UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

La Universidad Católica de Loja

MODALIDAD PRESENCIAL

ESCUELA DE CIENCIAS DE LA COMPUTACIÓN

COMPILACIÓN Y AUTOMATIZACIÓN DE UN SISTEMA DE GESTIÓN DE REDES NOC CON HERRAMIENTAS DE CÓDIGO ABIERTO

Trabajo de fin de carrera previa a la obtención del título de ingeniero en sistemas informáticos y computación

Autores:

Ramírez Paucar Verónica Leonor

Director:

Ing. Aguilar Mora Carlos

LOJA-ECUADOR

2011

CERTIFICACIÓN

Ingeniero
Carlos Aguilar
DIRECTOR DE TESIS

CERTIFICA:

Haber dirigido y supervisado el desarrollo del presente proyecto de tesis previo a la obtención del título de **INGENIERA EN SISTEMAS INFORMÁTICOS Y COMPUTACIÓN**, presentado por la alumna **Srta. Verónica Leonor Ramírez Paucar**, y una vez que este cumple con todas las exigencias y los requisitos legales establecidos por la Universidad Técnica Particular de Loja, autoriza su presentación para los fines legales pertinentes.

Loja, 21 de marzo de 2011

Ing. Carlos Mora Aguilar
DIRECTOR DE TESIS

CERTIFICACIÓN

Ingeniero
Julia Pineda

CODIRECTORA DE TESIS

CERTIFICA:

Haber dirigido y supervisado el desarrollo del presente proyecto de tesis previo a la obtención del título de **INGENIERA EN SISTEMAS INFORMÁTICOS Y COMPUTACIÓN**, presentado por la alumna **Srta. Verónica Leonor Ramírez Paucar**, y una vez que este cumple con todas las exigencias y los requisitos legales establecidos por la Universidad Técnica Particular de Loja, autoriza su presentación para los fines legales pertinentes.

Loja, 21 de marzo de 2011

Ing. Julia Pineda

CODIRECTORA DE TESIS

AUTORÍA

El presente proyecto de tesis con cada una de sus observaciones, análisis, evaluaciones, conclusiones y recomendaciones emitidas, son de absoluta responsabilidad de su autor.

Además, es necesario indicar que la información de otros autores empleada en el presente trabajo está debidamente especificada en fuentes de referencia y apartados bibliográficos.

_____

f) Verónica Leonor Ramírez Paucar

CESIÓN DE DERECHOS

Yo, Verónica Leonor Ramírez Paucar declaro ser autora del presente trabajo y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales.

Adicionalmente declaro conocer y aceptar la disposición del Art. 67 del Estatuto orgánico de la Universidad Técnica Particular de Loja que su parte pertinente textualmente dice: "Forman parte del parte del patrimonio de la universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado que se realicen a través, o con el apoyo financiero académico o institucional (operativo) de la universidad".

f) Verónica Leonor Ramírez Paucar

DEDICATORIA

A mi abuelito (†) por su afecto, sus enseñanzas, sus consejos y valores impartidos que me supieron formar como persona y practicarlos a lo largo de mi vida universitaria.

A mis padres Miguel Ramírez y Digna Paucar porque gracias a su cariño, apoyo y enseñanzas me encuentro aquí, alcanzando unos de los anhelos más grandes de mi vida.

A mis hermanos por haberme inspirado a ser mejor con cada una de sus palabras consejos y acciones.

AGRADECIMIENTO

A dios por guiar mis pasos y darme la fortaleza necesaria a lo largo de mi carrera universitaria y permitirme llegar a este momento tan importante.

A mis padres porque a ellos les debo todo lo que soy y siempre serán mi referencia en la vida, son las personas que más admiro y quiero. Les agradezco por la oportunidad de existir, por los sacrificios que han hecho, por su ejemplo de superación incansable, por su comprensión y confianza.

A mi abuelito (†) porque aunque ya no esté conmigo siempre me ayudo a salir adelante. Siempre te llevare en mi mente y mi corazón

A mis hermanos por su cariño, enseñanza, apoyo incondicional y consejos.

Al Ing. Carlos Aguilar, director de tesis y la Ing. Julia Pineda, codirectora por su gran sentido de responsabilidad, su tiempo, sus ideas y dedicación.

A mis profesores de carrera por su conocimiento y experiencia profesional trasmitida.

A mis amigos pasados y presentes porque sin ustedes mi estancia en la escuela no hubiera sido lo mismo, gracias por compartir conmigo tantas experiencias, risas y llantos.

A todos aquellos que de manera directa o indirecta aportaron con su conocimiento para la realización de la presente investigación

Tabla de contenidos

INDICE DE FIGURAS

ÍNDICE DE TABLAS

RESUMEN

El presente proyecto tiene como resultado, la implementación de un sistema de gestión de red para el NOC[1] de la Universidad Técnica Particular de Loja, permitiendo el monitoreo de diferentes tipos de dispositivos, recursos, servicios y su gestión mediante el uso del protocolo SNMP[2], logrando así mantener la red disponible el mayor tiempo posible.

Es importante acotar que la solución está integrada con herramientas muy importantes de la gestión de redes incluyendo Cacti y Nagios que aportan con las funcionalidades necesarias para la gestión de la red.

El presente documento posee información detallada como: estándares del NOC, estudio del NOC-UTPL , investigación y selección del NMS[3], instalación y configuración de la herramienta, pruebas y monitoreo, terminando con algunas conclusiones y recomendaciones que se espera sirva para el desarrollo de trabajos futuros relacionados con la gestión de redes. Todos estos temas se encuentran distribuidos en el presente documento de la siguiente manera:

Consideraciones teóricas de gestión de redes: se realiza un estudio de los conceptos referentes a la gestión de redes: componentes, protocolos, áreas funcionales, actividades de un NOC.

[1] **Network Operation center**: centro de operación de red responsable de analizar el funcionamiento y operación de todos los equipos que componen la red y el Centro de Datos.
[2] **SNMP** (Simple Network Management Protocol) protocolo de gestión de red.
[3] **Network Management Systems:** Sistema de gestión de red , es un software utilizado para supervisar y administrar una red

Situación actual: se muestra un vistazo a la situación actual del NOC-UTPL, su infraestructura y organización.

Definición de los requerimientos: en base a necesidades e información recolectada de los NOC externos además del NOC-UTPL, se especifica los requerimientos a cumplir en el presente proyecto.

Estudio y selección de herramientas: se realiza una investigación de las herramientas necesarias para la gestión de la red, finalmente en base a ciertos criterios se evalúa y selecciona la herramienta para el NOC de la UTPL.

Implementación de la solución: se realiza una descripción de la aplicación a nivel general, cumplimiento de requerimientos, implementación de la herramienta en un entorno de pruebas, preparación del entorno, instalación y configuración de Zenoss en el servidor del NOC-UTPL, se realiza la documentación respectiva de los errores encontrados y de las pruebas de monitoreo efectuadas en el servidor NOC.

Trabajos futuros: se especifica algunos de los posibles trabajos que se puedan desarrollar en futuras investigaciones.

Conclusiones y recomendaciones: finalmente se puntualizan algunas conclusiones y recomendaciones del proyecto y un conjunto de referencias bibliográficas en las que se ha fundamentado este trabajo.

INTRODUCCIÓN

Con el paso del tiempo las redes se han incrementado continuamente, llegando en la actualidad a la existencia masiva de redes y por tanto resulta más compleja y difícil la tarea de gestión de los dispositivos de red. Para las personas encargadas de la gestión y monitoreo de equipos o servidores de una empresa, es muy primordial conocer el estado y tener control de la misma, razón por la cual es importante lograr una gestión satisfactoria, haciendo uso de las herramientas de monitoreo y gestión de redes, mediante las cuales se puede conocer el estado de los equipos de red, los procesos que se están ejecutando en cada equipo , la carga del sistema, el uso de memoria, el tráfico de red de cada interfaz, el inventario de software que posee un dispositivo entre otros aspectos primordiales para el administrador de red a la hora de detectar fallos y actuar con responsabilidad.

Actualmente existe una variedad de software para el monitoreo de redes tanto libres como comerciales. Esta tesis está orientada al estudio e implementación de herramientas de software libre.

OBJETIVOS

OBJETIVO GENERAL

Realizar el estudio e implementación de herramientas de código abierto que permitan realizar el monitoreo y la gestión de red dentro de un NOC, para el correcto desempeño y funcionamiento de la misma.

OBJETIVOS ESPECÍFICOS

- Investigar la estructura y procesos de un NOC.
- Analizar el esquema y estado actual del NOC-UTPL.
- Identificar herramientas de código abierto con funcionalidades relevantes para el NOC.
- Implementación y configuración de la solución en la red UTPL.
- Elaboración del manual de procesos.

1

CAPÍTULO 1. CONSIDERACIONES TEÓRICAS DE LA GESTIÓN DE REDES

1.1 GESTIÓN DE RED

Definición

Conjunto de actividades dedicadas a la planificación, organización, vigilancia y control de los recursos que conforman la red para garantizar la seguridad, rendimiento y disponibilidad.

"La gestión de redes incluye el despliegue, integración y coordinación del hardware, software y los elementos humanos para monitorear, probar, sondear, configurar, analizar, evaluar y controlar los recursos de la red para conseguir los requerimientos de tiempo real, desempeño operacional y calidad de servicio a un precio razonable"(T.Magedanz, 1996) .

La ISO[4] define la gestión de red como:

- *"El conjunto de elementos de control y supervisión de los recursos que permiten que la comunicación tenga lugar sobre la red".*

Objetivo

"Mejorar la disponibilidad y rendimiento de las redes e incrementar su efectividad, lográndose una mayor productividad en la institución y un aumento de la satisfacción de los usuarios" (Cubells, 2005) .

1.1.1 COMPONENTES DE GESTIÓN DE REDES

Los principales componentes existentes en la gestión de redes son:

- **Dispositivo Gestionado (hardware):** dispositivo o nodo[5] de la red que permite el almacenamiento de la información de gestión de la red.
- **Sistema de Gestión de Red (NMS):** es una aplicación o el sistema general dedicado a la gestión de la red que permite ejecutar aplicaciones para desplegar datos de gestión, monitorear y controlar dispositivos gestionados y comunicarse con los agentes.
- **Agente:** es un software residente en los dispositivos gestionados que permite la gestión del dispositivo.

[4] **ISO** (International Organization for Standardization), es la Organización Internacional para la Estandarización.
[5] **Nodo** Punto de conexión, ya sea un punto de redistribución o un punto final de la comunicación

1.2 NOC (Network Operation Center)

Definición

- NOC es un centro responsable de las operaciones diarias, la supervisión, monitoreo y gestión de la red para la obtención de información como: disponibilidad, estado, estadísticas y fallos.

Objetivo

- Simplificar el seguimiento de la red mediante un NMS para controlar supervisar, monitorear y vigilar la misma logrando mantenerla disponible el mayor tiempo posible.

La administración o gestión de red está centralizada en el NOC (centro de operación de gestión) de la empresa, desde donde se realiza todas las actividades correspondientes al monitoreo de los recursos de la red. Este centro consta de tres recursos principales:

- *Procesos o métodos de gestión:* establecen pautas o procesos a seguir que rigen el comportamiento de los otros componentes de gestión ante sucesos determinados.
- *Recurso Humano:* las personas encargadas del funcionamiento eficiente y correcto del NOC.
- *Herramientas:* aplicaciones relacionadas con la gestión de red que faciliten las tareas llevadas a cabo por el personal del NOC en la gestión de la red y por ende reduzcan el número de personas del NOC.

1.3 PRINCIPALES ACTIVIDADES A REALIZAR PARA FORMAR UN NOC

Según Oppenheimer (Oppenheimer & Press, 2007), las principales actividades que todo centro de operación de gestión debe llevar a cabo para gestionar la red son:

- Determinar los dispositivos de red a monitorear, que datos obtener de estos dispositivos y cómo interpretar esos datos.
- Desarrollar los procesos a seguir para gestionar el rendimiento, fallas, configuración, seguridad y contabilización.
- Desarrollar una arquitectura de gestión de redes.
- Elegir los protocolos y herramientas para la gestión de la red.

1.3.1 ASPECTOS FUNCIONALES DE GESTIÓN DE RED.

La ISO posee uno de los principales modelos de administración para entender las funciones principales de los sistemas de gestión de redes denominadas Áreas Funcionales Específicas de Gestión, las cuales se muestran a continuación:

- Performance management (Gestión de rendimiento).
- Configuration management (Gestión de configuración).
- Accounting management (Gestión de contabilización).
- Fault management (Gestión de fallas).
- Security management (Gestión de seguridad).

1.3.1.1 Gestión de rendimiento

Medir y proveer información del desempeño de la red manteniendo así un nivel aceptable de funcionamiento de la red.

La gestión de rendimiento implica los siguientes pasos:

- Recolección de la información del funcionamiento de la red.
- Análisis de la información recolectada para determinar los niveles (umbrales) normales de la red.
- Establecer niveles límite del rendimiento y verificar la información, si esta excede los umbrales establecidos de la red, enviar mensajes comunicando los problemas de la red.

1.3.1.2 Gestión de configuración

Supervisar información sobre la configuración de los dispositivos de la red para realizar un seguimiento de las versiones de software y hardware disponibles en los dispositivos.

La gestión de configuración implica lo siguiente:

- Recoger información del estado actual de la red.
- Gestión de inventario.
- Mantenimiento de directorios.
- Control operacional de la red.

1.3.1.3 Gestión de contabilización

Realizar un seguimiento de los parámetros de utilización de la red para regular las aplicaciones usadas de los usuarios y grupos, y así reducir al mínimo los problemas de red. Algunas de las tareas a desarrollarse en esta gestión son:

- Seguimiento del uso de la red y recolección de datos sobre el uso de los recursos de la misma e informar a los usuarios de estos datos.

- Establecer costos asociados con el uso de los recursos de la red.
- Encontrar a las personas que usan más recursos de los que deberían y cobro a los usuarios por la utilización de los recursos.

1.3.1.4 Gestión de fallas

Localizar, registrar, notificar y corregir los problemas existentes en la red.

La gestión de falla en la red involucra las siguientes tareas:

- Identificar de la falla.
- Aislar la falla.
- Reaccionar ante la falla.
- Corregir la falla.

