, seguridad, log's, topologías, software y sistemas. Utilizando un protocolo propietario de gestión.

Este servidor está conformado por dos tipos de software: El software principal y el software de mediación. El software principal se encarga de las funciones del sistema en general; y el software de mediación es utilizado para la adaptación del módulo a distintos elementos de red propios del fabricante. [9]

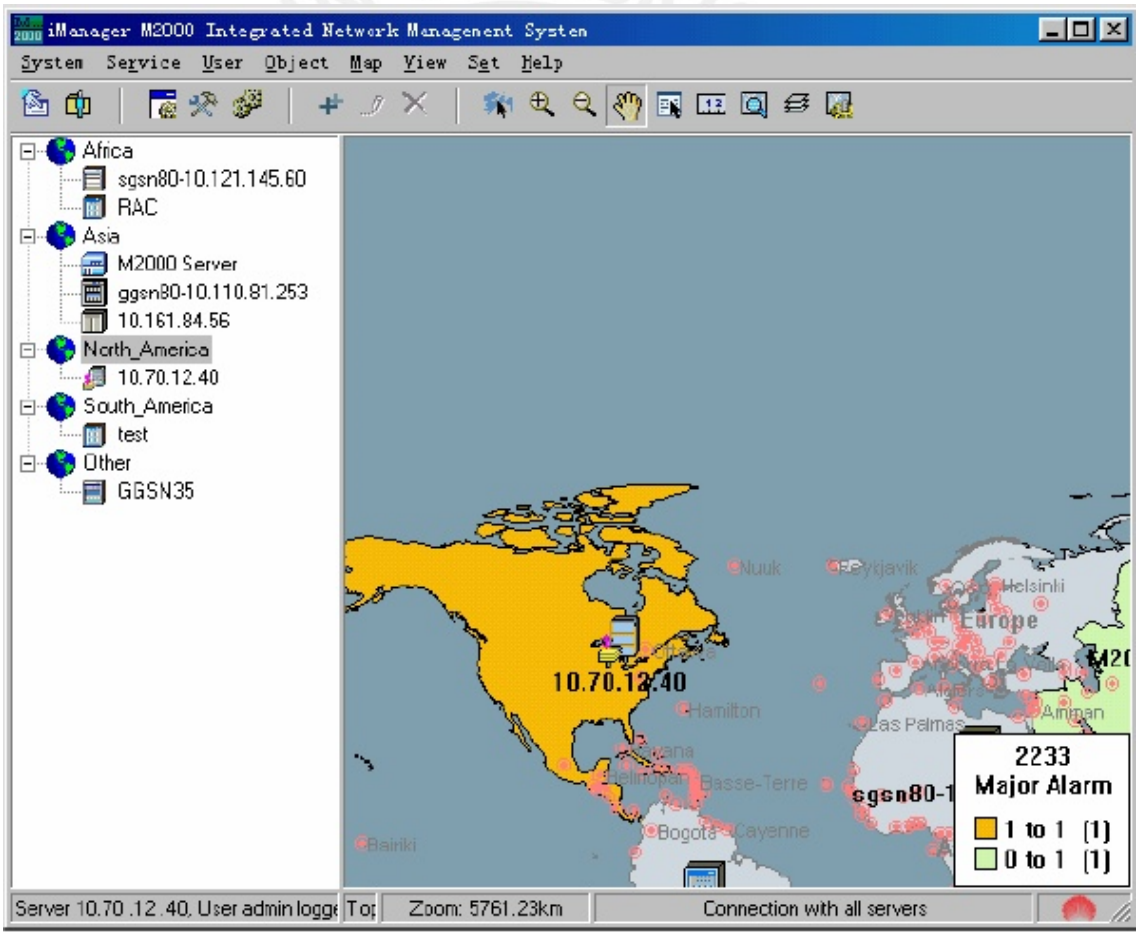

Figura 2.5: Ejemplo de topología de red creada en iManager M2000 [9]

2.5.2 Software libre

2.5.2.1 Cacti

Es una interfaz web, escrita en PHP, diseñada para hacer uso de RRDtools para las funciones de almacenamiento de datos y gráficos con ayuda de MySQL, entro otros programas.

Cacti ofrece la posibilidad de utilizar scritps externos para realizar el monitoreo de equipos; así obtener cualquier tipo de información y crear plantillas de datos que se adecuen a las necesidades del usuario [10].

Cuenta con una arquitectura de plugings las cuales permiten agregar nuevas funcionalidades; además permite que un usuario administrador, pueda agregar nuevos usuarios con distintos privilegios así como restringir ciertas áreas del Cacti.

La interfaz gráfica del gestor se puede apreciar en la figura 2.6.

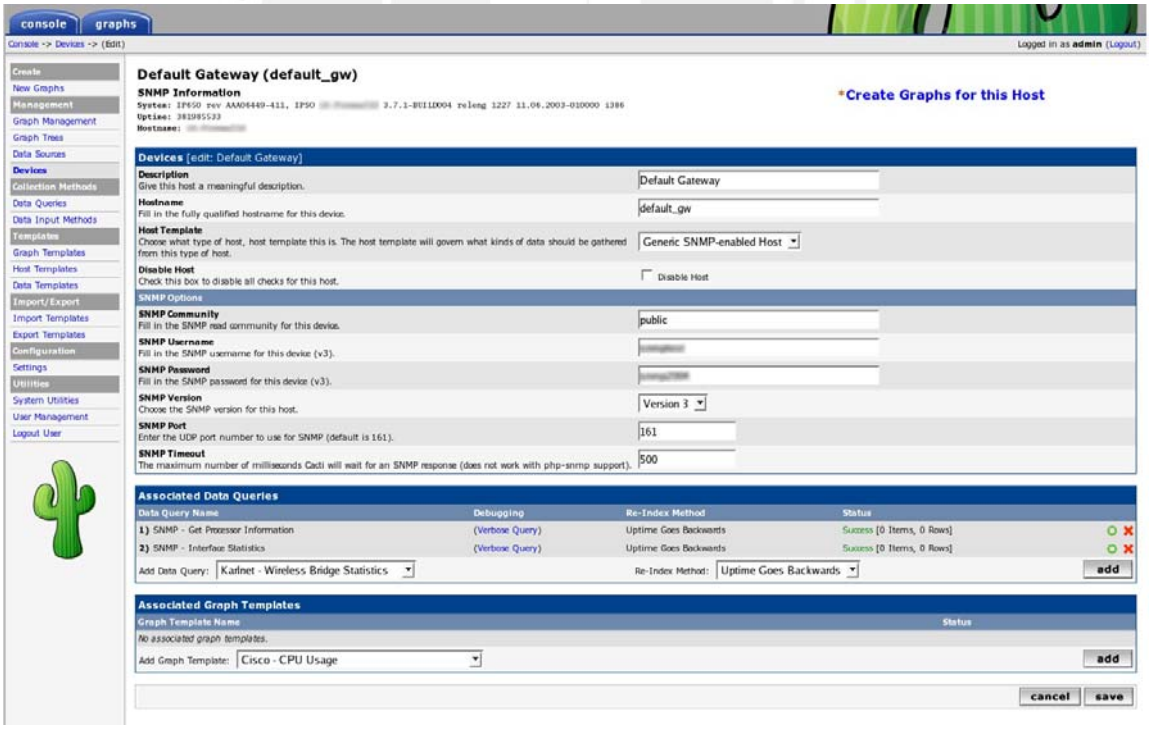

Figura 2.6: Interfaz principal de Cacti

2.5.2.2 NAGIOS

Nagios trabaja en un servidor como un *daemon* (o servicio) y utiliza *plugins* en el mismo servidor o de forma externa; lo cual facilita la creación de plantillas que permiten al usuario ordenar la información de la red de forma personalizada. Originalmente se diseñó para funcionar únicamente en Linux, pero ahora puede trabajar en distintas variantes Unix [11].

La interfaz gráfica del gestor se puede apreciar en la figura 2.7.