1.3.1.5 Gestión de seguridad

Controlar el acceso a los recursos de la red en base a las políticas de seguridad de manera que esté protegida contra intrusos maliciosos y usuarios no autorizados, etc.

Las funciones que realiza esta gestión son:

- Identificación de la información más relevante a proteger y localización de los dispositivos más sensibles de la red en donde se encuentra dicha información.
- Análisis de la relación existente entre los dispositivos sensibles y los grupos de usuarios.
- Monitoreo y protección de los puntos de accesos a los dispositivos sensibles de la red.

- Almacenamiento de los intentos de acceso a los recursos de la red para su análisis posterior.

Los aspectos funcionales de la gestión de red brindan servicio a las actividades de Monitoreo y Control de red, y se las puede ubicar de la siguiente manera:

a) **Monitoreo de la red**: obtiene información de los elementos (IT-AUT-UAH, 2005):

- Gestión de rendimiento.
- Gestión de fallas.
- Gestión de contabilidad.
- Gestión de configuraciones.

b) **Control de la red**: actúa sobre los elementos (IT-AUT-UAH, 2005):

- Gestión de configuraciones.
- Gestión de seguridad.

1.3.2 ARQUITECTURA DE GESTIÓN DE REDES

Durante el transcurso de los años han ido evolucionando las redes y esto ha dado la existencia de distintos tipos de gestión:

- ***Gestión Autónoma:*** cada nodo poseía un administrador, al surgir algún problema que afectaba a más de un nodo los administradores se ponían en contacto para resolverlo.
- ***Gestión Heterogénea:*** en los ochenta, las redes crecieron y para solucionar la administración de las mismas aparecieron aplicaciones para la supervisión remota, pero solo funcionaban para dispositivos de un mismo fabricante.

- ***Gestión Integrada:*** surgen sistemas que permiten el uso de un centro de gestión para controlar los entornos heterogéneos y para ello es necesario una estandarización de la gestión de red.

Con el continuo surgimiento de los diversos tipos de gestión, hoy en día la mayoría de centros de operación de red mantienen una gestión integrada por las facilidades que esta otorga. El NOC-UTPL mantiene este tipo de arquitectura para la gestión de la red. Siendo esta la más destacada y utilizada en la UTPL. A continuación se procede al estudio de la misma conjuntamente con los modelos de gestión que esta ofrece.

Actualmente existen 3 modelos de la gestión de red.

- **Gestión de Red OSI**[6] *(Open Systems Interconnection*). Este modelo es definido por la ISO, su objetivo es lograr la gestión de los recursos del modelo de referencia OSI.
- **Gestión Internet (TCP/IP)**. Definido por la *Internet Society,* fue creado con el objetivo de gestionar el modelo de referencia TCP/IP[7].
- **Arquitectura TMN**[8] *Telecommunications Management Network*. Definida por la ITU-T. Este modelo define la estructura de red basada en los modelos anteriores.

[6] **OSI** (Open System Interconnection), es un estándar ISO para las comunicaciones en todo el mundo que define un marco para la aplicación de protocolos de redes en siete capas.

[7] **TCP/IP** Conjunto de protocolos de red en los que se basa Internet y que permiten la transmisión de datos entre redes de computadoras.

[8] **TMN** Es un modelo de protocolo para la gestión de los sistemas abiertos en una red de comunicaciones.

1.3.3 PROTOCOLOS DE GESTIÓN DE RED

A continuación se presentan los principales protocolos y los más utilizados para la gestión de red, distribuidos de acuerdo al modelo de gestión al que pertenecen:

Modelo de gestión de red OSI (Open Systems Interconnection)

El protocolo más destacado en el modelo de gestión de red OSI es el CMIP

CMIP

CMIP, por sus siglas “”Common Management Information Protocol” es un protocolo de gestión de red que define la comunicación entre las aplicaciones de administración de la red y la gerencia de los agentes, este protocolo está basado en el modelo OSI.

Características (Chemary, 2006)

- Las especificaciones son difíciles de realizar y tediosas de implementar en aplicaciones.
- La comunicación con los agentes está orientada a conexión.
- La estructura de funcionamiento es distribuida.
- El protocolo asegura que los mensajes llegan a su destino.

Funciones (Veloso, 2006)

- **Administración Contable:** Proporciona una forma de monitorear el uso de la red para cobrar a los usuarios o para medir los costos y prevenir sobregiros de los presupuestos.
- **Administración de la Configuración:** Por medio de interfaces gráficos, el administrador puede seleccionar y reconfigurar puentes, enrutadores y otros dispositivos de comunicación.
- **Administración de Fallas:** Detecta y corrige fallas en la red. Analiza aspectos que ayudan a determinar las causas de falla. Dispone de alarmas para alertar a los administradores.
- **Administración del Comportamiento:** Proporciona servicios para monitorear la red y mejorar su comportamiento. Llevar estadísticas es uno de los elementos esenciales para administrar el comportamiento.
- **Seguridad:** Proporciona servicios de seguridad de alto nivel que puede autentificar a los usuarios, detecta intrusiones y asegura la confidencialidad en la transmisión de datos.

Servicios

A través de CMISE *(Common Management Information Service Element),* CMIP proporciona tres tipos de servicio:

- Manejo de datos: usado por el gestor para solicitar y alterar información de los recursos del agente.
- Informe de sucesos: usado por el agente para informar al gestor sobre diversos sucesos de interés.

- Control directo: usado por el gestor para solicitar la ejecución de diversas acciones en el agente.

Modelo de gestión internet (TCP/IP).

Los protocolos de gestión más importantes y que se han destacado en este modelo son: SNMP, RMON.

SNMP

(Simple Network Management Protocol) es un protocolo de Gestión de Red ubicado en la capa de aplicación que facilita el intercambio de información entre dispositivos de red. Permite a los administradores supervisar el desempeño de la red, buscar y resolver sus problemas, y planear su crecimiento.

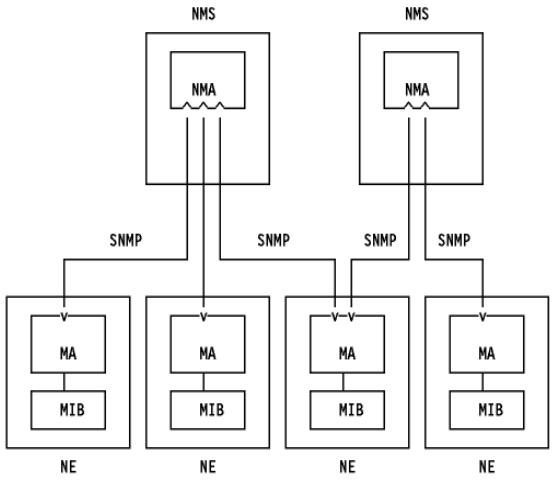

Figura 1-1 Componentes SNMP **(ULPGC, 2010)**

El modelo SNMP de una red administrada consta de cuatro componentes:

- Nodos administrados.
- Estaciones administradas.
- Información de administración.
- Un protocolo de administración.

Nodos administrados: Entre estos tenemos hosts, enrutadores, puentes, impresoras u otros dispositivos. Para ser administrado directamente por el SNMP, un nodo debe ser capaz de ejecutar un proceso de administración SNMP, llamado agente SNMP. Cada agente mantiene una base de datos local de variables que describen su estado e historia y que afectan a su operación. (IT-AUT-UAH, 2005)

Estaciones administradoras: Es aquella en donde se realiza la administración de la red por ejemplo ordenadores con un software de administración especial. Esta estación contiene procesos que se comunican con los agentes a través de la red emitiendo comandos y recibiendo respuestas.

Información de administración: El SNMP describe la información exacta de cada tipo de agente que tiene que administrar la estación administradora y el formato con el que el agente tiene que proporcionarle los datos. Cada dispositivo mantiene una o más variables que describen su estado, estas variables se llaman objetos. El conjunto de todos los objetos posibles de una red se da en la estructura de datos llamada MIB (Management Information Base, Base de Información de Administración). (IT-AUT-UAH, 2005)

Protocolo de administración: Se requiere un protocolo de comunicación entre gestor y agentes. Sus funciones son:

- Leer y actualizar los atributos de los objetos gestionados.
- Ordenar la ejecución de funciones específicas a los objetos gestionados.
- Reportar los resultados obtenidos por los objetos gestionados
- Crear y suprimir objetos gestionables.

Para esto se utiliza el protocolo SNMP el cual es utilizado como medio para la interacción entre la estación administradora y los agentes permitiéndole a la estación consultar y modificar el estado de los objetos de un agente.

VERSIONES SNMP

Hasta la actualidad existen 3 versiones SNMP, estas versiones poseen la misma estructura pero se diferencia entre sí por las nuevas características que se añaden en cada versión. La v1 posee las siguientes peticiones get-request, get-next-request, get-response, set-request, set-next-request, trap[9], la seguridad se basa en comunidades que permiten usar unos dispositivos u otros si se conoce el password. La v2, se le añaden dos nuevos comandos de petición get-bulk-request e inform-request para Reducir la carga de tráfico adicional para la motorización y corrige los problemas de motorización remota o distribuida con las sondas RMON. La v3 posee cambios significativos con relación a sus predecesores como autenticación robusta,

[9] **Trap** Es un código o una señal diseñada para capturar los errores y poner de manifiesto dónde están.

privacidad débil, método SHA[10] sobre todo en aspectos de seguridad, sin embargo no ha sido mayoritariamente aceptado en la industria.

MENSAJES ENVIADOS POR SNMP

GET REQUEST: Solicita el valor de una o más variables de un objeto o equipo en específico. Es transmitido por el NMS y recibido por el agente.

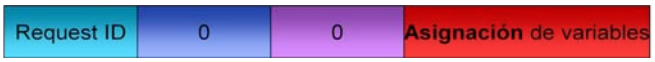

Figura 1-2 GetRequest

GET NEXT REQUEST: Solicita el valor de la siguiente variable de un objeto o equipo específico.

Figura 1-3 GetNextRequest

GET BULK REQUEST: Presente en SNMP v2, solicita un amplio conjunto de valores en vez de ir solicitando uno por uno para facilitar la transferencia de grandes bloques de datos.

SET REQUEST: Actualiza y modifica uno o varios valores de variables específicas de un equipo.

Figura 1-4 SetRequest

[10] **SHA** (Secure Hash Algorithm) Un popular algoritmo hash unidireccional que se utiliza para crear firmas digitales.

SET NEXT REQUEST: Actualiza el siguiente valor de variables de un objeto o equipo en específico.

GET RESPONSE: Devuelve los atributos solicitados. Es transmitido por el agente (o nodo administrado) y recibido por el NMS (o nodo administrador).

Figura 1-5 GetResponse

TRAP: es un mensaje enviado por SNMP informando fallos ocurridos en el agente (como pérdida de la comunicación, caída de un servicio, problemas con la interfaz, etc.).

Otros mensajes utilizados en SNMP:

INFORM REQUEST: Describe la base local de información de gestión MIB para intercambiar información entre gestores.

RMON

RMON1 consiste en el uso de un agente remoto para colectar información de gestión bajo demanda. Se diseñó para redes LAN[11] Ethernet[12] (posteriormente para Token Ring[13]) y entrega las funcionalidades de los analizadores de redes y protocolos. (Toares, 2010).

[11] **LAN** Red de área local), es una red informática que abarca una área física pequeña.
[12] **Ethernet** Es un estándar de red para la transmisión de datos mediante cable de par trenzado o coaxial.
[13] **Token Ring** Es una arquitectura de red desarrollada por IBM en los años 1970 con topología lógica en anillo y técnica de acceso de paso de testigo.

RMON permite el uso de agentes "inteligentes" que responden de acuerdo con acontecimientos excepcionales. Esto reduce el tráfico asociado con la red de gestión, mientras que permite al equipo remoto alertar a la plataforma de gestión SNMP cuando ocurre algún problema.

Características (Chemary, 2006)

- Monitorización preventiva: se puede enviar periódicamente información de estatus de la red.
- Múltiples gestores: RMON permite la estructura de plataformas gestoras dispuestas de forma distribuida y jerárquica.

Funciones (Toares, 2010)

- ***Statistics*.** Provee estadísticas desde una red, hub de LAN o usuario. Por ejemplo, la cantidad de errores en una puerta específica.
- ***History.*** Forma la historia de las estadísticas anteriores. Es útil para establecer la actividad en la red.
- ***Alarm.*** Entrega un mecanismo de selección para el seteo de umbrales o intervalos para enviar *Trap*.
- ***HostTable Group*.** Soporta estadísticas de tráfico para la red, hub o usuario.
- ***HostTopNTable.*** Soporta estadísticas del host para tabla de direcciones.
- ***TrafficMatrixGroup*** Indica el tráfico en una matriz para cada par de estaciones.
- ***Filter.*** Provee un filtro programable para datos, contador o para ejecutar eventos.

- ***PacketCaptura.*** Captura paquetes de acuerdo con el criterio seleccionado en el *Filter.*
- ***Event*.** Crea entidades, envía alarmas y ejecuta acciones.

1.3.4 SISTEMA DE GESTIÓN DE RED

En la actualidad la mayoría de sistemas dedicados a la gestión usan una estructura básica llamada el paradigma gestor-agente (**Figura 1-1**).

Los sistemas de gestión se componen por:

- Interfaz, por medio de la cual el administrador realiza operaciones de gestión a los recursos de la red bajo su responsabilidad.
- Elementos de software y hardware ubicados entre los recursos de la red, estos elementos bajo el paradigma de gestor-agente se clasifican en:
    - Gestores: gestionan la interacción con el recurso humano para llevar a cabo acciones invocadas por el operador humano.
    - Agentes: son invocados por el gestor y radican en los dispositivos gestionados, es el software que reside en estos dispositivos para la gestión del mismo.

El paradigma de gestor-agente se basa en el intercambio de información entre el gestor y el agente (nodos gestionados) que posee información de las características y estado del dispositivo de la red. El gestor se comunica con el agente a través de un protocolo de red para la realización de operaciones indicadas por el gestor para conocer el estado del dispositivo, cuando en un dispositivo ocurre un evento o suceso, el agente sin ser invocado por el gestor, emite notificaciones hacia el gestor de los eventos ocurridos para su posterior gestión.

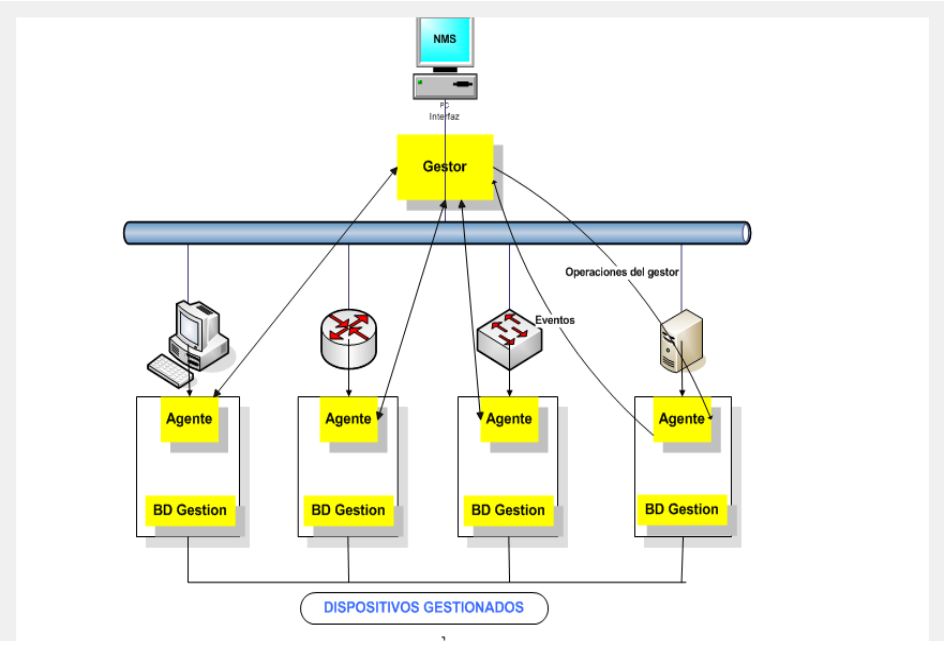

Figura 1-6 Paradigma Gestor-Agente

1.3.4.1 CRITERIOS DE EVALUACIÓN PARA UN NMS

El NOC debe poseer herramientas para la gestión de la red de acuerdo a sus requerimientos. A continuación se presentan algunos criterios de evaluación (Cubells, 2005) a tomar en cuenta para la selección del NMS del NOC-UTPL.

1. Descubrimiento automático de la topología de red.
2. Control de usuario.
3. Mecanismos de notificación.
4. Registro y manipulación de eventos.
5. Generación de reportes.
6. Extensión de funcionalidades.
7. Utilización de protocolos Estándar (SNMP, RMON).
8. Interfaz web.
9. Costo.

2

CAPÍTULO 2. SITUACIÓN ACTUAL DEL NOC-UTPL

Antes de empezar con el análisis y evaluación de la información de las herramientas de gestión de red es importante conocer la situación actual del NOC-UTPL que involucra temas como: objetivos y metas del NOC, dispositivos monitoreados, procesos, arquitectura, herramientas y protocolos para la gestión de red terminando con el planteamiento de la problemática.

2.1 NOC-UTPL

OBJETIVO

"El NOC-UTPL es el grupo encargado de controlar, planificar, coordinar y monitorizar toda la infraestructura de red, enlaces dedicados, enlaces de

Internet y servicios de red de la UTPL, asegurando la disponibilidad, niveles de desempeño y su óptimo funcionamiento." (UTPL, 2009)

METAS

- Proporcionar un esquema operativo adecuado para garantizar el óptimo funcionamiento de la red de la UTPL.
- Seguimiento oportuno para la resolución de fallas en el menor tiempo posible.
- Proporcionar el apoyo necesario a los administradores del Campus, Centros Regionales, Aulas Virtuales y demás servicios para pruebas técnicas.
- Mantener una base de conocimientos de problemas y soluciones.
- Difundir conocimientos adquiridos tanto a los profesionales en formación como a los administradores de servicios de la UTPL.
- Difundir el estado operacional de la red.
- Participación en la definición de estándares y normativas de la operación de la red de CEDIA.
- Desarrollar e implementar herramientas que faciliten las tareas de administración y monitoreo de la red.
- Proveer el servicio de monitoreo a terceros.

(UTPL, 2009)

2.2 DISPOSITIVOS MONITOREADOS

La Red de la UTPL cuenta con dispositivos de red, entre los más importantes: routers[14], swicht y servidores que necesitan un control y monitoreo constante. Actualmente, de estos dispositivos se monitorean:

[14] **Routers** Dispositivo de hardware o software para interconexión de redes de computadoras que opera en la capa tres (nivel de red) del modelo OSI.

servicios (SMTP[15], POP3[16], HTTP[17], etc.), estado de enlace (up, down) y recursos (interfaz, disco, procesador, memoria entre otros).

La red UTPL cuenta con los siguientes dispositivos:

- ROUTERS
  - 13 routers cisco (Aproximado).
- SWICHT
  - 140 swicht. (Aproximado).
- SERVIDORES
  - 20 servidores (Aproximado).

2.3 PROCESOS PARA LA GESTIÓN DE LA RED

Actualmente el NOC-UTPL cuenta con el proceso de gestión de fallas **Figura 1-7** debidamente esquematizado sin contar con una especificación detallada de las actividades a seguir, motivo por el cual se establecerán procesos para la gestión de red amparado en el modelo presentado por la ISO por ser una de las principales opciones que se están utilizando a escala mundial, puesto que garantizan una gestión de calidad y desde el punto de vista económico reduce costes, tiempo y trabajo. Este modelo abarca: gestión del rendimiento, gestión de fallas, gestión de configuración, gestión de seguridad y gestión de contabilización quedando como entregable un manual de procesos para la gestión de la red para el NOC-UTPL.

[15] **SMTP** (Simple Mail Transfer Protocol) es un protocolo TCP / IP protocolo utilizado para enviar y recibir correo electrónico.

[16] **POP3** (Post Office Protocol 3), es un protocolo estándar para recibir mensajes de e-mail.

[17] **HTTP** HyperText Transfer Protocol, es el método más común de intercambio de información en la world wide web.

Proceso para la gestión de fallas establecido por el NOC-UTPL.

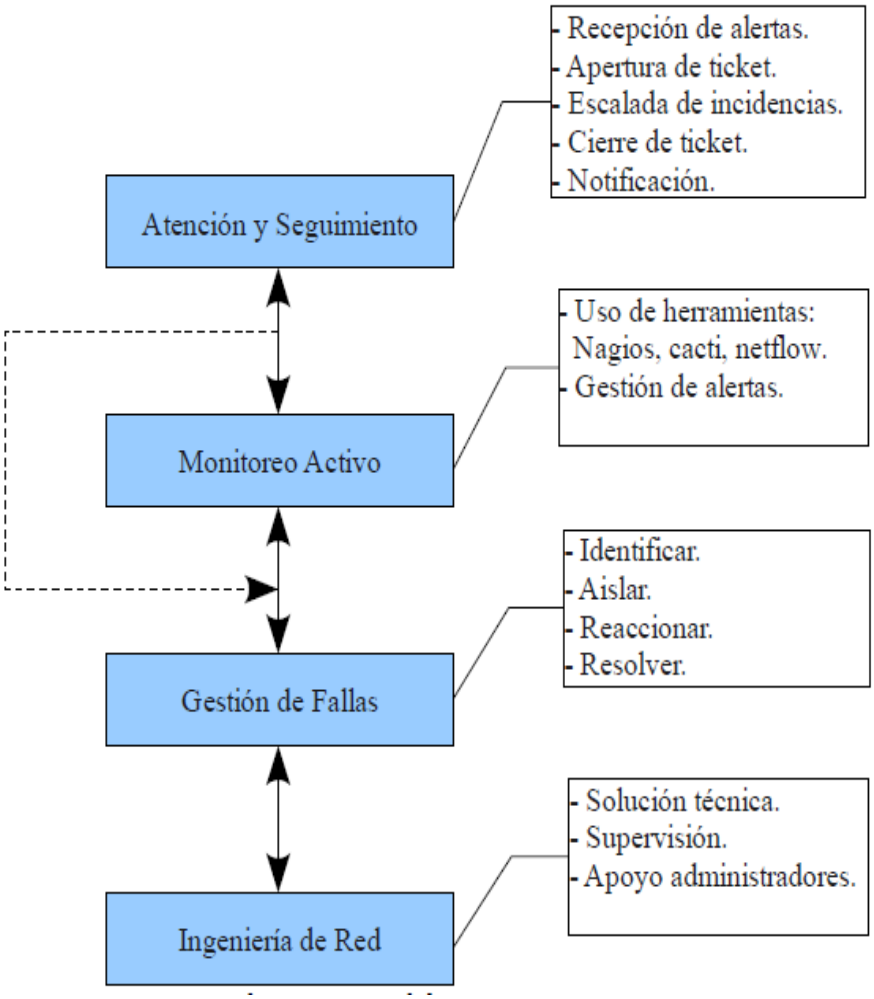

Figura 2-1 Esquema de operación del NOC-UTPL (UTPL, 2009)

2.4 ARQUITECTURA DE GESTIÓN DE REDES

El NOC-UTPL realiza una gestión integrada disponiendo de un centro de gestión para dar seguimiento y solución a los problemas de la red. Dejando de esta manera la gestión de la red centralizada en el NOC-UTPL desde donde se realizan las actividades de monitoreo.

2.5 HERRAMIENTAS Y PROTOCOLOS PARA LA GESTIÓN DE LA RED

Para la gestión de red, el NOC-UTPL cuenta las siguientes herramientas, las mismas que se encuentran instaladas en el servidor NOC

2.5.1 HERRAMIENTAS INSTALADAS

2.5.1.1 NAGIOS:

Nagios es un sistema de monitorización de equipos y de servicios de red, creado para ayudar a los administradores a tener siempre el control de lo que está pasando en la red y conocer los problemas que ocurren en la infraestructura que administran antes de que los usuarios de la misma los perciban, alertándonos cuando las cosas van mal y cuando se normalizan. (Cayuqueo, 2009). Nagios usa como protocolo de gestión NRPE[18].

Actualmente el NOC-UTPL utiliza Nagios para el monitoreo y emisión de alertas a través de mensajes a e-mail y SMS.

2.5.1.2 CACTI:

Cacti es una herramienta de gráficas de red, diseñado para aprovechar el poder de la rrdtool[19] almacenamiento de datos y la funcionalidad de gráficos.(Oz & E&M, 2009). Cacti usa SNMP v2 como protocolo de gestión

[18] **NRPE** (Nagios Remote Plug-in Executor), le permite ejecutar remotamente plugins de Nagios en otros equipos Linux / Unix.
[19] **RRDTOOL** Es un sistema para almacenar y presentar datos en series temporales Su propósito principal es crear una representación gráfica agradable

Cacti es usado para obtener datos estadísticos del uso de los canales de internet y recursos.

2.5.1.3 IPPLAN:

IPplan es una práctica herramienta para la administración de direcciones IP. IPplan puede manejar una sola red o atender a múltiples redes.

IPPLAN es utilizado para el registro de espacio de direccionamiento asignado a usuarios.

2.5.1.4 NETFLOW ANALIZER:

Según López (López, Vergara, Bellido, & Fernández, 2004), Manage Engine Netflow Analize (Manage Engine, 2011), es una herramienta de monitorización de ancho de banda basado en tecnología Web. Permite analizar la utilización de ancho de banda y ofrece visibilidad completa sobre routers y switches Cisco. Esta herramienta utiliza netflow para la gestión de red.

El NOC-UTPL utiliza Netflow Analizer para el monitoreo de enlaces de internet.

2.6 PROBLEMÁTICA

En base al estudio realizado al NOC-UTPL, se puede expresar que el centro de operaciones de red cuenta con el proceso de gestión de fallas aunque no está estandarizado, además poseen herramientas de monitoreo que trabajan de forma independiente, es decir cada cual tiene su propia funcionalidad, dejando a la vista la falta de un manual de procesos detallado y de aplicaciones nuevas de gestión de red, para enfrentar problemas como: la necesidad de acceder al entorno de cada

herramienta para la vigilancia y seguimiento de la red y la utilización de herramientas de gestión de red que están siendo obsoletas frente al surgimiento de nuevas aplicaciones para el monitoreo.

De acuerdo a lo antes vertido, el NOC de la UTPL plantea la necesidad de la implementación de un sistema de gestión de red integrado con sus respectiva documentación (manual de administrador y manual de usuario), que permita realizar un monitoreo de acuerdo a estándares y solventar las necesidades, además se propone la elaboración del manual de procesos con miras a formalizar el equipo NOC-UTPL.

3

CAPÍTULO 3. DEFINICIÓN DE LOS REQUERIMIENTOS

En base al estudio de algunos centros de gestión de red como: NOC-UNAM (UNAM, 1996) , NOC-CLARA (CLARA, 2008), (Ludwing, 2004), CORED (CORED, 2007) e información recolectada del NOC-UTPL se planteo ciertos requerimientos que se consideró necesarios e importantes para la elección del NMS. El presente capítulo está dedicado a la definición de los requerimientos que nos ayudaran en la elección de la herramienta.

3.1 ESPECIFICACIÓN DE REQUERIMIENTOS

Una vez realizadas las Investigaciones necesarias, se planteo una lista de los requerimientos para un NOC, se realizó un análisis en base a los objetivos y necesidades del NOC-UTPL llegando a la obtención de los requerimientos más relevantes que se resumen en la **Tabla 2-1**

COMPILACIÓN Y AUTOMATIZACIÓN DE UN SISTEMA DE GESTIÓN DE REDES NOC CON HERRAMIENTAS DE CÓDIGO ABIERTO.

SRS-ESPECIFICACIÓN DE REQUERIMIENTOS

N°	Nombre	Grado Necesidad
1	Monitoreo de servicios	alto
Descripción	El sistema deberá Monitorear los servicios (HTTP, POP3, SNMP, FTP).	