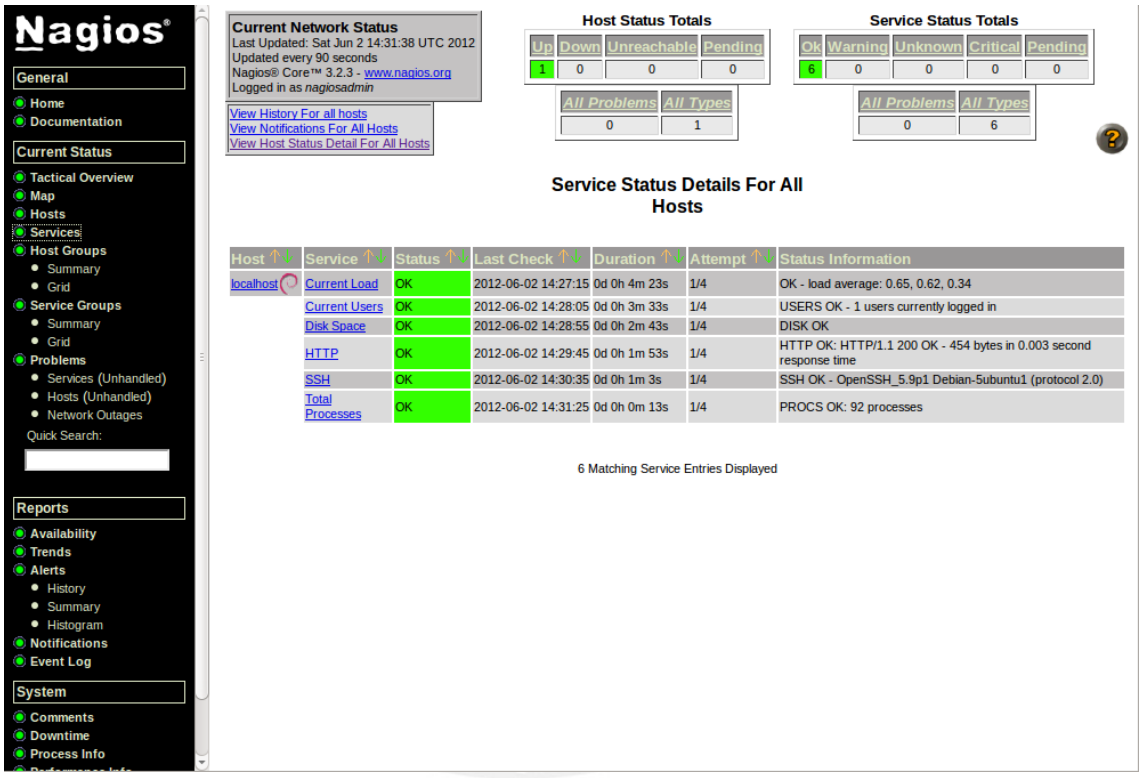

Figura 2.7: Interfaz de monitoreo de entidades de Nagios versión 3.2.3

2.5.2.3 OpenNMS

OpenNMS está escrito en Java, y por esta razón solo puede implementarse en plataformas que soporten Java SDK versión 1.6 o mayor [12].

Este Software puede ser instalado en la mayoría de las distribuciones de Linux, así también como en Windows, Solaris y OS X. Además requiere de la base datos

PostgreSQL para su correcto funcionamiento. Al igual que la mayoría de los softwares de gestión, este utiliza el SNMP para recibir y solicitar información de los dispositivos administrados.

La interfaz gráfica del gestor se puede apreciar en la figura 2.8.

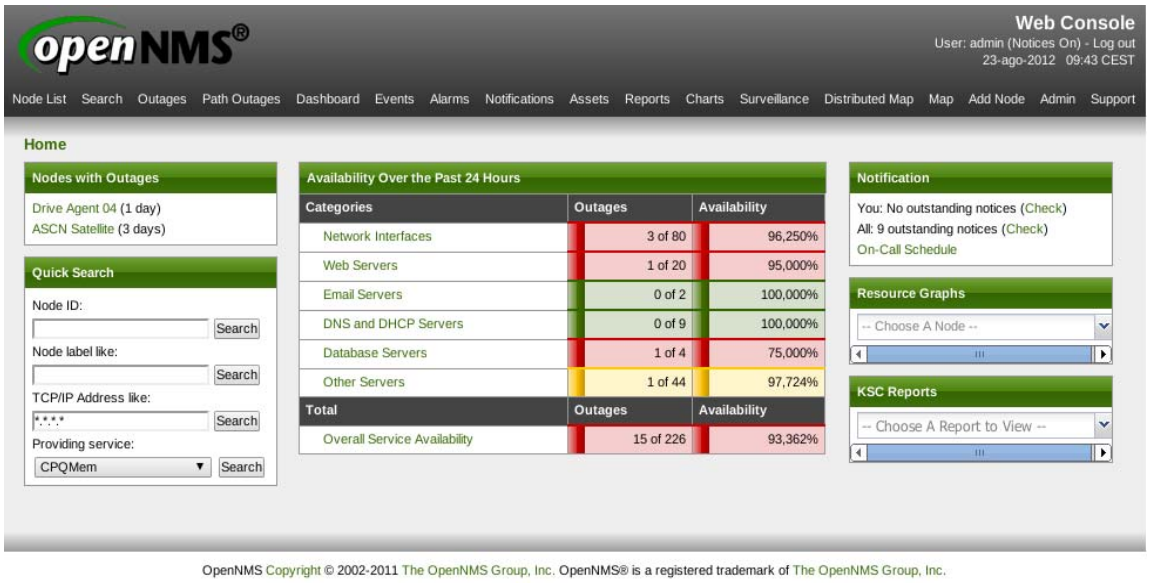

Figura 2.8: Interfaz principal de OpenNMS

Capítulo 3

Diseño e implementación en la plataforma de gestión

3.1 Sistema a implementar

El diseño fue realizado pensando en la problemática que se presenta en el operador móvil, la cual radica en la extracción de información relevante de los equipos utilizados dentro de la red sin importar el fabricante. De esta manera, se asegura el correcto funcionamiento de los equipos; así también, constatar el buen servicio que se brinda a los clientes garantizando los suficientes recursos de red o prevenir algún suceso que afecte el servicio.

Asimismo, se contará con una solución flexible y dinámica para facilitar el trabajo de los operadores de red. Finalmente, el sistema debe converger con la actual solución de software libre que cuenta implementada el operador: La plataforma de gestión Cacti versión 0.8.7.h.

Cabe recalcar que además de la problemática presentada en el operador móvil, el diseño también se demostró mediante el desarrollo de pruebas locales, en un ambiente

contralado, haciendo uso del simulador de entornos de red GNS3; el cual se encuentran bajo licencia libre.

3.2 Herramientas a utilizar

Las siguientes herramientas fueron empleadas para el desarrollo de la presente implementación:

- Sistema operativo GNU/LINUX Ubuntu Server 11.10 de 64 bits (Kernel 2.6.32-28-server): Sistema operativo basado en Debian, compuesto por software distribuido bajo una licencia libre y de código abierto.

- Cacti versión 0.8.7h: Gestor de código abierto que realiza gráficos, estadísticas mediante el uso de RRDTool's, almacenamiento de datos y administración de la red. Sé utilizó esta versión debido a que se encuentra implementada actualmente en el operador móvil y fue la última versión en el momento de su implementación.

- Lenguaje C++: Lenguaje de programación utilizado para la conexión a los equipos, extracción y filtrado de datos. Se ha escogido este lenguaje para realización de scripts de pruebas porque mantiene la rapidez en cuanto a performance y tratamiento de los datos.

- XML: Lenguaje de etiquetas que está diseñada para transportar y almacenar información; además permite expresar las diversas operaciones y mensajes de gestión.

- Bash: Es el intérprete por defecto en la mayoría de sistemas GNU/Linux cuya función es precisamente, interpretar órdenes de forma rápida y flexible.

- Expect: Herramienta utilizada para automatizar la interacción con programas que cuentan con una interfaz terminal.

3.3 Planteamiento del sistema

El sistema actual de monitoreo del operador móvil es Cacti, el cual se encuentra basado en software libre; además provee gran facilidad y dinamismo en cuanto al manejo gráficos y presentación de la información extraída de los equipos de la red.

Este sistema de monitoreo permite extraer los datos de los equipos utilizando NET-SNMP [13] y también mediante el desarrollo de scripts. Los datos obtenido por Cacti son almacenados en archivos de base de datos cíclicas (RRA), controladas con el RRDTools; el cual también tiene como tarea de generar las gráficas de los datos extraídos de los equipos.

Cacti hace uso de una base de datos relacional llamada MySQl como se muestra en la figura 3.1; en donde se almacenan la mayor parte del sistema y de los parámetros de configuración. Además, en esta base de datos también se almacenan los logs que son exportados por los dispositivos de la red y del mismo sistema.

Los manejos de los plugins por separado se dan gracias al Plugin Architecture; el cual ya viene incluido como parte de la arquitectura de Cacti desde la versión 0.8.8 y; además, permite que se integren las distintas funcionalidades que se encuentran en interacción con el Cacti.