N°	Nombre	Grado Necesidad
2	Monitoreo de recursos	medio
Descripción	El sistema deberá Monitorear los recursos de los dispositivos(disco, memoria, CPU)	

N°	Nombre	Grado Necesidad
3	Monitoreo de conectividad	medio
Descripción	El sistema deberá Monitorear el estado del dispositivo (up, down)	

N°	Nombre	Grado Necesidad
4	Alertas y Notificaciones	alto
Descripción	El sistema deberá permitir la Generación de alertas basado en notificaciones que nos informen el estado de los servicios y dispositivos a través de un medio de comunicación	

N°	Nombre	Grado Necesidad
5	Descubrimiento de red	medio
Descripción	El sistema permitirá el Descubrimiento y visualización de la red junto con los dispositivos que la conforman	

N°	Nombre	Grado Necesidad
6	Reportes	alto
Descripción	El sistema deberá permitir la generación de reportes	

N°	Nombre	Grado Necesidad
7	Eventos	alto
Descripción	El sistema deberá permitir la gestión de eventos	

N°	Nombre	Grado Necesidad
8	Herramienta integrada	alto
Descripción	El sistema deberá estar integrado con otras herramientas de gestión sobresalientes en este campo.	

N°	Nombre	Grado Necesidad
9	Extensión de funcionalidades	medio
Descripción	La herramienta debe permitir extender sus funcionalidades, permitir la creación de paquetes para incrementar sus características.	

Tabla 3-1 Especificación de Requerimientos

En la **Tabla 2-1** se resumen los requerimientos que se detallan en la plantilla de especificación de requisitos (ver anexo 1).

4

CAPÍTULO 4. ESTUDIO Y SELECCIÓN DE LA HERRAMIENTA

Los NMS, son herramientas de software que ayudan al administrador de red a gestionar una red de datos. La decisión de elegir una herramienta para la gestión de red debe basarse en los requisitos y criterios de evaluación de un NMS que determinan qué es lo que se necesita tener y a partir de ahí buscar una aplicación que cumpla con estos requisitos. En este capítulo se abordara el estudio y selección del sistema de gestión de red partiendo de un análisis de las herramientas más relevantes en este campo para luego someterlas a una evaluación de criterios y obtener la solución más optima para la gestión de red

4.1 ANÁLISIS DE HERRAMIENTAS

A nivel de software de gestión de red se destacan herramientas como: Zenoss (Contreras, Aricapa, & Restrepo, 2008), Cacti (Sanz Tapia, Sanchez Cid, & Almorza Daza, 2006) (Hervey, 2008) (CACTI, 2009), Nagios (Dominguez Dorado

& Zarandieta Moran, 2003) (Gonzales, 2005), JFFNMS (Vega Tirado, Henao Alvarez, & Loaiza Garcia, 2008) (Padilla Jaume, Aris, & Fibla, 2007) (LINUX-MAGAZINE, 2009), OPManager (AdventNet Inc 5200, 2005), OSSIM (Madrid Molina, y otros, 2008), OPutils (AdventNet, 2008), Hiperic (Liguori De Gottig, 2009) (ITLinux, 2009) (Nance & World, 2007). A partir de las cuales se evalúan sus características y se obtiene una lista de las herramientas más relevantes que a continuación se detallan para un estudio más profundo

4.1.1 ZENOSS

Figura 4-1: Zenoss Core **(Zenoss, 2010)**

Zenoss es una herramienta de supervisión y administración de redes altamente automatizada. Desarrollada en Python[20] en el marco de un

[20] **Python** Lenguaje de programación orientado-objeto desarrollado por Guido van Rossum.

proyecto OpenSource[21], Zenoss ha sido desarrollada a partir de retazos de código libre: escrita en Python dentro de un entorno Zope[22], MySQL[23], RRDtool, CRicket, PySNMP, Net-SNMP, etc. Zenoss permite visualizar las relaciones entre cada elemento de tu red. Entre los protocolos utilizados, citamos SNMP, WMI[24] y Telnet[25]/SSH[26]. Los datos se registran en una base de datos exportable a XML[27], y el seguimiento de la actividad de la red se hace a través de tests SCMP y TCP programados. (Contreras, Aricapa, & Restrepo, 2008)

4.1.1.1 Características

- Funciona sobre Linux, FreeBSD , Mac OS X y VMWare Player
- Desarrollada en Python dentro de un entorno Zope, MySQL, RRDtool, CRicket, PySNMP, Net-SNMP, etc.
- Vigilancia de red.
- Brinda soporte para monitoreo de infraestructura de servidores virtuales.
- Corre sobre AJAX[28].
- Es muy flexible.
- Gestión de inventario y cambio.
- Posee Administración centralizada.

[21] **Open Source** Software libre y de código abierto, es el software que está licenciado de tal manera que los usuarios pueden estudiar, modificar y mejorar su diseño mediante la disponibilidad de su código fuente.

[22] **Zope** Es un servidor de aplicaciones libre, integra un servidor Web, un lenguaje de srcipt (Python) y un servidor de bases de datos.

[23] **MySQL** Es un sistema de gestión de bases relacionales (RDBMS), basado en SQL (Structured Query Language).

[24] **WMI** (Windows Management Instrumentation), es Una interfaz de programación de Windows que permite que el sistema y los dispositivos de red puedan ser configurados y administrados.

[25] **Telnet** Es un comando de usuario y una subyacente TCP / IP de protocolo de acceso a equipos remotos.

[26] **SSH** (Secure SHell). Protocolo seguro que sirve para acceder a máquinas remotas a través de una red.

[27] **XML** Son las siglas de Extensible Markup Language, una especificación/lenguaje de programación diseñado especialmente para los documentos de la web.

[28] **AJAX** acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones.

4.1.1.2 Ventajas

- Interfaz simple e intuitiva.
- Instalación fácil.
- Extensible: uso de zenpacks[29], Incluso con plugins[30] de Nagios
- Integración con Nagios, Cacti.
- Soporte directo de SNMP.
- Licencia GPL

4.1.1.3 Desventajas

- La instalación es monolítica.
- Hardware de altas prestaciones

Link de descarga:

Página principal: http://www.zenoss.com/

[29] **Zenpacks** Son grupos empaquetados de funciones y modelos de plantillas para tipos específicos de dispositivos.
[30] **Plugins** Software del módulo que añade una función específica o servicio a un sistema más grande.

4.1.2 CACTI

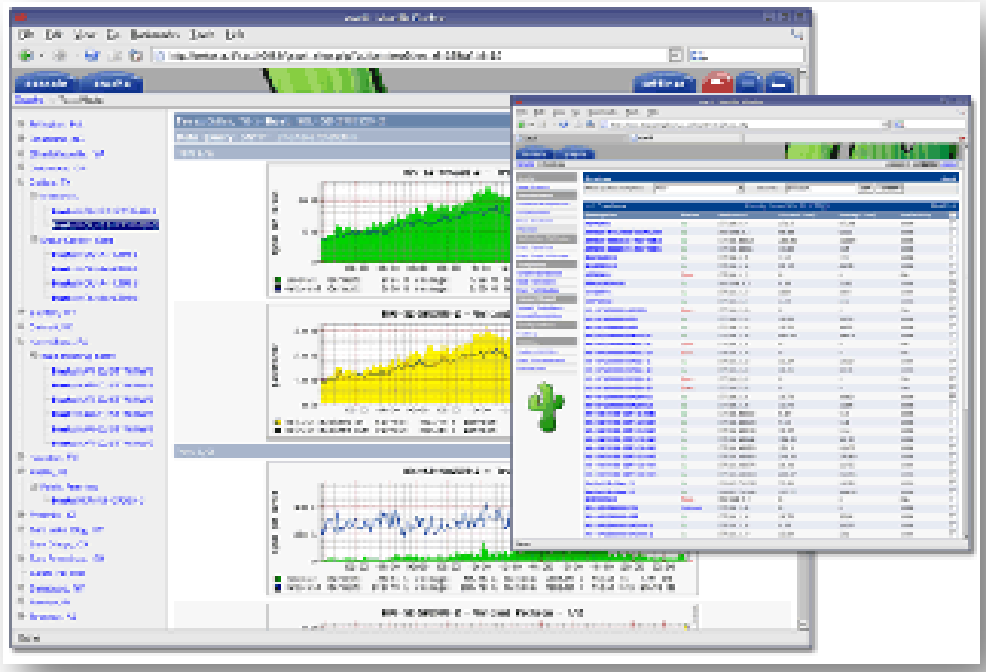

Figura 4-2 Cacti (Sanz Tapia, Sanchez Cid, & Almorza Daza, 2006)

Esta herramienta se utiliza para capturar información cualitativa sobre el estado de los dispositivos, Fue publicada en 1994, y desde entonces han sido desarrolladas nuevas versiones hasta la versión actual 4.2, esta herramienta accede a cada dispositivo y extrae información sobre su estado actual mediante consultas SNMP o mediante scripts[31] específicos escritos por el personal del Servicio de Informática. La flexibilidad de Cacti permite tener múltiples dispositivos monitorizados, de cada uno de los cuales se extrae información que se almacena en un archivo histórico RRA[32] (round robin archive). A partir de esos archivos se generan gráficos de manera sencilla y reutilizable mediante plantilla. (Sanz Tapia, Sanchez Cid, & Almorza Daza, 2006) CACTI es una herramienta que calcula los

[31] **Scripts** Conjunto de instrucciones generalmente almacenadas en un archivo de texto que deben ser interpretados línea a línea en tiempo real para su ejecución.

[32] **RRA** Son archivos de datos de una base de datos RRD.

principales parámetros de memorias cache como: tiempos de acceso, tiempos de ciclo, área de silicio y consumo tanto estático como dinámico.

4.1.2.1 Características

- Desarrollada en PHP[33]
- Está diseñada al rededor de RRDTool
- Usa MySQL como base de datos
- Soporta SNMP y MRTG
- Permite crear plantillas para reutilizar las definiciones de gráficos, fuentes de datos y dispositivos(Hervey, 2008), (CACTI, 2009).
- Permite medir la Disponibilidad, Carga y Errores de los dispositivos(Hervey, 2008), (CACTI, 2009).
- Permite utilizar todas las funciones de rrdgraph para definir los gráficos y automatiza algunas de ellas(Hervey, 2008), (CACTI, 2009).

4.1.2.2 Ventajas

- La principal ventaja que ofrece CACTI es que en un tiempo corto, y solamente a partir de parámetros arquitectónicos de la memoria cache (tamaño, asociatividad, número de puertos), es capaz de ofrecer resultados con márgenes de error dentro del 10% del resultado que se obtendría con SPICE. (Viictorio Juan, Torres Moreno, & Viñals Yufera, 2007)
- Multiplataforma.
- Cacti es muy flexible por su idea de plantillas.

[33] **PHP** Permite a los desarrolladores web, crear contenido dinámico que interactúa con las bases de datos

4.1.2.3 Desventajas

- La Configuración de Dispositivos es tediosa(Hervey, 2008).
- No es fácil hacer un “re-descubrimiento” de dispositivos.

Link de descarga:

Página principal: http://www.cacti.net.

4.1.3 NAGIOS

Figura 4-3 Nagios (Dominguez Dorado & Zarandieta Moran, 2003)

Nagios es un software usado en todo el mundo, que debe correr en sistemas Linux. Nagios es un sistema de monitorización de equipos y de servicios de red, creado para ayudar a los administradores a tener siempre el control de qué está pasando en la red y conocer los problemas que ocurren en la infraestructura que administran antes de que los

usuarios de la misma los perciban. (Dominguez Dorado & Zarandieta Moran, 2003)

4.1.3.1 Características(Gonzales, 2005) :

- Nagios está escrito en C y es Software Libre.
- Representación en mapas de los elementos monitoreados.
- Notificación de eventos.
- Consolas web y wap[34].
- Puede generar alertas vía e-mail, sms[35], pager[36] (beeper) o pop-ups[37].
- Genera reportes publicables en la web.
- Monitorización de servicios de red: SMTP, POP3, HTTP, SSH, D NS[38], etc.
- Monitorización de recursos: Carga de procesador, espacio libre e n filesystems, uso de la memoria, etc.
- Capacidad de desarrollar plugins de forma sencilla que permite a los usuarios programar sus propios chequeos. Flexibilidad!!
- Capacidad de definir una topología o jerarquía de red que permita distinguir entre servicios caídos o inaccesibles.

4.1.3.2 Ventajas:

- Es un sistema completo en cuanto a sus características que además hace uso en algunos casos de diversos sistemas como

[34] **Wap** hace referencia a las siglas de Wireless Application Protocool, y se trata de un estándar internacional para aplicaciones de comunicaciones de red a través de un entorno sin cables.
[35] **SMS** (Short Message Service) Servicio de Mensaje Corto. Es un servicio de mensajería por teléfonos celulares.
[36] **Pager** Es un pequeño dispositivo de telecomunicación donde se reciben mensajes que aparecen escritos en un display.
[37] **Pop-ups** Son ventanas no abiertas por el usuario que aparecen al acceder a una página.
[38] **DNS** (Domain Name System) Sistema de Nombres de Dominio. Conjunto de protocolos y servicios para la identificación/conversión de una dirección de internet expresada en lenguaje natural por una dirección IP.

por ejemplo sistemas gestores de bases de datos, servidores web, etc. (Dominguez Dorado & Zarandieta Moran, 2003)

- Permite extender su funcionalidad con la utilización/creación de extensiones. (Sanz Tapia, Sanchez Cid, & Almorza Daza, 2006)
- Está liberado bajo licencia GPL[39] de la Free Software Foundation.
- Muestra los resultados de la monitorización y del uso de los diversos componentes en una interfaz web a través de un conjunto de CGI's[40] y de un conjunto de páginas HTML[41] que vienen incorporadas de serie.(Sanz Tapia, Sanchez Cid, & Almorza Daza, 2006)
- Sigue el estado de múltiples servicios de red en múltiples servidores y avisar a personas o grupos de personas responsables de los mismos.
- Permite escaladas de avisos en función del tiempo de parada y otros parámetros, múltiples métodos de aviso (correo electrónico, SMS...) y presentación gráfica muy práctica del estado actual y del histórico de estados.(Sanz Tapia, Sanchez Cid, & Almorza Daza, 2006)

4.1.3.3 Desventajas:

- Es relativamente complejo de instalar y configurar
- No posee interfaz gráfica de administración
- Sin autodescubrimiento
- No posee Gráficos históricos

[39] **GPL** (General Public License). Licencia creada por la Free Software Foundation y orientada principalmente a los términos de distribución, modificación y uso de software libre

[40] **CGI's** (Common Gateway Interface). Tecnología de la WWW que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web.

[41] **HTML** Hyper Text Markup Language, es un lenguaje de programación muy sencillo que se utiliza para crear los textos y las páginas web

Link de descarga:

Página principal: http://www.nagios.org

4.1.4 JFFNMS

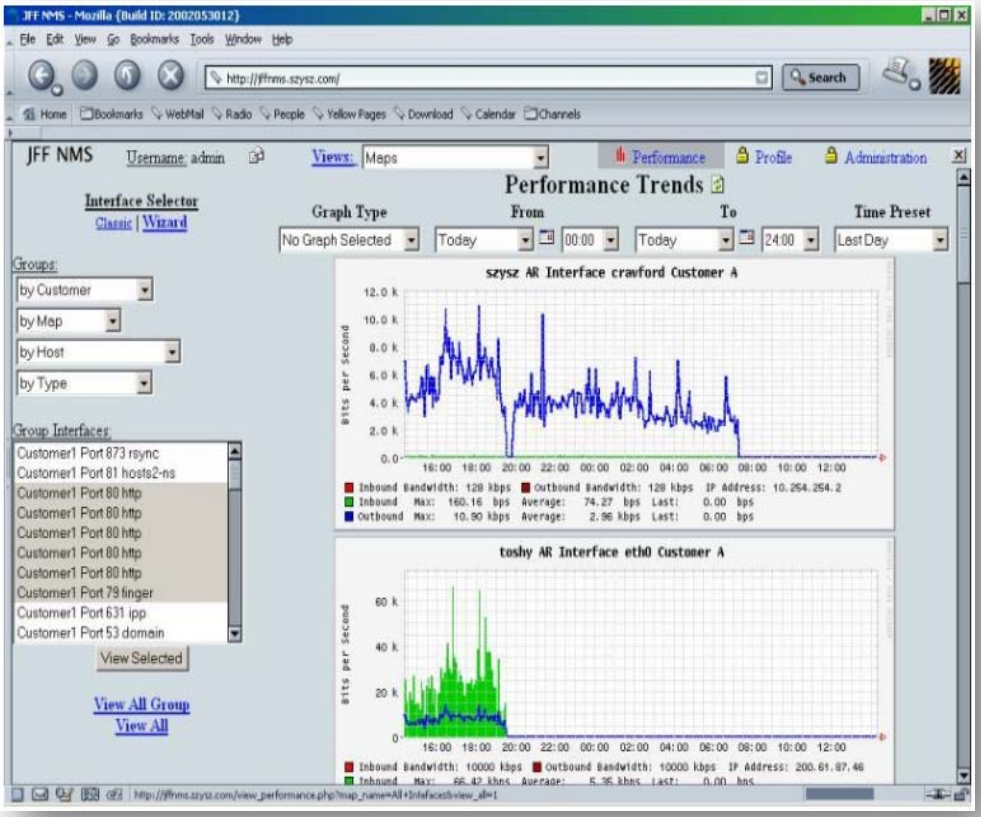

Figura 4-4 JFFNMS (Padilla Jaume, Aris, & Fibla, 2007)

JFFNMS está bajo licencia GPL, permite monitorizar una red IP mediante SNMP, Syslog[42] y Tacacs+. Puede ser utilizado para monitorizar cualquier dispositivo SNMP, servidor, router, puerto TCP o cualquier elemento que

[42] **Syslog** Es un servicio de registro de datos que es muy frecuentemente usado en entornos Linux y Unix.

se desee siempre que se programe una extensión adecuada a dicho elemento para JFFNMS.

JFFNMS está escrito en PHP y funciona en entornos GNU/Linux, FreeBSD y Windows 2000/XP. La consola de eventos muestra todos los tipos de eventos de manera ordenada en el mismo Display. JFFNMS genera gráficas para todos los dispositivos de la red tráfico de red, utilización de CPU[43], errores, etc. (Padilla Jaume, Aris, & Fibla, 2007)

JFFNMS es muy modular y extensible lo que significa que se pueden programar extensiones en caso de que no se disponga de soporte para los elementos específicos de la red. Se basa en las tecnologías: Apache, Cron, MySQL, PHP, RDDTool y SNMP. Dispone de un Mapa de Estado que permite visualizar la red de una manera sencilla. La lista de dispositivos, o "Tipos de Interfaz" que JFFNMS puede monitorizar es extensa, y gracias a sus desarrolladores y usuarios está creciendo. (Padilla Jaume, Aris, & Fibla, 2007)

4.1.4.1 Características (Vega Tirado, Henao Alvarez, & Loaiza Garcia, 2008):

- Permite monitorizar una red IP mediante SNMP.
- Puede ser utilizado para monitorizar cualquier dispositivo SNMP (servidor, router, puerto TCP y UDP[44]).
- JFFNMS está escrito en PHP, el cual funciona en Sistemas Operativos GNU/Linux, FreeBSD y Windows 2000/XP.
- Tiene soporte de base de datos (MySQL o PostgreSQL[45]), integra logs[46] de Syslog.

[43] **CPU** Central Processing Unit (unidad de proceso central, es el cerebro del ordenador.
[44] **UDP** Protocolo de Datagrama de Usuario, es un protocolo sin conexión que, funciona en redes IP, no establece un diálogo previo entre las dos partes, ni tampoco mecanismos de detección de errores
[45] **PostgreSQL** Es la base de datos open source más avanzada del mundo. Usada en empresas, universidades, instituciones públicas y privadas.
[46] **Logs** Archivo que registra movimientos y actividades de un determinado programa (log file).

- JFFNMS se basa en las tecnologías: Apache, Cron, MySQL, PHP, RDDTool y SNMP.
- Necesita la instalación y configuración del complemento (agente) SNMP en los clientes.
- Gráficos de resultados para todo, tráfico del interfaz, errores, uso de la CPU, etc.
- Base de datos (MySQL o PostgreSQL)
- Libre, bajo licencia GPL

4.1.4.2 Ventajas:

- JFFNMS es muy modular y extensible lo que significa que se pueden programar extensiones en caso de que no se disponga de soporte para los elementos específicos de la red (Padilla Jaume, Aris, & Fibla, 2007).
- Dispone de un Mapa de Estado que permite visualizar la red de una manera sencilla.
- La lista de dispositivos, o Tipos de Interfaz que JFFNMS puede monitorizar es extensa (Padilla Jaume, Aris, & Fibla, 2007).
- JFFNMS permite una administración del mismo mediante un entorno gráfico.

4.1.4.3 Desventajas:

- Tolera scripting multi sitio XSS[47], lo que permite a atacantes remotos inyectar un script web arbitrario o código HTML a través del parámetro de usuario. (LINUX- MAGAZINE, 2009)
- Existen múltiples aberturas para la inyección de SQL[48] en auth.php cuando se encuentra deshabilitada la opción

[47] **XSS** Cross-Site-Scripting Problema de seguridad en las páginas web, generalmente por vulnerabilidades en el sistema de validación de datos entrantes.

magic_quotes_gpc. Un atacante remoto podrá ejecutar comandos SQL arbitrarios. (LINUX- MAGAZINE, 2009)

- Un problema en el script admin/ setup.php permite a atacantes remotos leer y modificar las opciones de configuración. (LINUX-MAGAZINE, 2009)

Link de descarga:

Página principal: http://www.jffnms.org/#

4.1.5 OSSIM

Figura 4-5 Ossim **(Madrid Molina, y otros, 2008)**

[48] **SQL** (Structured Query Language) Lenguaje utilizado para base de datos, SQL es un lenguaje de definición de datos (LDD), un lenguaje de definiciones de vistas (LDV) y un lenguaje de manipulación de datos (LMD).

OSSIM (Open Source Security Information Management), es una distribución de productos Open Source integrados para construir una infraestructura de monitorización de seguridad.

Su objetivo es ofrecer un marco para centralizar, organizar y mejorar las capacidades de detección y visibilidad en la monitorización de eventos de seguridad de la organización.

Es una consola de seguridad de código abierto que une las cualidades de NTOP, MRGT, ARPWATCH entre muchos otros, Tiene la capacidad de consolidar alertas de una gran cantidad de sistemas de seguridad basados en código abierto, y es altamente configurable, de tal manera que permite procesar información de programas y dispositivos de seguridad. La arquitectura de OSSIM es distribuida y comprende cuatro elementos básicos. (Madrid Molina, y otros, 2008)

Elementos de captura de información: Recolectan la información requerida por OSSIM, en los diferentes sitios del sistema informático en donde se desea hacer control.

Base de datos: Almacena todos los eventos recibidos de los diferentes elementos de captura de información, así como las alarmas generadas por el motor de correlación del servidor.

Servidor: El servidor correlaciona los eventos registrados en la base de datos, con el fin de detectar patrones que evidencien una vulnerabilidad en el sistema o un ataque informático, y a la vez actúa como filtro para tratar de eliminar la mayor cantidad posible de falsos positivos. Además, con base en las alarmas que se presenten y en el valor de importancia relativa que el administrador haya asignado a cada uno de los activos informáticos de la empresa, OSSIM es capaz de calcular también el nivel de riesgo informático del negocio.

Consola de gestión: La consola es el front-end[49] gráfico del sistema. Funciona vía web, y permite al administrador del sistema consultar las alarmas, reportes y estadísticas que genera el sistema. (Madrid Molina, y otros, 2008)

4.1.5.1 Características

- Arpwatch, utilizados para la detección de anomalía de mac.
- P0f, utilizadas para los sistemas operativos, detección y análisis de los cambios.
- Pads, utilizados para el servicio de detección de anomalías.
- Nessus, utilizada para la evaluación de la vulnerabilidad y de correlación cruzada (IDS vs Security Scanner).
- Snort, el IDS, también se utiliza para cruzar la correlación con nessus.
- Spade, la estadística de paquetes de motor de detección de anomalías. Se usa para obtener conocimientos sobre los ataques sin firma.
- Tcptrack, utilizado para los datos de la sesión de información que puede dar información útil para el ataque correlación.
- Ntop, que construye una impresionante red de base de datos de información que podemos obtener de detección de anomalías de comportamiento aberrante.
- Nagios. Que se alimentan de la base de datos de activos de acogida supervisa la disponibilidad del servicio de acogida y de información.
- Osiris, un gran HIDS.
- OCS-NG, Cruz-Plataforma inventario solución.
- OSSEC, la integridad, de rootkit, detección y registro de más.

[49] **Front-end** Hace referencia al estado inicial de un proceso, es responsable de recoger entradas de los usuarios, y ser procesadas de tal manera que cumplan las especificaciones.

4.1.5.2 Ventajas

- Integra diferentes aplicaciones de seguridad reconocidas.
- Permite realizar análisis forense con los eventos almacenados.
- Tiene soporte de una comunidad abierta mundial en crecimiento constante.

4.1.5.3 Desventajas

- Maneja tanta información que a veces la forma de presentarla no resulta demasiado intuitiva.
- Solo se encargan de almacenar los eventos y reportarlos, no realiza ninguna acción para detener los ataques.

Link de descarga:

Página principal: http://www.ossim.net/

4.1.6 HYPERIC

Figura 4-6 Hyperic (Liguori De Gottig, 2009)

Hyperic HQ es una herramienta que nos permite realizar el seguimiento, análisis y monitoreo de gran cantidad de componentes de nuestra red. Posee un amplio campo de trabajo soportando diversas bases de datos (Tomcat, MySQL, PostgreSQL, etc.), infinitésimos SO y otras aplicaciones generales no menos importantes (Apache, JBoss y demás). (Liguori De Gottig, 2009)

4.1.6.1 Características

- Desarrollada en java y C.
- soporta diversas bases de datos Tomcat, MySQL, PostgreSQL.
- Autodescubrimiento.
- Monitoreo en tiempo real.
- Administración de eventos.
- Análisis y generación de reportes.
- Implementación rápida.
- Control completo.
- Seguridad empresarial.

4.1.6.2 Ventajas (ITLinux, 2009)

- Auto-Descubre todo el software y hardware para hacer inventarios con un solo clic.
- Monitoriza distintos tipos de productos y/o tecnologías incluyendo Servidores Web, Servidores de Aplicaciones, Bases de Datos, Servidores de Correo, Dispositivos de Red, entre otros, bajo más de 9 tipos distintos de Sistemas Operativos.
- Realiza un seguimiento del performance, configuración y cambios de seguridad.
- Maximiza la disponibilidad con alertas y acciones de control correctivas para corregir problemas antes de que estos ocurran.

- Extensible, y personalizable para poder manejar de manera única lo que tu empresa necesita.

4.1.6.3 Desventajas (Nance & World, 2007)

- Documentación y soporte en español deficiente.
- La remediación es un proceso muy manual.

Link de descarga:

Página principal: http://www.hyperic.com/

4.2 SELECCIÓN DEL SISTEMA DE GESTIÓN DE RED

Para la selección del sistema se realizó los siguientes pasos:

- Identificar los requerimientos del NOC-UTPL.
- Investigar, consultar y realizar una lista los sistemas más óptimos para la gestión de red.
- Seleccionar el sistema apropiado de un conjunto de alternativas, en base a los criterios y a los requerimientos especificados (Ver anexo # 1).

En la **Tabla 3-1** se presenta las herramientas seleccionadas y los criterios de evaluación.