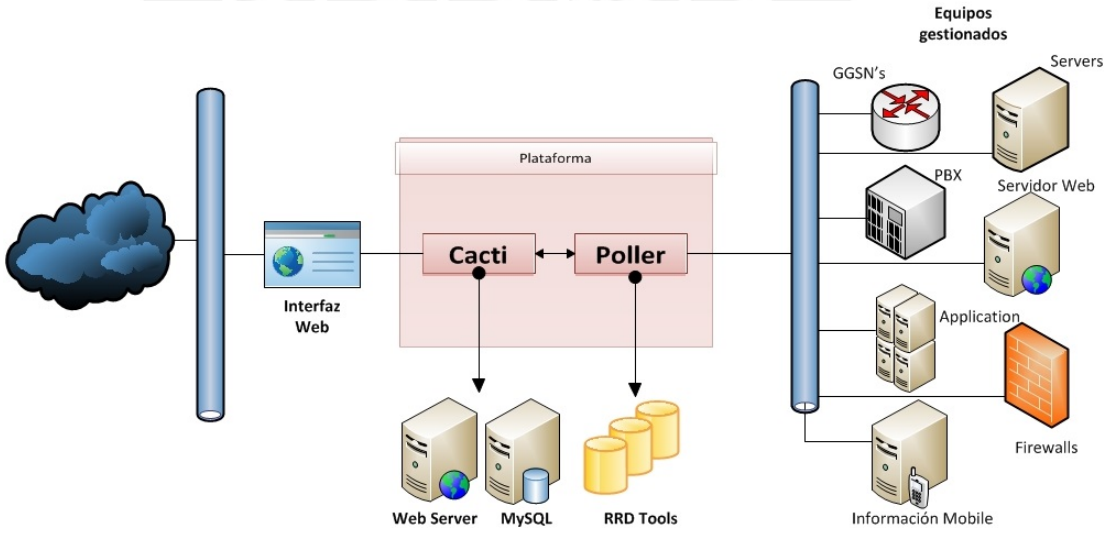

Figura 3.1: Arquitectura de Cacti [10]

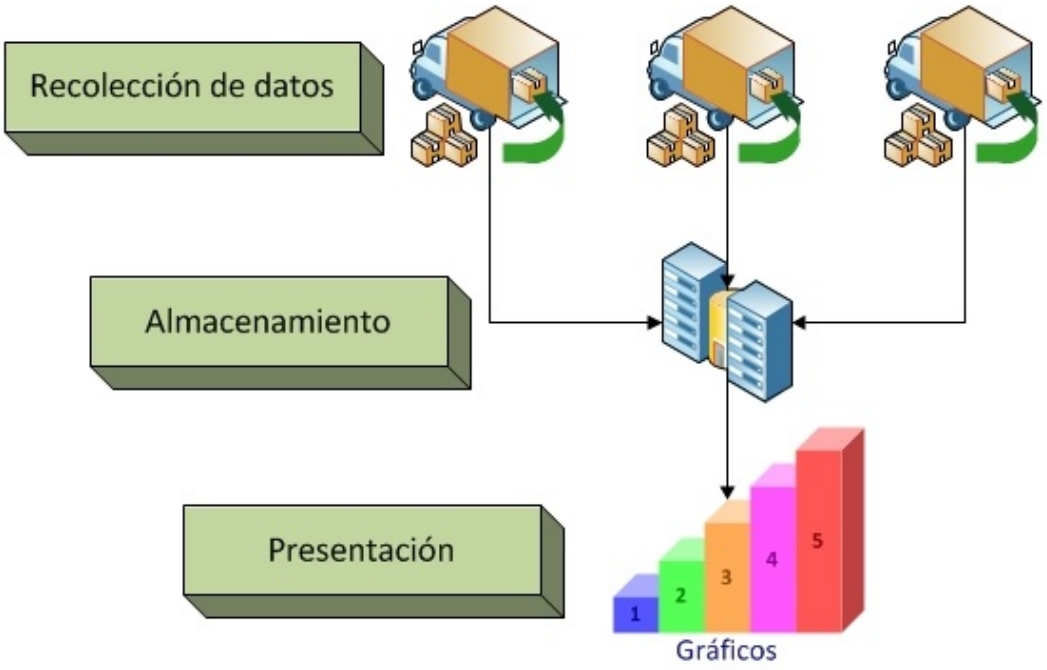

Figura 3.2: Etapas de recolección de datos de Cacti

En esta tesis se extenderá el funcionamiento del actual gestor de redes, que se encuentra en uso por el operador móvil, esto se desarrollará mediante el uso de scripts que realizan la conexión remota a los equipos de red de forma automatizada para, posteriormente, filtrar la información extraída y darle el mismo formato de respuesta que el protocolo snmp. Este protocolo, SNMP, es utilizado por la plataforma de gestión del operador móvil; el cual cuenta con una red lógica como lo muestra las figuras 3.3 y 3.4.

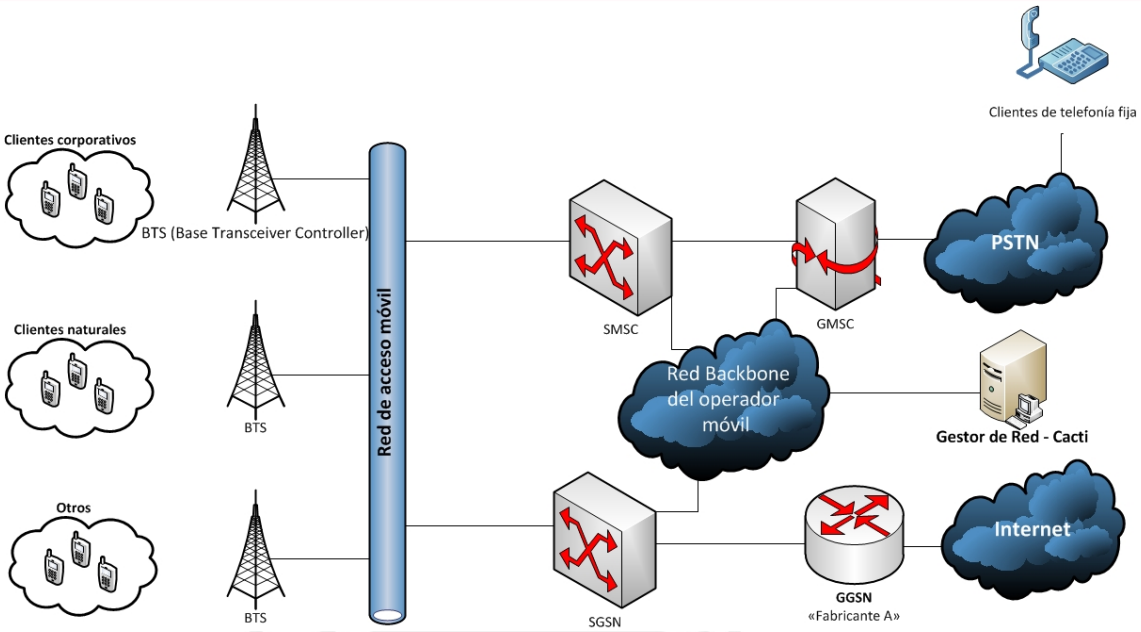

Figura 3.3: Visión general del sistema de red del operador móvil

A continuación, el esquema específico de la solución a implementar en el sistema de gestión del operador móvil.

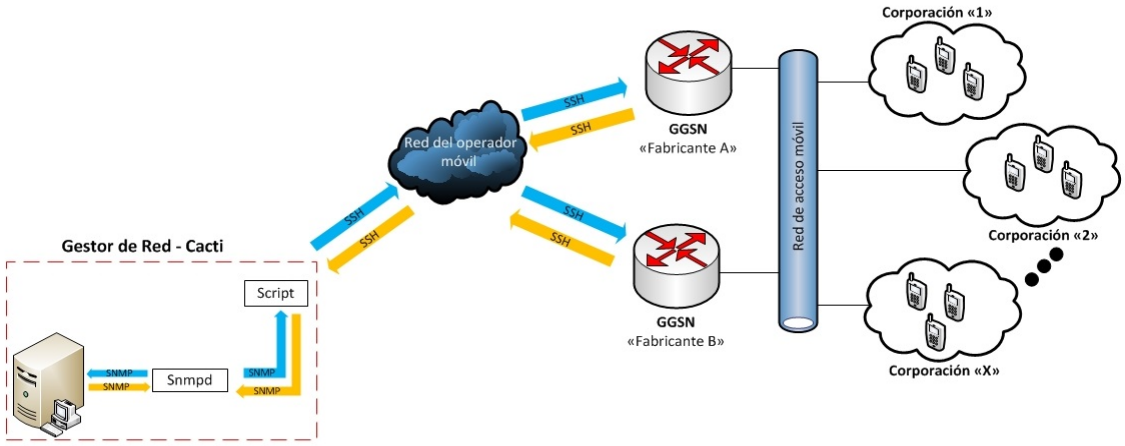

Figura 3.4: Esquema del sistema de monitoreo del GGSN de Huawei

3.4 Configuración en el servidor

Dado que en el operador móvil se están reemplazando los equipos GGSN, los cuales se encargan de tareas tales como asignar parámetros de red, como la IP a los dispositivos móviles de los clientes, se ha notado que estos equipos cuentan con una

implementación básica del protocolo SNMP; lo cual implica la necesidad de uso de un software de gestión propietario para el monitoreo de valores críticos importantes para los operadores. Por esta razón, se necesitará adecuar el sistema de Cacti para extraer la información de los equipos requerida por los operadores de la red como estadísticas, cantidad de recursos usados, entre otros; e introducirlos al sistema de monitoreo actualmente utilizado.

Para todo esto se decidió definir una MIB e implementar los scripts, con ayuda del lenguaje de programación en C++, que permitan obtener los parámetros que se necesiten.

En esta sección se explicarán los detalles de la extracción de un parámetro en particular extraído del GGSN de la marca Huawei; llamado *All GTP PDP context* el cual indica la cantidad de usuarios con sesiones activas y listas para iniciar la navegación en internet desde sus dispositivos móviles.

Además, también, en los anexos A y B se muestra la extracción de parámetros de equipos de red de marcas distintas; cuya implementación se realizó en un ambiente de laboratorio.

3.4.1 Definición de una MIB

El objetivo primordial es obtener indicadores de la calidad de la red y cantidad de recursos disponibles o utilizados de los equipos que actualmente se encuentran en uso dentro de la red de packet core y red móvil que no pueden ser monitoreados usando SNMP.

En este caso se explicará los pasos a seguir para extraer el parámetro *All GTP PDP context* del GGSN de la marca Huawei, por lo que se definió una MIB para obtener los parámetros a monitorear de este equipo.

La MIB proveerá la estructura jerárquica y los identificadores (OIDs) que utilizaremos. La MIB que se describe a continuación ha sido definida dentro de la rama de enterprises y sigue la estructura de la figura 3.5.

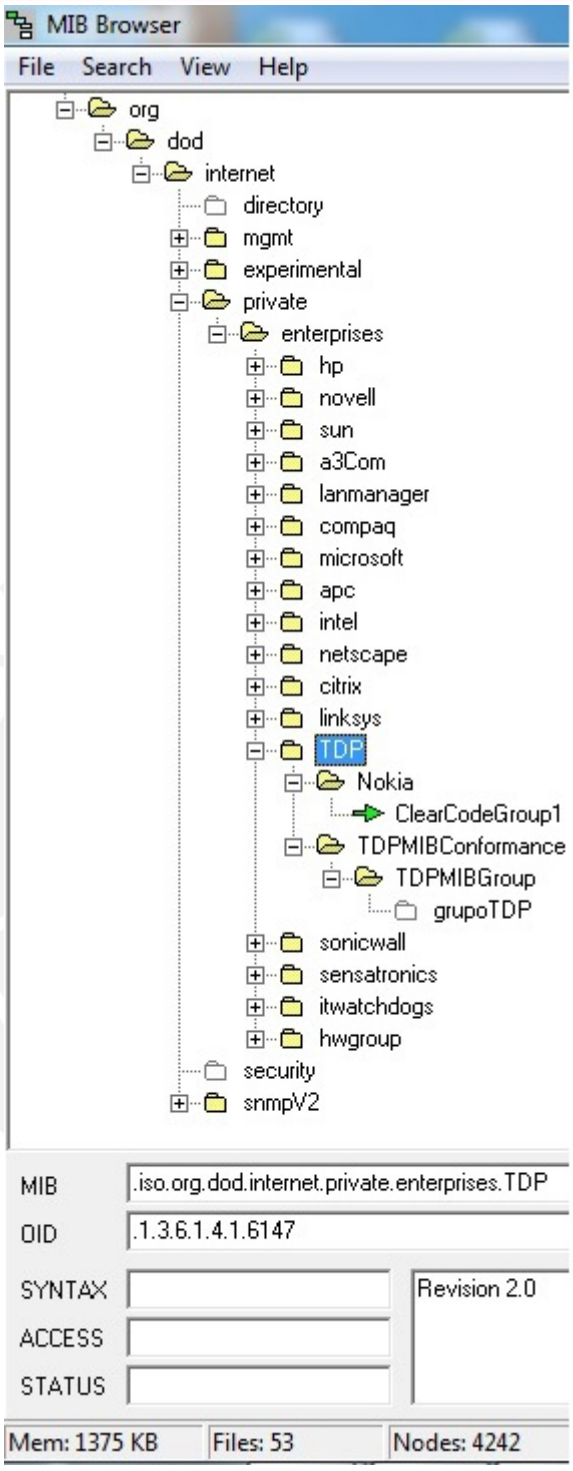

Figura 3.5: Estructura de árbol de la MIB que se creó para extraer el pdp context

Para esto, la definición utilizada fue la que se muestra a continuación:

Tabla 3.1: Definición de MIB para los equipos GGSN de la marca Huawei

```
TDP-MIB DEFINITIONS ::= BEGIN
IMPORTS
        MODULE-IDENTITY, OBJECT-TYPE, Integer32, enterprises
                FROM SNMPv2-SMI
        OBJECT-GROUP    FROM SNMPv2-CONF

TDP MODULE-IDENTITY
        LAST-UPDATE     "20120824" - 24/ago/2012
        ORGANIZATION    "OPERADOR MOVIL"
        CONTACT-INFO    "Authors: Hernan Romano / Antonio Ocampo
                 Email: h.romanoc@pucp.edu.pe / aocampo@pucp.edu.pe"
        DESCRIPTION     "MIB para gestionar los equipos que carecen de SNMP"
        REVISION        "20120824" -- 24/ago/2012
        DESCRIPTION     "Revision 2.0"
        ::= {enterprises  6147}

Nokia               OBJECT IDENTIFIER ::= {TDP 1}
TDPMIBConformance       OBJECT IDENTIFIER ::= {TDP 2}

ClearCodeGroup1 OBJECT-TYPE
            SYNTAX Integer32
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION "Clear Code Group 1"
            ::= {Nokia 1}

TDPMIBGroup     OBJECT IDENTIFIER
            ::= { TDPMIBConformance 1 }

grupoTDP        OBJECT-GROUP
        OBJECTS {
            ClearCodeGroup1
        }
```

```
    STATUS current
    DESCRIPTION "Objetos para el monitoreo de los equipos que carecen de SNMP"
    ::= { TDPMIBGroup 1 }
END
```

3.4.2 Herramientas de verificación de la MIB

Luego de realizar la definición de la MIB, se necesitará verificar su correcto funcionamiento, lo cual implica utilizar las siguientes herramientas de verificación:

3.4.2.1 SNMPTRANSLATE

Esta herramienta puede realizar la traducción numérica o textual de los objetos de una MIB y además puede buscar la definición de un objeto dentro de un archivo MIB, sin necesidad de hacer uso del nombre del host o de los parámetros de la comunidad.

La sintaxis del snmptranslate es la siguiente: *Snmptranslate options objectID*

Ahora se mostrará algunas formas de asegurarse que la MIB se encuentra correctamente traducida dentro del servidor. Como se puede ver en la Figura 3.6

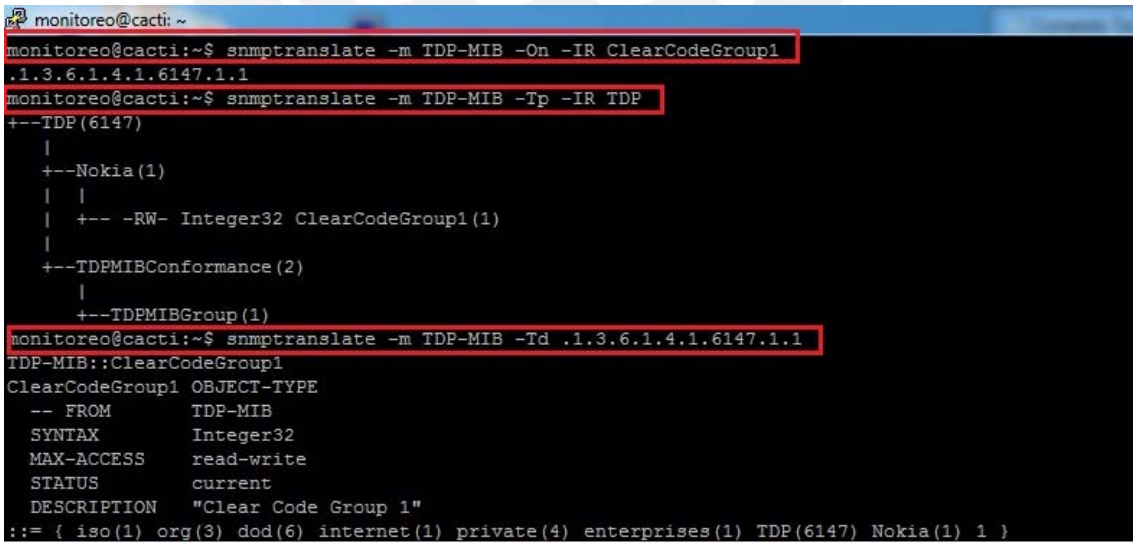

Figura 3.6: Verificación de la MIB mediante SNMPTRANSLATE

3.4.2.2 SNMPSET

Este comando es usado para variar el valor del objeto administrado. En este sentido, cualquier objeto dentro de una MIB definido como “read-write” o “read-only” puede ser alterado. Un aspecto importante de este comando es que el NMS puede lograr asignar varios valores a distintos objetos a la vez.

La sintaxis del snmpset es la siguiente: *Snmpset options hostname objectID type value*

Esta herramienta se utilizó al inicio de la implementación para constatar que el sistema está reconociendo correctamente los tipos de variables y lo valores sean los correctos.

Como se muestra en la figura 3.7.