CRITERIOS DE EVALUACIÓN	*NAGIOS*	*CACTI*	*OSSIM*	*HYPERIC*	*JFFNMS*	*ZENOSS*
Descubrimiento automático		✓		✓	✓	✓
Control de usuario	✓	✓	✓	✓	✓	✓
Mecanismos de notificación	✓	✓	✓	✓		✓
Registro y manipulación de eventos	✓		✓	✓	✓	✓
Generación de reportes	✓		✓	✓	✓	✓
Extensión de funcionalidades	✓				✓	✓
Utilización de protocolos estándar	✓	✓	✓		✓	✓
Interfaz web	✓	✓	✓	✓	✓	✓
Tipo de licencia	Open Source	Open Source	Open Source	Open Source	Open Source	Open Source
Mapa de la topología	✓	✓	✓		✓	✓
Grafica de datos de rendimiento		✓		✓	✓	✓
Base de datos	MySQL	MySQL	MySQL	tomcat, MySQL, PostgreSQL	MySQL o PostgreSQL	MySQL,
Integración con herramientas	✓		✓			✓

Tabla 4-1 Evaluación de herramientas

Basándose en la **Tabla 3-1** y según características y criterios expuestos de cada herramienta se eligió la herramienta de Zenoss para su implementación en el presente proyecto.

Observando la **Tabla 3-1** podemos concluir que existen 2 herramientas: Zenoss y Nagios que cumplen con 2 de las especificaciones principales que son la integración de herramientas y la extensión de funcionalidades, de estas dos se eligió Zenoss por ser la herramienta que además de cumplir estos requerimientos, cumple con todos los criterios de selección, mantiene una comunidad activa dedicada a la generación de Zenpacks que incrementa constantemente sus funcionalidades, utiliza protocolos estándar para la gestión de la red, posee flexibilidad para agregar funcionalidades y genera reportes potentes, entre otras.

5

CAPÍTULO 5. IMPLEMENTACIÓN DEL NMS

Anteriormente se definió que la herramienta a implementarse a nivel de gestión de red es Zenoss, la misma que será instalada en un servidor con sistema operativo Linux distribución Ubuntu, la razón de la elección de la distribución Ubuntu es por ser la distribución que actualmente posee el servidor NOC en producción, su facilidad de manejo, disponibilidad de paquetes y facilidad de adaptación.

En este capítulo se abordarán temas relacionados con la descripción de la aplicación a nivel general, además se realiza una evaluación de la solución para verificar el cumplimiento de los requerimientos, implementación de Zenoss en un entorno de pruebas, preparación del entorno en producción, instalación y configuración, problemas resueltos culminando con las pruebas y monitoreo respectivo.

5.1 DESCRIPCIÓN DE LA APLICACIÓN

Zenoss está basado en las siguientes tecnologías de software libre:

- **Zope:** Servidor de aplicaciones orientadas a objetos, trabajado en la web escrito en Python.
- **Python:** extensible lenguaje de programación.
- **Twisted:** Un evento impulsado por motor de la creación de redes escrito en Python.
- **NetSNMP:** protocolo monitoreo que recolecta información sobre la situación de los sistemas.
- **RRDtool:** Gráfico y registro de datos de series temporales.
- **MySQL:** Una base de datos de código abierto.

Luego del proceso de selección, se obtuvo la aplicación de gestión de red Zenoss, constituida por 19 pestañas organizadas en 4 secciones:

Vistas principales

- Dashboard
- Event console
- Devices list
- Network map

Clases

- Events
- Devices
- Services
- Processes
- Products

Navegar por

- Systems
- Groups
- Locations
- Networks
- Reports

Administración

- Add devices
- MIBs
- Collectors
- Settings
- Event Manager

Figura 5-1 Ventana principal de Zenoss.

Una vez lista la aplicación se generó documentación como:

- Manual de administrador
- Manual de usuario

5.2 CUMPLIMIENTO DE REQUERIMIENTOS

En la **Tabla 4-1** se detallan todos los requerimientos planteados en el presente proyecto y se especifican las características y funcionalidades por medio de las cuales Zenoss cumple con cada uno de estos requerimientos.

CÓDIGO	REQUERIMIENTO	DETALLE
REQ-01	Monitoreo de servicios	Zenoss permite monitorear servicios de los dispositivos tales como: SNMP, HTTP, FTP, etc.
REQ-02	Monitoreo de recursos	Zenoss permite el monitoreo de: • Hardware: memoria, disco, uso de la CPU. • Software: aplicaciones instaladas en el dispositivo. • Sistema Operativo. • Interfaces, etc.
REQ-03	Monitoreo de conectividad	Zenoss cumple este requerimiento mediante la página status de cada dispositivo, por medio de la cual permite conocer el estado del dispositivo
REQ-04	Alertas y Notificaciones	Este requerimiento se lo cumple a través del envío de correos electrónico,

		mensajes a celular por parte del servidor de Zenoss a los administradores de los dispositivos que generan inconvenientes y por tanto generan alertas.
REQ-05	Descubrimiento de red	Para el cumplimiento de este requerimiento la herramienta proporciona el descubrimiento automático y agregación manual de dispositivos de la red así como también el descubrimiento automático de las subredes existentes, junto a esto nos permite la visualización de la red y sus dispositivos en un entorno Flash (uso de Flash Player para la visualización de la red).
REQ-06	Reportes	Una de las características principales de Zenoss es la generación de reportes por disponer de una gran variedad de reportes como: • Reportes de dispositivos. • Reportes de eventos • Reportes de gráficas • Reportes de multigráficas • Reportes de rendimiento • Reportes de usuarios
REQ-07	Eventos	Para el cumplimiento del requerimiento, Zenoss permite el manejo de los eventos provenientes de los dispositivos monitoreados con la generación de notificaciones correspondientes, administración, monitoreo y creación de

		eventos, además permite ordenar y filtrar los eventos.
REQ-08	Herramienta integrada	Zenoss es una herramienta de monitoreo de redes y servidores integrada con otras herramientas como Cacti y Nagios (ambas Open Source) que permite a los administradores tener un control completo sobre la infraestructura de red.
REQ-09	Extensión de funcionalidades	Este requerimiento se cumple mediante la instalación de Zenpacks que nos permite extender constantemente las funcionalidades de Zenoss gracias al aporte de la comunidad activa que posee.

Tabla 5-1 Cumplimiento de requerimientos

5.3 IMPLEMENTACIÓN EN UN ENTORNO DE PRUEBA

Inicialmente se implementó la herramienta Zenoss en una máquina de pruebas para realizar las configuraciones y estudio correspondiente, esta implementación se realizó en una computadora con características limitadas, en la cual se efectuó la instalación, configuración y resolución de errores presentados.

En la máquina de pruebas se instaló Zenoss por dos veces, cada una de diferente manera, la primera se la realizó desde el código y la segunda desde repositorios, dejando como resultado errores con la instalación de la primera mientras que la segunda tuvo éxito manteniendo un nivel bajo de errores, finalmente se optó por la instalación desde repositorios debido a investigaciones

y soporte mediante correo electrónico por parte de personas inmersas y con un conocimiento y alta experiencia en Zenoss.

En la máquina de pruebas se instaló dos versiones de Zenoss la 2.5.1 y posteriormente la 2.5.2, esto fue debido a que durante la primera implementación la versión estable era la 2.5.1 pero durante el transcurso de la implementación de la herramienta, se libera la versión 2.5.2 lo cual produjo la implementación de Zenoss en esta última versión.

Los resultados de su implementación fueron exitosos. Debido a las características y propiedades de la máquina de pruebas en donde se instaló Zenoss dio como resultado un sistema lento para consultas, descubrimientos de red, agregación de equipos, búsqueda, etc.

5.3.1 PROBLEMAS RESUELTOS

Durante la implementación de Zenoss en la máquina de pruebas se presentaron ciertos errores, los cuales fueron resueltos satisfactoriamente gracias a investigaciones en la web, foros y consultas a personas con un mayor conocimiento de la herramienta. A continuación se detallan cada uno de los errores con su respectiva descripción y solución correspondiente

TIPO	DESCRIPCIÓN	OBSERVACIÓN	SOLUCIÓN

ERROR DE GOOGLE MAPS	Al hacer clic sobre alguna locación de Google Maps nos presenta una pantalla de error.	Documentación completa del error en el anexo 2 de errores.	1. Loguearse como usuario Zenoss ***su Zenoss***
			2. Digitar **zendmd** e ingresar las siguientes líneas
			>>> dmd.ZenLinkManager.layer3_catalog.manage_catalogClear();
			>>> commit();
			3. Digitar lo siguiente:
			zenmigrate --step=ReindexIpAddressNetworkIds
			
			La solución está disponible en el siguiente enlacee: **http://forums.zenoss.com/viewtopic.php?p=22959**

PROBLEMAS CON LA INTERFAZ WEB DE ZENOSS	La interfaz web de Zenoss no muestra absolutamente ninguna información, no realiza la conexión con el servidor, al momento de digitar la URL visualiza una página en blanco.	Existen otros archivos con extensión .zec que pueden ocasionar problemas posteriores, para evitar inconvenientes podemos eliminar estos archivos con el mismo comando expuesto en la solución. Documentación completa del error en el anexo 2 de errores.	1. Loguearse como usuario Zenoss ***su Zenoss*** 2. Limpiar la cache de zope • `zopectl stop` • `rm $ZENHOME/var/zeo1-1.zec` • zeoctl start La solución está disponible en el siguiente enlace: **http://www.sysadminwiki.net/wiki/index.php?title=Common_Zenoss_Errors_-_Post_Install**
INSTALACION DE ZENPACKS VIA UI	No permite la instalación de zenpacks. Al momento de instalar zenpacks desde la UI de Zenoss, visualiza lo siguiente Output:	Actualmente no hay solución, la comunidad de Zenoss se encuentra resolviendo el problema pero hasta la fecha no se presenta solución pero dan una alternativa instalar el	**ALTERNATIVA** Instalación manual del Zenpack utilizando SSH: 1. Loguearse en el servidor como un usuario mediante ssh 2. Cambiarse de usuario a usuario Zenoss haciendo:

	zenpack --install /tmp/ZenPacks.zenoss.ApacheMonitor-2.1.0-py2.4.egg /bin/sh: zenpack: not found Done installing ZenPack.	Zenpack desde consola. Documentación completa del error en el anexo 2 de errores	***su zenoss*** 3. Moverse al lugar en donde descargamos los Zenpacks ***cd /ruta_de_los_zenpacks_descargados*** 4. Usamos el siguiente comando para instalar los Zenpacks seguido del nombre del zenpack a instalar ***zenpack --install ZenPacks.zenoss.ApacheMonitor-2.1.0-py2.4.egg.*** Para asegurarnos de la instalación de Zenpack, se ubica en la consola web de Zenoss, en Settings→Zenpacks y se puede visualizar el Zenpack instalado. Si observamos que el Zenpack es reportado como Broken o missing es porque necesita que el servidor se reinicie para tomar los cambios efectuados. Para esto nos conectamos con el servidor vía ssh y como usuario root digitamos lo siguiente: • /etc/init.d/zenoss-stack stop • /etc/init.d/zenoss-stack start El ticket creado para este problema está en el siguiente enlace: **http://dev.zenoss.com/trac/ticket/6250**

Tabla 5-2Errores durante la implementación de Zenoss

5.4 IMPLEMENTACIÓN EN UN ENTORNO DE PRODUCCIÓN

5.4.1 PREPARACIÓN DEL ENTORNO

Dentro de esta sección se abordarán temas relacionados con las especificaciones del servidor en producción, los prerrequisitos que posee la aplicación para su funcionamiento y sistemas operativos soportados por Zenoss.

5.4.1.1 ESPECIFICACIONES DEL SERVIDOR

En la **Tabla 5-1** se detallan las características de hardware y software que posee el servidor en donde se instalará Zenoss.

HARDWARE	SOFTWARE
Espacio disponible en disco: 80GB **Memoria**: 1GB **Procesador:** Intel(R) Xeon(TM) CPU 3.06GHz .	**Sistema Operativo:** Ubuntu

Tabla 5-3 Especificaciones del servidor

Las especificaciones de servidor dependen de la cantidad y frecuencia de datos a colectar. Zenoss recomienda las siguientes especificaciones de hardware (Zenoss, 2010):

- Redes con 250 dispositivos.
  - 4 GB RAM.
  - Core 2 Duo E6300 1.86/1066 RTL.
  - 75 GB en disco de almacenamiento.
- Redes con más de 250 dispositivos.
  - 8 GB RAM.

- XEON 5120 DC 1.86/1066/4MB.
- Cuatro discos de 75 GB in two RAID-1 pairs.

Actualmente la red de la Universidad Técnica Particular de Loja cuenta con un número aproximado de 173 dispositivos, por lo cual se recomienda tomar en cuenta las especificaciones de hardware emitidas anteriormente para redes con 250 dispositivos.

5.4.1.2 SISTEMAS OPERATIVOS SOPORTADOS

Zenoss soporta los siguientes sistemas operativos:


Red Hat Enterprise Linux 4, 5


CentOS 4, 5


Fedora 9, 10

SUSE Enterprise 10


openSUSE 10.3, 11.1


Debian 5


Ubuntu Server 6.06, 8.04


Mac OS X 10.5, 10.6

5.4.2 INSTALACIÓN Y CONFIGURACIÓN

Existen cuatro formas de instalación de Zenoss Core:

- Virtual Appliance
- Desde repositorios

- Instalación Binaria y
- Mediante la compilación del código.

Una instalación Virtual Appliance es elegida cuando se requiere evaluar o demostrar Zenoss y es utilizado para supervisar redes pequeñas con pocos dispositivos.

Desde repositorios es elegida cuando no se necesita descargar todo, sólo lo que realmente se necesita instalar, su instalación es fácil, es menos propensa a fallos.

Una instalación Binaria es utilizada para evitar la creación de la fuente de Zenoss.

Realizar una instalación desde el código nos permite crear una fuente de Zenoss con una variedad de entornos basados en UNIX, Ubuntu O MAC OS. Este tipo de instalación nos da la posibilidad de instalar Zenoss en el medio ambiente que elijamos pero su instalación es más tediosa y propensa a fallos.

De los cuatro tipos de instalación se ha elegido la instalación desde repositorios por su facilidad de instalación, por mantener un nivel bajo de errores y por ser la más recomendada por parte personas inmersas y con un conocimiento y alta experiencia en Zenoss, este tipo de instalación se detalla paso a paso en los manuales de la herramienta.

5.5 PRUEBAS Y MONITOREO

5.5.1 PRUEBAS

5.5.1.1 Rendimiento del Servidor

Para la realización de estas pruebas se tomó datos del servidor del Noc-utpl manteniendo los servicios de Zenoss en un estado bajo y luego se procedió a levantar los servicios de Zenoss en el servidor para tomar

datos del funcionamiento del servidor y así poder realizar un análisis de los datos recolectados.