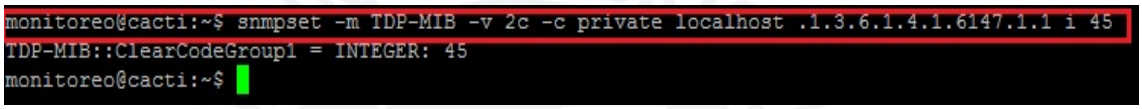

Figura 3.7: Verificación de la MIB mediante el comando SNMPSET

3.4.2.3 SNMPGET

El comando snmpget es enviada por el NMS hacia el agente para obtener el valor de un objeto del dispositivo. El agente lo recepciona y procesa la solicitud. Algo importante que recalcar, es que al igual que el comando snmpset, se puede obtener valores de distintos objetos a la misma vez.

La sintaxis del snmpget es la siguiente: *Snmpget options hostname objectID*

En la actual implementación, debido a que los dispositivos a monitorear, cuentan con una implementación básica de SNMP. Lo que se realizó fue la creación de scripts que permitan filtrar los datos para obtener el valor requerido por los operadores de la red; el cual se explicará en la sección 3.4.3.

En la figura 3.8 se muestra los resultados obtenidos de los comandos ejecutados para obtener el valor real del parámetro *All GTP PDP context* de los equipos GGSN de Huawei.

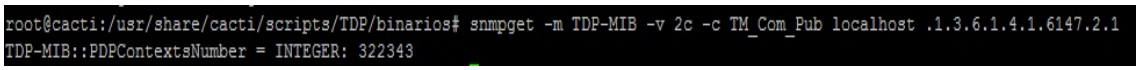

Figura 3.8: Salida del comando SNMPGET

3.4.3 Extracción de datos y filtrado

Una vez verificado el correcto funcionamiento de la MIB, se pasa a extraer la información necesaria de aquellos equipos que solo cuentan con una implementación básica del protocolo SNMP.

Luego de que el OID es traducido, será necesario relacionarlo con los scripts correspondientes. Para ello, en el servidor, que aloja al gestor de red Cacti, se procedió a modificar el archivo de configuración del proceso SNMP. De la siguiente manera:

Tabla 3.2: Ejemplo de modificación en el archivo de configuración snmpd

```
pass .1.3.6.1.4.1.6147.1.1 ./usr/share/cacti/scripts/TDP/binarios/GGSN-PDP-Contexts.sh
```

Está línea indica que cada vez que el Cacti realice un query apuntando a este OID se ejecutará un script; el cual se muestra en la tabla 3.3 que se encargará de conectarse al equipo GGSN vía SSH, ejecutará un comando dentro del equipo, posteriormente se filtrará el resultado y finalmente se devolverá el resultado al proceso SNMP con el formato esperado.

Tabla 3.3: Archivo Expect que se encargó de la conexión SSH y almacenamiento de la información del equipo

```
#!/usr/bin/expect –f

set usuario "cacti"
set equipo "10.10.x.x"
set clave "password"
set timeout -1
set prompt "<GGSN>"
match_max 100000
log_file -a ./GGSN-PDP-Contexts.log

spawn ssh -oStrictHostKeyChecking=no -oCheckHostIP=no $usuario@$equipo
expect "*?assword:*"
send -- "$clave\r"
```

```
expect -exact "$prompt"
send -- "display pdp-number\r"

expect -exact "$prompt"
send -- "quit\r"
expect eof
```

Tabla 3.4: Bash que ejecuta el expect y filtra los datos

```
#!/bin/bash

obtener_PDP_Contexts=$(pwd)"/GGSN-PDP-Contexts.expect_e"
archivo=$(pwd)"/GGSN-PDP-Contexts.log" /*alamacenamos la ruta exacta del archivo
que almacenó los datos recibidos del equipo monitoreado*/
CAT=$(which cat)
RM=$(which rm)
EXPECT=$(which expect)

test=$($obtener_PDP_Contexts)
valor=$($CAT $archivo | grep "ALL GTP PDP context" | awk '{print $6}')
$RM $archivo
echo ".1.3.6.1.4.1.6147.2.1"
echo "Integer32"
echo $valor
```

Para la presente solución se creó un script que se encarga de acceder vía SSH al equipo GGSN-LV02 con el usuario y contraseña del administrador; de esta manera se procederá a ejecutar un comando en donde podamos obtener la cantidad de usuarios con sesiones activas.

La ejecución del comando dará como resultado una serie de valores como se muestra en la figura 3.9; así que dentro del script se filtrará el resultado utilizando expresiones regulares hasta obtener el dato deseado.

```
monitoreo@cacti:/usr/share/cacti/scripts/TDP/binarios$ sudo ./GGSN-PDP-Contexts
spawn ssh -oStrictHostKeyChecking=no -oCheckHostIP=no cacti@10.10.222.176
cacti@10.10.222.176's password:

*******************************************************
*          All rights reserved (2000-2010)            *
*       Without the owner's prior written consent,    *
* no decompiling or reverse-engineering shall be allowed. *
* Notice:                                             *
*      This is a private communication system.        *
*   Unauthorized access or use may lead to prosecution.   *
*******************************************************

Info: The max number of VTY users is 10, and the number
      of current VTY users on line is 1.
<GGSN-LV02>display pdp-number
PDP context(s) Number Information:
---------------
Slot 7  CPU 0  GTP v0 PDP context(s)  =  5
Slot 7  CPU 0  GTP v1 PDP context(s)  =  158179
Slot 7  CPU 1  GTP v0 PDP context(s)  =  3
Slot 7  CPU 1  GTP v1 PDP context(s)  =  158082
      ALL GTP PDP context(s)  =  316269
```

Figura 3.9: Resultado del comando “Display pdp-number” ingresado en el GGSN de la marca Huawei

```
monitoreo@cacti:/usr/share/cacti/scripts/TDP/binarios$ cat log.out | grep "ALL GT
P PDP context" | awk '{print $6}'
317964
```

Figura 3.10: Filtrado de datos utilizando Expresiones regulares

Luego de obtener el valor deseado, se procede a devolver el OID y el tipo de dato que en el formato esperado. Como se muestra en la figura 3.11.

```
monitoreo@cacti:/usr/share/cacti/scripts/TDP/binarios$ ./GGSN-PDP-Contexts
.1.3.6.1.4.1.6147.2.1
Integer32
322343
```

Figura 3.11: Resultado del Script

Como se mencionó en el punto 3.1 y 3.4, al igual que la extracción del parámetro *ALL PDP-CONTEXT* en el GGSN del operador móvil, también se realizaron pruebas en un entorno de laboratorio, como lo muestran los anexos A y B, las cuales permitieron el monitoreo de la cantidad de usuarios Telnet conectados a un router y además se extrajo información sobre los procesos activos del equipo; los cuales fueron mostrados de forma dinámica, en un ranking en el plugin weathermaps, instalado en el gestor Cacti.

Estos resultados se mostrarán en el capítulo 4.

Capítulo 4

Resultados

4.1 Gráficas en Cacti

Luego de realizar la optimización del gestor, se logró monitorear el equipo GGSN-LV02 haciendo uso del protocolo SNMP. Ahora, el servidor podrá realizar solicitudes SNMP con la herramienta snmpget; la cual devolverá el valor del parámetro *All GTP PDP context* y se integrará al sistema de Cacti haciendo uso de archivos XML como lo muestra la tabla 4.1. Y, asimismo se logró la extracción de la cantidad de sesiones Telnet activas y el porcentaje de procesamiento de cada uno de los procesos presentes en un router; cuyas implementaciones se encontraran en el anexo A y B.

4.1.1 Monitoreo del GGSN del operador móvil

Luego de realizar la adaptación, se logró el monitoreo del GGSN mediante el uso del protocolo SNMP e integrarlo al sistema de gestión del operador móvil de forma transparente. Para una mejor visualización de los resultados se creó una plantilla de datos basada en el lenguaje XML, como se muestra en la tabla 4.1.

Tabla 4.1: Archivo XML creado para la creación de tabla PDP Context en el Cacti

```
<interface>
<name>Get PDP Context</name>
 <oid_index>.1.3.6.1.4.1.6147.2.1</oid_index>
<oid_index_parse>OID/REGEXP:.*\.([0-9]{1,16})$</oid_index_parse>
<index_order>Index</index_order>
<index_order_type>numeric</index_order_type>
<index_title_format>|chosen_order_field|</index_title_format>
<fields>
<Index>
<name>Index</name>
<method>walk</method>
<source>index</source>
<direction>input</direction>
</Index>
<Tag>
<name>PDP Context Value</name>
<method>walk</method>
<source>value</source>
<direction>input</direction>
<oid>.1.3.6.1.4.1.6147.2.1</oid>
</Tag>
</fields>
</interface>
```

Una vez importado el archivo XML, se integró a la plataforma y se obtuvo el resultado de la figura 4.1.

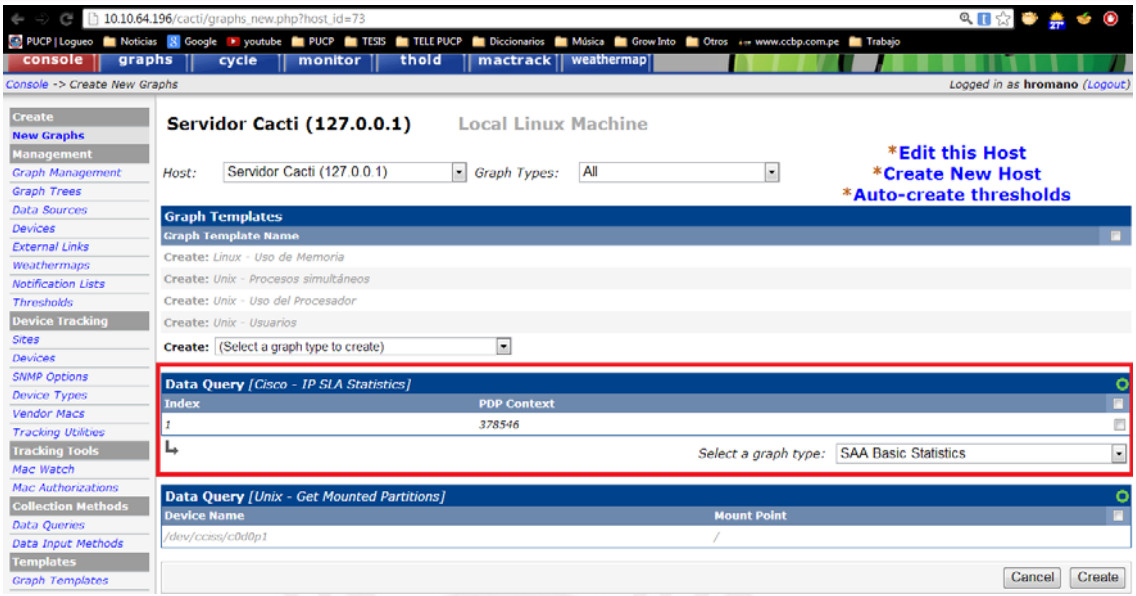

Figura 4.1: Tabla XML del parámetro All GTP PDP context

Finalmente se mostrará en un gráfico los valores obtenidos periódicamente como se encuentra en la figura 4.2.

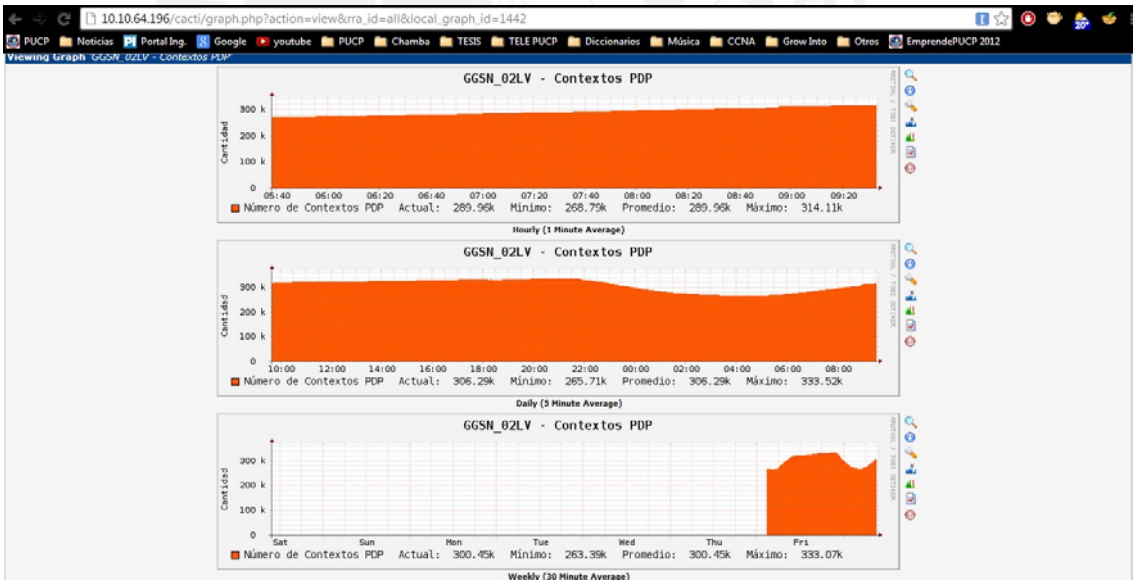

Figura 4.2: Gráfica del parámetro All GTP PDP context del equipo GGSN de la marca Huawei

4.1.2 Monitoreo de las sesiones Telnet activas

Luego de realizar la adaptación, siguiendo el procedimiento presentado en el anexo A, se logró el monitoreo de las sesiones Telnet activas de un router cisco vía el protocolo SNMP e integrarlo al sistema de gestión Cacti de forma transparente.

Finalmente se mostrará en un gráfico los valores obtenidos periódicamente como se encuentra en la figura 4.3.

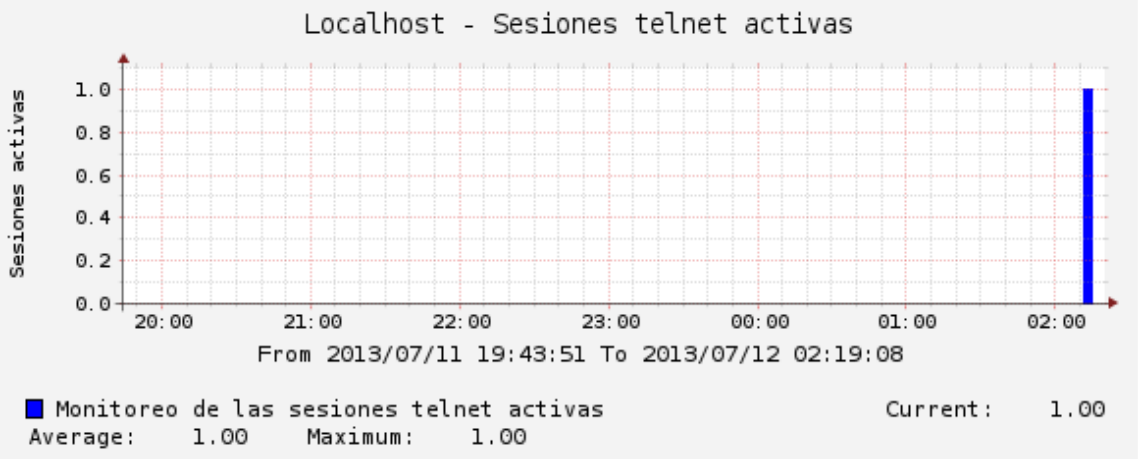

Figura 4.3: Gráfica de la cantidad de sesiones Telnet activas de un router de la marca Cisco

4.1.3 Monitoreo de procesos de un router

Luego de realizar la adaptación, siguiendo el procedimiento presentado en el anexo B, se logró el monitoreo de los procesos ejecutándose en un router de la marca Cisco Systems mediante el uso del protocolo SNMP e integrarlo al sistema de gestión del operador móvil de forma transparente. Para una mejor visualización de los resultados se creó una plantilla de datos basada en el lenguaje XML, como se muestra en la tabla 4.2.

Tabla 4.2: Archivo XML creado para la creación de la tabla procesos

```
<interface>
  <name>Get SNMP Interfaces</name>
  <description>
  Solicitud de informacion de procesos
  </description>
```

```
<oid_index>.1.3.6.1.4.1.8072.2.7000.1.1</oid_index>
<index_order>porcentajein:nombreProceso:index</index_order>
<index_order_type>numeric</index_order_type>
<index_title_format>|chosen_order_field|</index_title_format>
<fields>
 <index>
  <name>ID del proceso</name>
  <method>walk</method>
  <source>value</source>
  <direction>input</direction>
  <oid>.1.3.6.1.4.1.8072.2.7000.1.1</oid>
 </index>
 <nombreProceso>
  <name>Nombre del proceso</name>
  <method>walk</method>
  <source>value</source>
  <direction>input</direction>
  <oid>.1.3.6.1.4.1.8072.2.7000.1.3</oid>
 </nombreProceso>
 <porcentajein>
  <name>Porcentaje de procesamiento (%)</name>
  <method>walk</method>
  <direction>input</direction>
  <source>value</source>
  <oid>.1.3.6.1.4.1.8072.2.7000.1.5</oid>
 </porcentajein>
 <porcentaje>
  <name>Porcentaje de procesamiento (%)</name>
  <method>walk</method>
  <source>value</source>
  <direction>output</direction>
  <oid>.1.3.6.1.4.1.8072.2.7000.1.2</oid>
 </porcentaje>
</fields>
```

```
</interface>
```

Una vez importado el archivo XML, se integró a la plataforma y se obtuvo el resultado de la figura 4.4.