Para recolectar los datos del servidor se hizo uso del comando Top, presentando a continuación los datos recolectados del servidor del NOC-UTPL manteniendo todos los servicios de Zenoss en un estado Down (apagado).

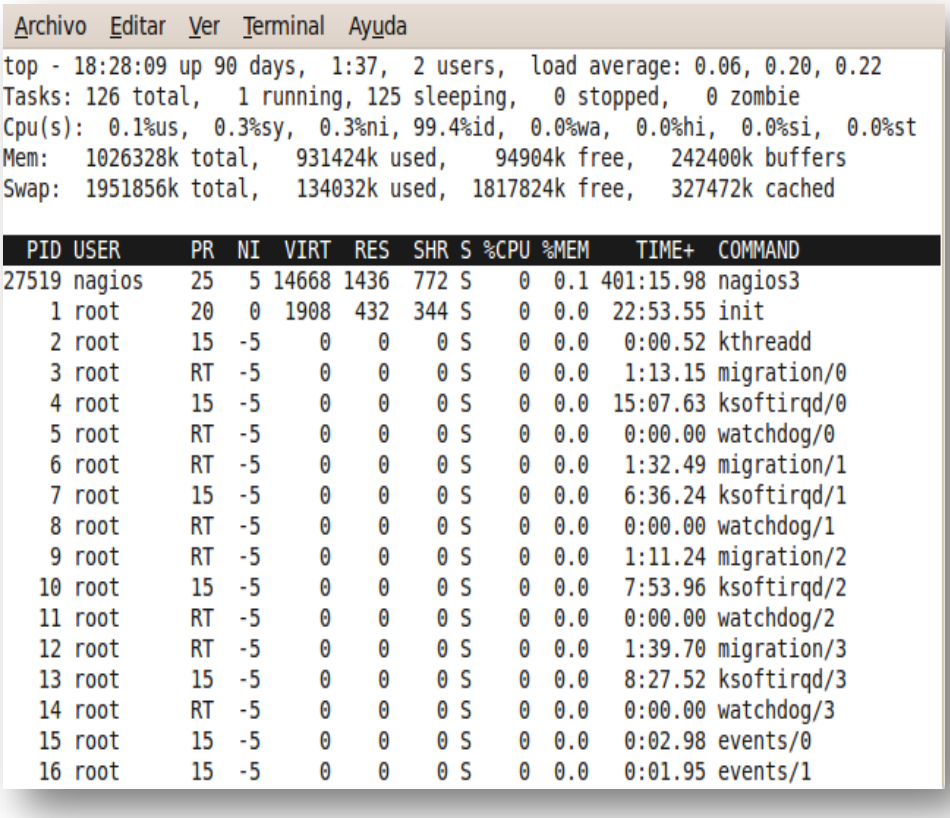

Figura 5-2 Rendimiento del servidor con Zenoss deshabilitado (comando ntop)

Los datos recolectados se los interpreta de la siguiente manera:

Carga media

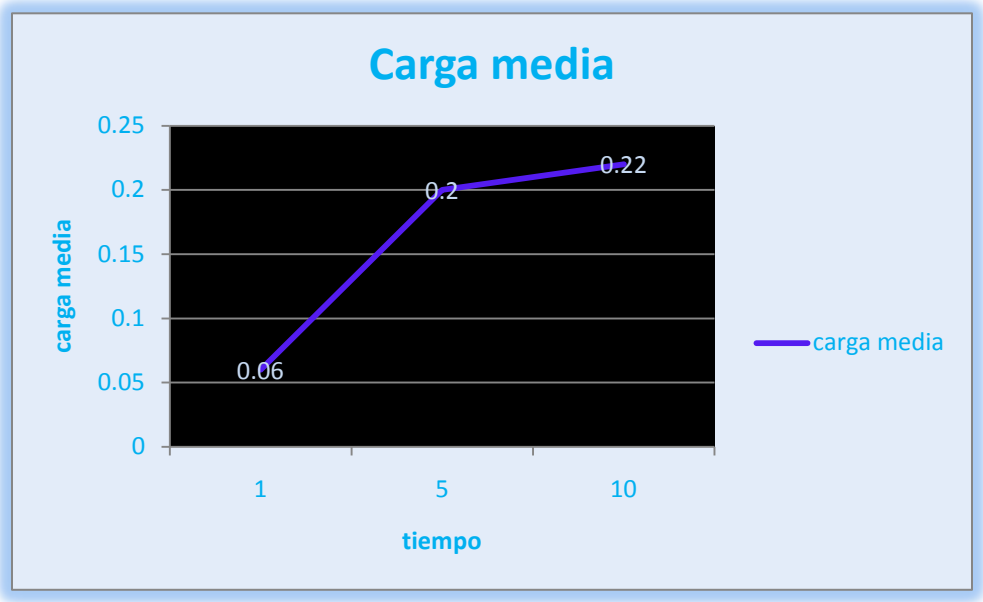

Figura 5-3 Carga media del servidor sin Zenoss

- La carga media del servidor es de “0.06, 0.20, 0.22”.
- Durante el último minuto, la CPU posee una carga promedio de 0.06 con 0.06 procesos en cola.
- Durante los últimos 5 minutos, la CPU posee una carga promedio de 0.20 con 1 procesos en cola.
- Durante los últimos 15 minutos, la CPU posee una carga promedio de 0.22 con 3,3 procesos en cola.

Tareas

Figura 5-4 Tareas existentes en el servidor sin Zenoss

- Actualmente existen 126 tareas en total en el servidor dividiéndose de la siguiente manera: activas (1), detenidas (0), dormidas (125) y zombis (0).

CPU

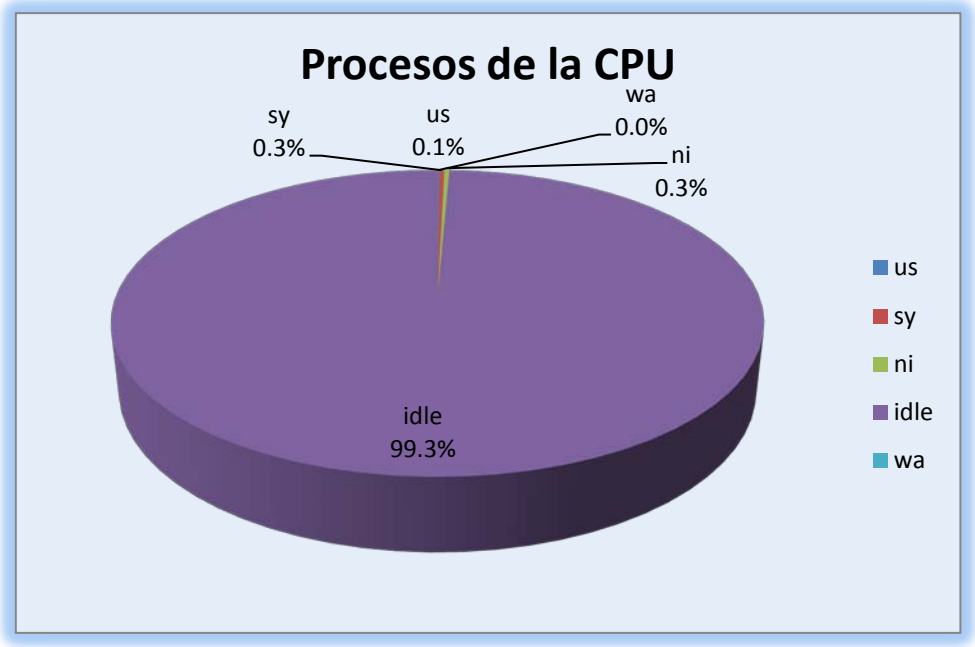

Figura 5-5 Procesos de la CPU del servidor sin Zenoss

- Procesos iniciados por algún usuario (us): 0.1%.
- Procesos iniciados por el sistema (sy):0.3%.
- Procesos iniciados con prioridad especial (ni): 0.3%.
- Porcentaje sin usar (id, idle):99.3%
- Procesos esperando para continuar (wa, waiting):0.0%

Memoria

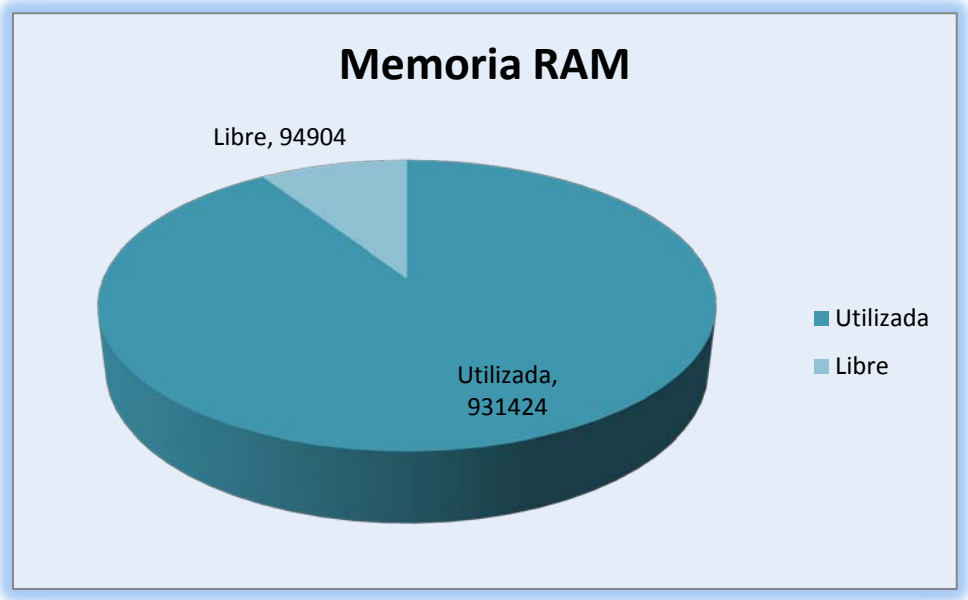

Figura 5-6 Memoria RAM del servidor sin Zenoss

- Memoria RAM total: 1026328k
- Memoria utilizada:931424k
- Memoria libre:94904k
- Memoria utilizada como buffers: 242400K

Swap

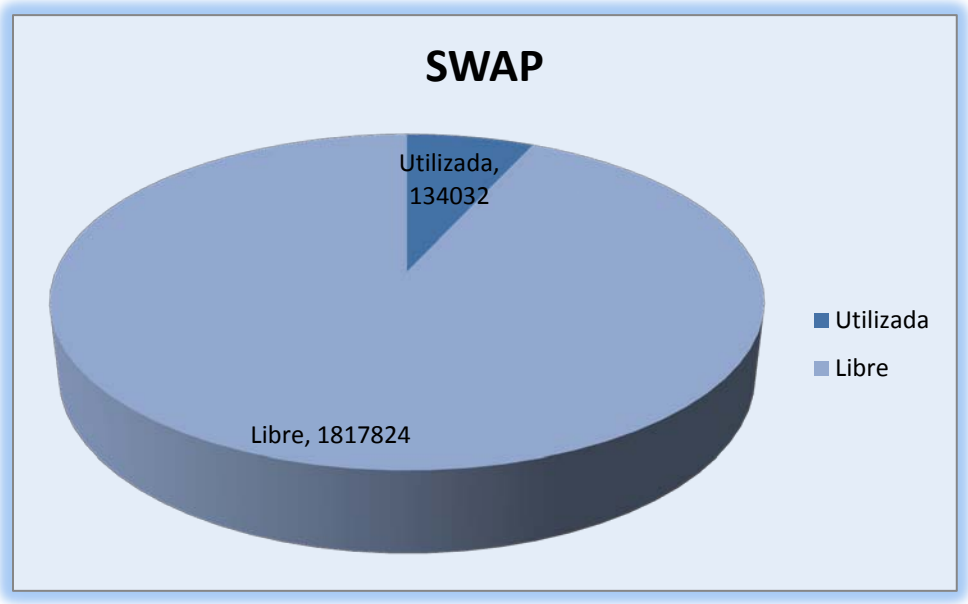

Figura 5-7 Memoria SWAP del servidor sin Zenoss

- Espacio de swap total:1951856k
- Swap utilizado:134032k
- Swap libre:1817824k
- Swap utilizado por páginas cacheadas:327472k

Procesos

Analizando la sección de procesos permite expresar lo siguiente: están corriendo diferentes procesos, iniciados por sus respectivos usuarios, hasta el momento existen dos usuarios Nagios y root. De todos los procesos visualizados, el proceso iniciado por Nagios es el que utiliza un mayor porcentaje de memoria física con un 0.1%, utiliza 1436k de memoria RAM física, indica que este proceso está en un estado de sleeping(S).