Data Query [procesos]

<< Previous Showing Rows 1 to 30 of 108 [**1**,2,3,4] **Next >>**

ID del proceso	Nombre del proceso	Porcentaje de procesamiento (%)
1	Chunk Manager	0.00%
2	Load Meter	0.00%
3	CEF Scanner	0.00%
4	Check heaps	0.00%
5	Pool Manager	0.00%
6	Timers	2.54%
7	Serial Backgroun	0.00%
8	Environmental mo	0.00%
9	HC Counter Timer	2.34%
10	ATM Idle Timer	0.00%
11	ARP Input	0.00%
12	AAA high-capacit	0.00%
13	AAA_SERVER_DEADT	4.01%
14	Policy Manager	0.00%
15	Crash writer	0.00%
16	DDR Timers	0.00%
17	RO Notify Timers	1.10%
18	Entity MIB API	0.00%
19	SERIAL Adetect	0.00%
20	SMART	0.00%
21	GraphIt	0.00%
22	Dialer event	0.00%
23	XML Proxy Client	0.00%

Figura 4.4: Tabla XML de los procesos ejecutándose en un router Cisco

Además, se mostrará en un gráfico los valores obtenidos periódicamente como se encuentra en la figura 4.5; donde se observa que el nivel de procesamiento varía en el tiempo y además los valores de procesamiento llegan a ser cero debido a que el proceso no se está llevando a cabo en algunos momentos.

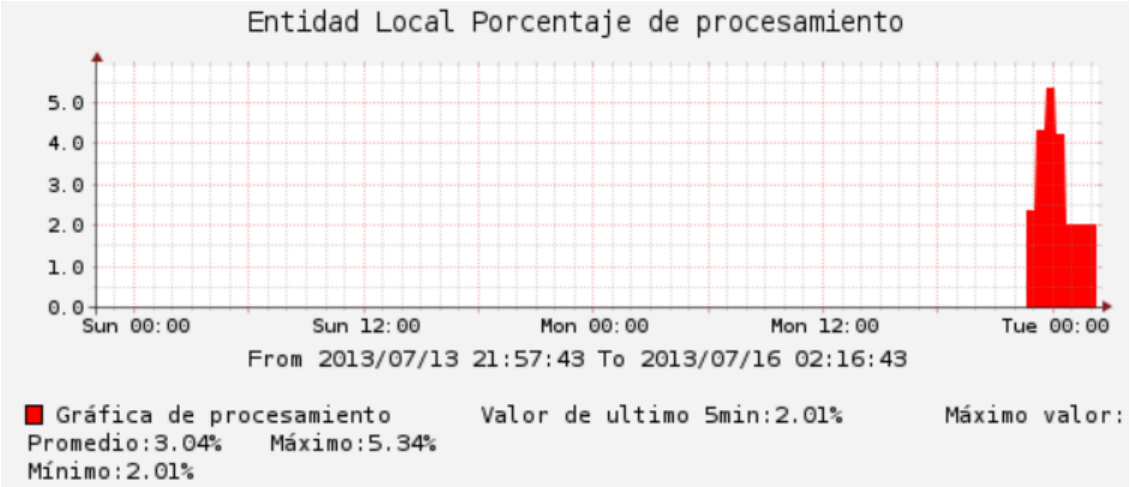

Figura 4.5: Gráfica del nivel de procesamiento del proceso timers ejecutándose en un router Cisco

Finalmente, mediante el uso del plugin Weathermaps, se mostrará un ranking de los procesos con un nivel de procesamiento alto como se encuentra en la figura 4.6 y 4.7. para lo cual se requirió la configuración, mostrada en la tabla 4.3, en el archivo de configuración del mapa creado.

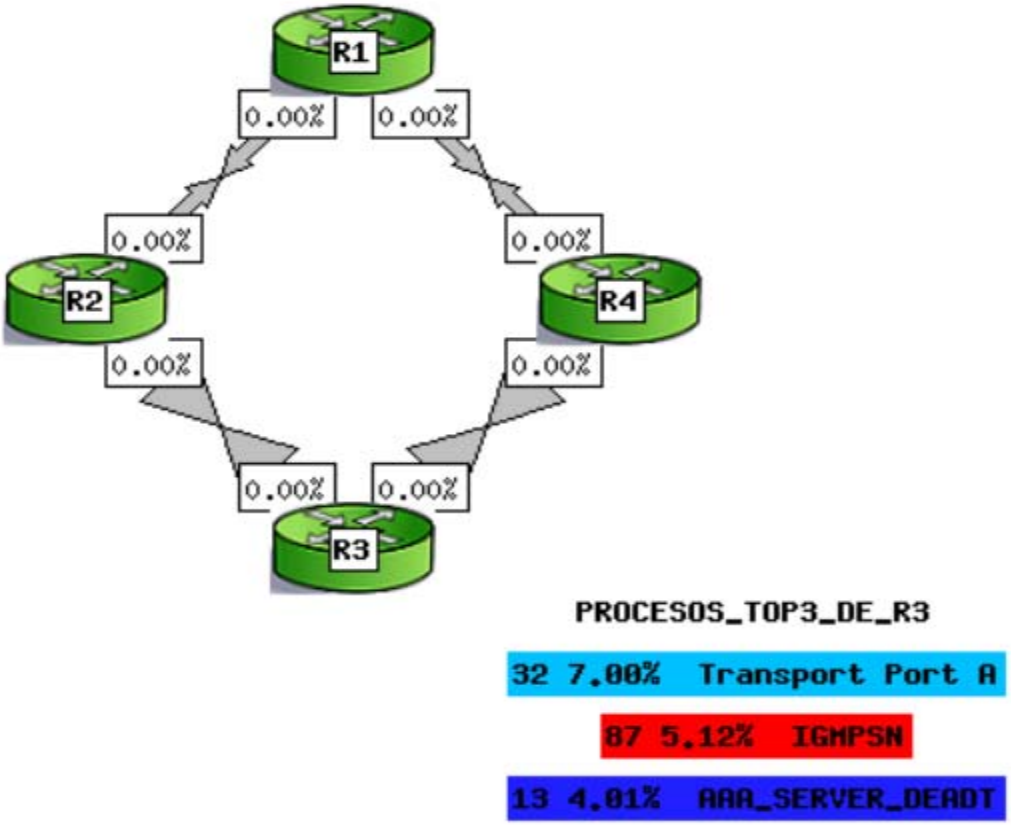

Figura 4.6: Gráfica de la topología de pruebas y ranking de los tres primeros procesos ejecutándose en un router Cisco en tiempo real

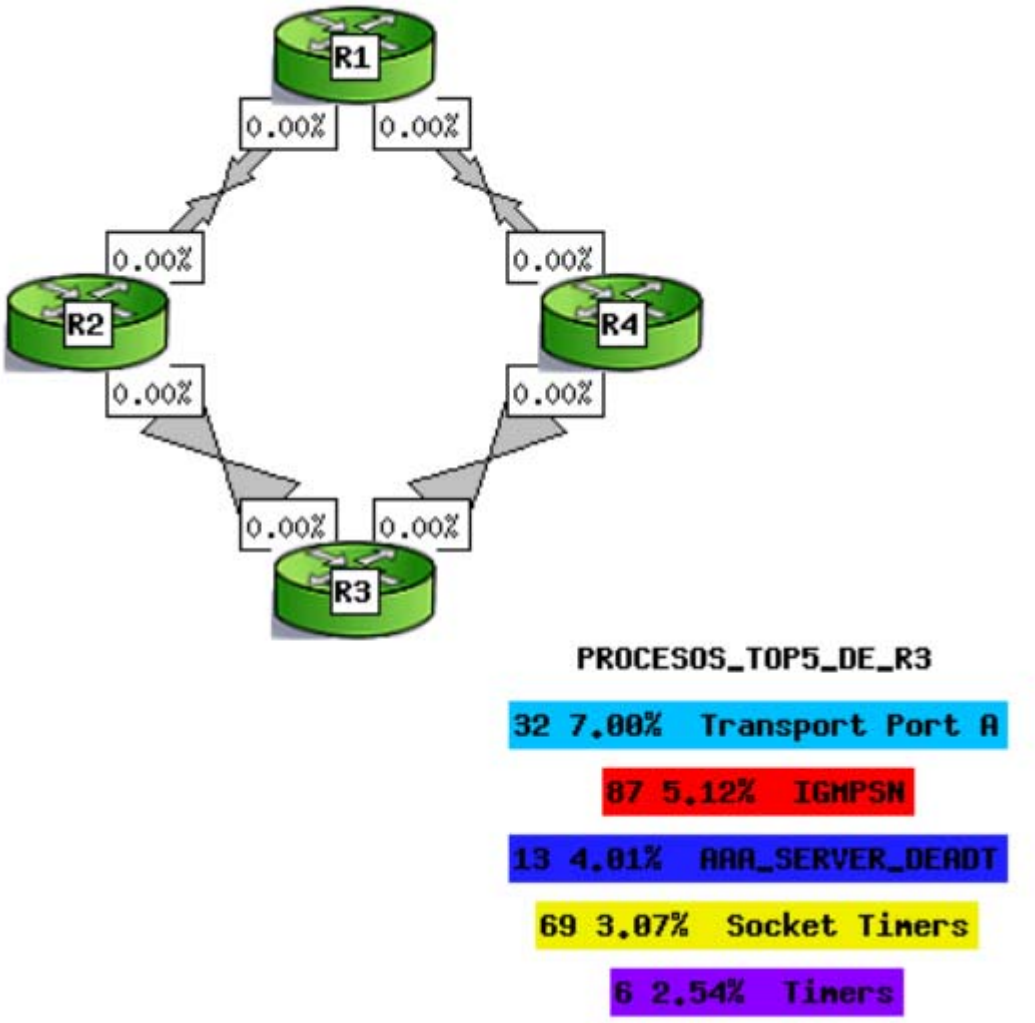

Figura 4.7: Gráfica de la topología de pruebas y ranking de los cinco primeros procesos ejecutándose en un router Cisco en tiempo real

A continuación, la configuración realizada en el archivo de configuración del plugin Weathermaps.

Tabla 4.3: Configuración de las etiquetas de ranking de procesamiento de los primeros 3 procesos con mayor consumo

```
NODE PROCESOS_TOP
   LABEL PROCESOS_TOP_DE_3
   LABELOFFSET E
   LABELOUTLINECOLOR none
      POSITION R1 151 225

NODE PROCESO_PUESTO_1
```

```
   SET community public
   SET ip localhost
   SET posiciOnDelProceso 1
   TARGET
snmp:{node:this:community}:{node:this:ip}:.1.3.6.1.4.1.8072.2.7
000.1.4.{node:this:posiciOnDelProceso}:-
   LABEL {node:this:snmp_in_raw}
   LABELOFFSET E
   LABELOUTLINECOLOR none
     POSITION R1 151 250

NODE PROCESO_PUESTO_2
   SET community public
   SET ip localhost
   SET posiciOnDelProceso 2
   TARGET
snmp:{node:this:community}:{node:this:ip}:.1.3.6.1.4.1.8072.2.7
000.1.4.{node:this:posiciOnDelProceso}:-
   LABEL {node:this:snmp_in_raw}
   LABELOFFSET E
   LABELOUTLINECOLOR none
     POSITION R1 151 275

NODE PROCESO_PUESTO_3
   SET community public
   SET ip localhost
   SET posiciOnDelProceso 3
   TARGET
snmp:{node:this:community}:{node:this:ip}:.1.3.6.1.4.1.8072.2.7
000.1.4.{node:this:posiciOnDelProceso}:-
   LABEL {node:this:snmp_in_raw}
   LABELOFFSET E
   LABELOUTLINECOLOR none
     POSITION R1 151 300
```

4.2 Análisis de los resultados

Los resultados obtenidos en el punto 4.1.1 hacen referencia a la cantidad de contextos PDP activos que se encuentran utilizando recursos de red del operador móvil.

El valor del contexto pdp es de suma importancia; debido a que cada vez que un usuario móvil se conecta a la red de un operador a través de un punto de acceso determinado (APN), se le asigna una dirección IP y a la relación entre esta dirección IP asignada y el APN utilizado para el acceso se le denomina sesión IP. Dentro de cada una de estas sesiones IP pueden establecerse, a su vez, varias asociaciones lógicas

entre la dirección IP del usuario móvil y los distintos recursos de red (a través del APN usado en dicha sesión IP). Cada una de las diferentes asociaciones lógicas establecidas constituye lo que se denomina contexto PDP.

Como se puede apreciar en la figura 4.2, los contextos PDP activos en promedio son mayores a 300 000; lo cual sirve para predecir comportamientos de los distintos usuarios del operador móvil.

Así como este resultado, se puede obtener otros valores relevantes para los operadores de red (ver anexos A y B); los cuales sirvan para predecir las tendencias de los usuarios; así mismo responder rápidamente a algún fallo ocurrido en la red. Disminuyendo el tiempo y costo en las operaciones de la red.

4.3 Análisis de costos

El total de costos asumidos, según las proyecciones realizadas, se describe en el siguiente cuadro.

Tabla 4.4: Análisis de costos

	Costo Mensual (S/.)	**Número de meses**	**Costo (S/.)**
Honorarios profesionales	12,000	6	72,000
Materiales de oficina	30	6	180
Servicios	300	6	1,800
		Costo Total	73,980

4.3.1 Honorarios profesionales

Teniendo en cuenta que el desarrollo de software fue realizado por una sola persona a tiempo completo; y estimando S/.75.00 el costo de la hora hombre, con 1 año de experiencia en tema, por 160 horas trabajadas mensualmente, la remuneración mensual recibida por el colaborador sería de S/.12,000.00.

Finalmente, debido a que la duración del proyecto fue de 6 meses, la cotización final por mano de obra sería de S/.72,000.00.

4.3.2 Materiales de oficina

En el presente trabajo se tomaron en cuenta los materiales de oficinas utilizados, como lapiceros, cuadernos, USB, lápices, entre otros. Necesarios para los apuntes pertinentes, reuniones de planificación y diseño de la solución. El costo total de estos materiales fue de S/.30.00 mensuales.

4.4 Servicios

En este punto se tomó en consideración los gastos producidos por el uso de servicios básicos como luz, agua, teléfono y acceso a internet. Los cuales permiten que el desarrollo de la solución se lleve bajo un ambiente agradable. El costo total de estos materiales fue de S/.300.00 mensuales.

Conclusiones

- Se cumplió con el objetivo principal de diseñar e implementar un método para extender las funcionalidades del actual gestor de red utilizado en el operador móvil mediante la integración de las distintas tecnologías, basado en software libre.
- El sistema permite obtener y almacenar estadísticas de los distintos recursos de red de las entidades que no cuentan con una implementación profunda del protocolo SNMP, observando sus características actuales e históricas con los intervalos de tiempo necesitados por los operadores de red cada 5 min, 1 hora, 1 día, 1 semana, 1 año, sucesivamente.
- El sistema ha sido implementado utilizando software libre, esto permitirá ahorros en licencias y la posibilidad de adaptar la solución a las necesidades que se requieran.

Recomendaciones

- Las consultas realizadas por el servicio extendido SNMP se realizaron con valores estáticos dentro del archivo de configuración del demonio snmpd del servidor. Si se desea agregar un nuevo parámetro a gestionar, este deberá ser añadido de forma manual, siguiendo el método presentado anteriormente.
  Por ello, para agregar un mayor dinamismo en el proceso, este deberá ser agregado con un mayor desarrollo en programación en el código fuente del gestor Cacti.
- Al momento de la implementación, se debe tener en cuenta los permisos necesarios de lectura y escritura dentro del sistema de red para el correcto funcionamiento del servicio SNMP.

Bibliografía

[1] CISCO SYSTEMS
2012 *"Network Based Application Recognition (NBAR)".* Consulta: 1 de julio de 2012. <http://www.cisco.com/en/US/products/ps6616/products_ios_protocol_group_home.html>

[2] Sanders, Geoffrey
2003 *"GPRS networks"*. Chichester : Wiley, 2003

[3] Díaz, Arturo
2007 *"Diseño e implementación del centro de operación y gestión de la red académica peruana en software libre"*. Tesis para optar el título de ingeniero de las telecomunicaciones. Lima: Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería.

[4] ISO
2012 *"About ISO".* Consulta: 18 de Julio de 2012. <http://www.iso.org/iso/home/about.htm>

[5] FCAPS
2013 *"Network Sales and Services Handbook"* <http://my.safaribooksonline.com/1587050900/ch02?portal=ciscopress#ch02 >

[6] MAURO Douglas y Kevin SCHMIDT
2005 "Essential SNMP". Segunda edición. Estados Unidos. O'Reilly Media.

[7] TL1
2013 *"What is TL1?"* Consulta: 18 de Julio de 2012. <http://www.dpstele.com/white-papers/essential_tl1_guide/what_is_tl1.php>

[8] HEWLETT-PACKARD
2013 *"HP Openview".*
Consulta: 09 de junio de 2013.
<http://support.openview.hp.com/>

[9] HUAWEI Tech.
2012 *"iManager M2000".*
Consulta: 14 de agosto de 2012.
<http://enterprise.huawei.com/en/products/network/wireless/lte/hw-198009.htm>

[10] The Cacti Group
2012 *"Cacti".*
Consulta: 18 de junio de 2012.
<http://www.cacti.net/>

[11] NAGIOS Enterprises
2012 *"Nagios".*
Consulta: 19 de junio de 2012.
<http://www.nagios.org/>

[12] OpenNMS
2012 *"OpenNMS".*
Consulta: 19 de junio de 2012.
<http://www.opennms.org/>

[13] NET-SNMP
2012 *"NET-SNMP".*
Consulta: 05 de agosto de 2012.
<http://www.net-snmp.org/>

[14] Ubuntu-es
2012 *"Bash"*
Consulta: 20 de septiembre de 2012.
<http://doc.ubuntu-es.org/Bash>