Una vez recolectado los datos del servidor se levantaron todos los servicios y demonios de Zenoss

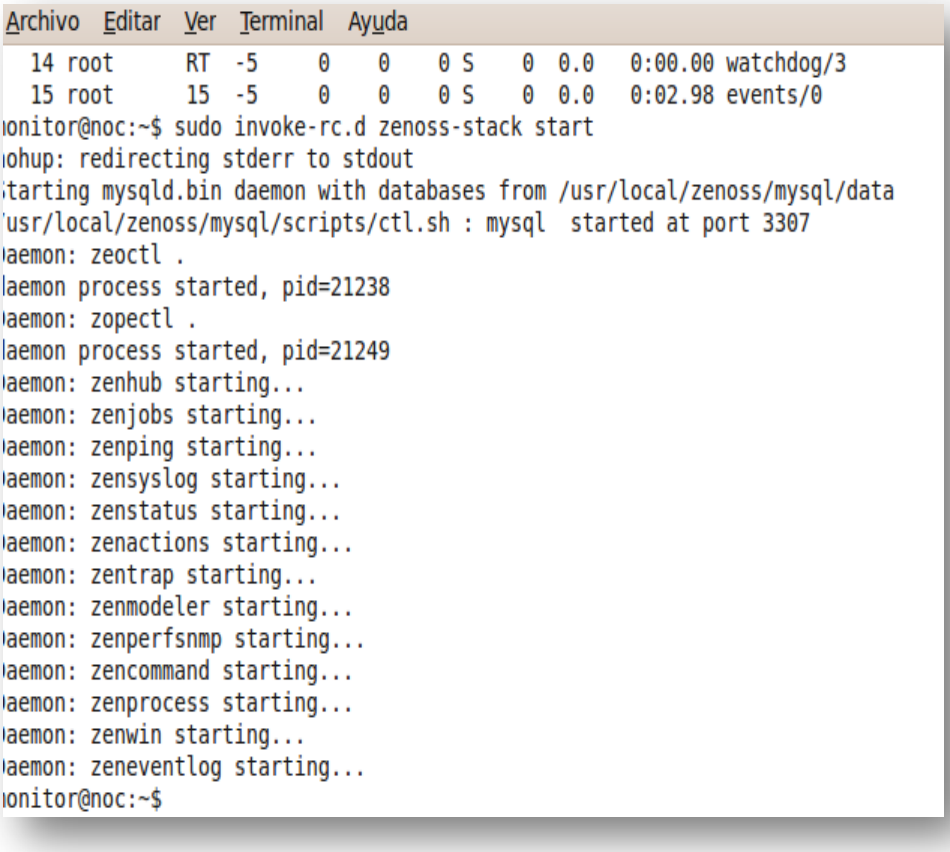

Figura 5-8 Comando para levantar Zenoss en el servidor

A continuación se presentan datos recolectados del servidor del NOC-UTPL con todos los servicios y demonios de Zenoss en un estado Up (Alto).

```
Archivo  Editar  Ver  Terminal  Ayuda
top - 18:31:02 up 90 days,  1:40,  2 users,  load average: 3.72, 1.55, 0.70
Tasks: 149 total,   2 running, 147 sleeping,   0 stopped,   0 zombie
Cpu(s): 25.1%us,  0.4%sy,  0.0%ni, 49.9%id, 24.6%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   1026328k total,  1010800k used,    15528k free,    52824k buffers
Swap:  1951856k total,   134476k used,  1817380k free,   238248k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
21299 zenoss    20   0 78772  50m 3628 R  100  5.1   0:45.69 .python.bin
10477 root      20   0     0    0    0 D    1  0.0   0:01.14 pdflush
 2753 snmp      20   0  9076 1684  800 S    0  0.2  33:17.95 snmpd
21072 mysql     20   0 99.3m  23m 4788 S    0  2.3   0:01.68 mysqld.bin
22378 www-data  20   0  8980 1748 1356 D    0  0.2   0:01.12 rrdtool
    1 root      20   0  1908  404  316 S    0  0.0  22:53.58 init
    2 root      15  -5     0    0    0 S    0  0.0   0:00.52 kthreadd
    3 root      RT  -5     0    0    0 S    0  0.0   1:13.16 migration/0
    4 root      15  -5     0    0    0 S    0  0.0  15:07.70 ksoftirqd/0
    5 root      RT  -5     0    0    0 S    0  0.0   0:00.00 watchdog/0
    6 root      RT  -5     0    0    0 S    0  0.0   1:32.50 migration/1
    7 root      15  -5     0    0    0 S    0  0.0   6:36.26 ksoftirqd/1
    8 root      RT  -5     0    0    0 S    0  0.0   0:00.00 watchdog/1
    9 root      RT  -5     0    0    0 S    0  0.0   1:11.24 migration/2
   10 root      15  -5     0    0    0 S    0  0.0   7:54.02 ksoftirqd/2
   11 root      RT  -5     0    0    0 S    0  0.0   0:00.00 watchdog/2
   12 root      RT  -5     0    0    0 S    0  0.0   1:39.70 migration/3
```

Figura 5-9 Rendimiento del servidor con Zenoss habilitado (comando ntop)

Los datos recolectados se los interpreta de la siguiente manera:

Carga media

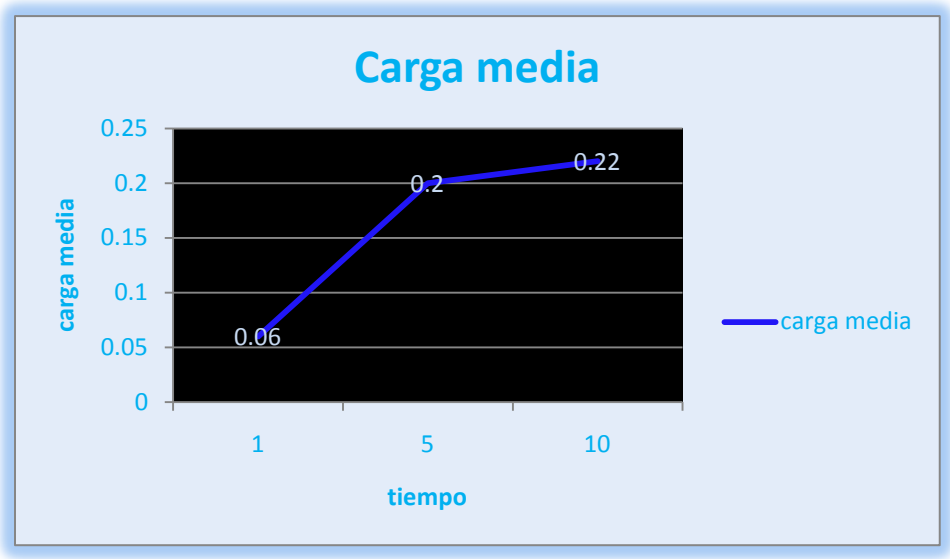

Figura 5-10 Carga media del servidor con Zenoss levantado

- La carga media del servidor es de “3.72, 1.55, 0.70”.
- Durante el último minuto, la CPU posee una carga promedio de 3.72 con 3.72 procesos en cola.
- Durante los últimos 5 minutos, la CPU posee una carga promedio de 1.55 con 7,75 procesos en cola.
- Durante los últimos 15 minutos, la CPU posee una carga promedio de 0.70 con 10.5 procesos en cola.

Tareas

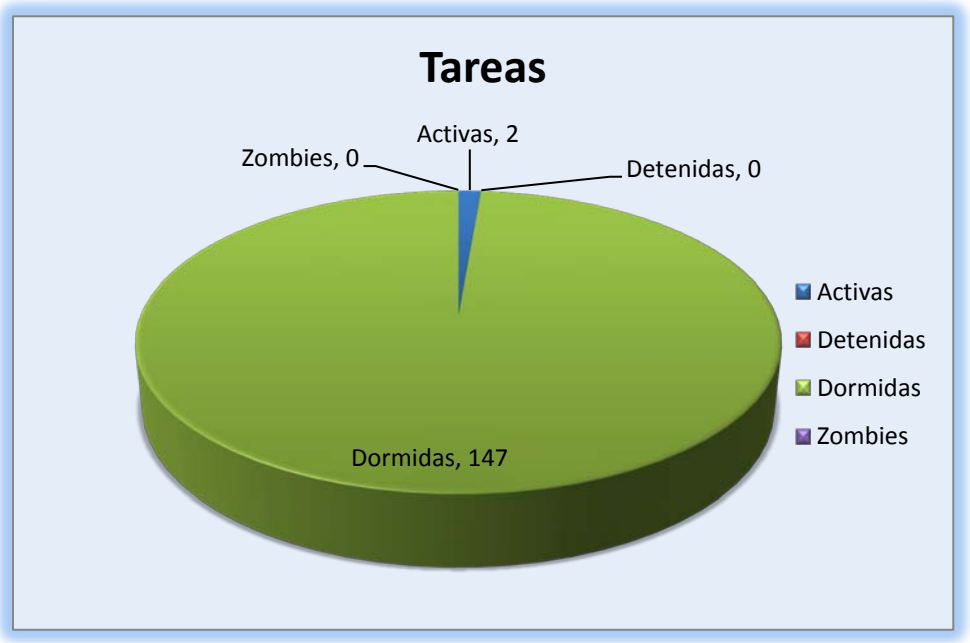

Figura 5-11 Tareas del servidor con Zenoss levantado

- Actualmente existen 149 tareas en total en el servidor dividiéndose de la siguiente manera: activas (2), detenidas (0), dormidas (147) y zombis (0).

CPU

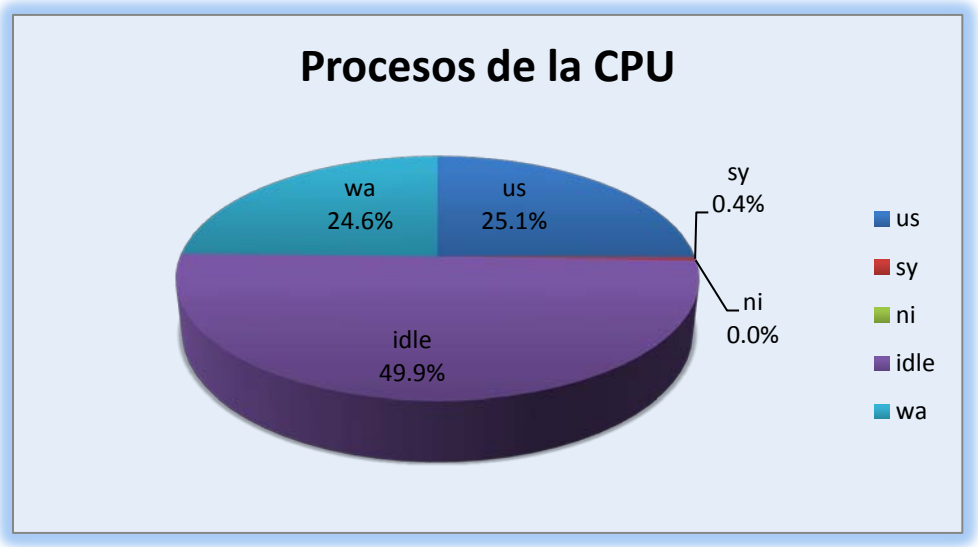

Figura 5-12 Procesos de la CPU del servidor con Zenoss levantado

- Procesos iniciados por algún usuario (us): 25.1%.
- Procesos iniciados por el sistema (sy):0.4%.
- Procesos iniciados con prioridad especial (ni): 0.0%.
- Porcentaje sin usar (id, idle):49.9%
- Procesos esperando para continuar (wa, waiting):24.6%

Memoria

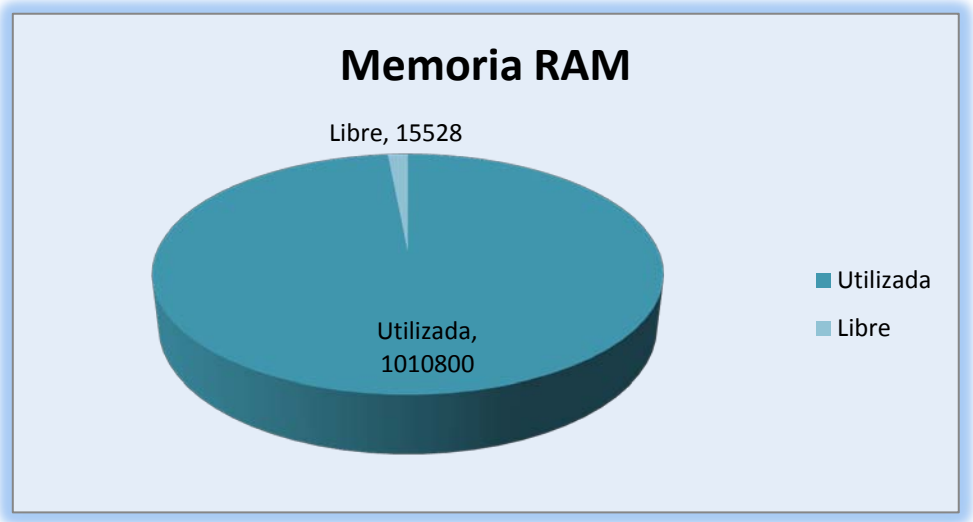

Figura 5-13 Memoria RAM del servidor con Zenoss levantado

- Memoria RAM total: 1026328k
- Memoria utilizada:1010800k
- Memoria libre: 15528k
- Memoria utilizada como buffers: 52824K

Swap

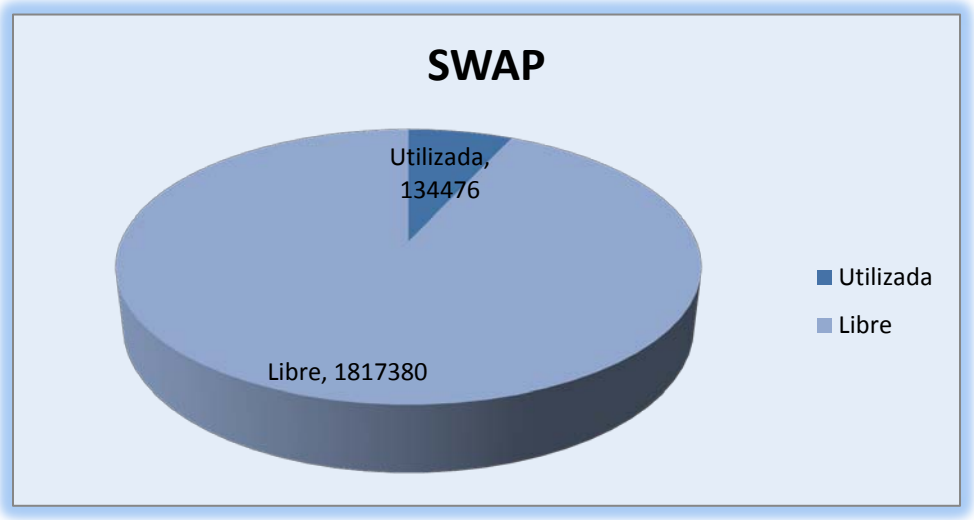

Figura 5-14 Memoria SWAP del servidor con Zenoss levantado

- Espacio de swap total:1951856k
- Swap utilizado:134476k
- Swap libre:1817380k
- Swap utilizado por páginas cacheadas:238248k

Procesos

Analizando la sección de procesos permite expresar lo siguiente: están corriendo diferentes procesos, iniciados por sus respectivos usuarios, hasta el momento existen usuarios como: Zenoss, MySQL, SNMP y root, de todos los procesos visualizados el proceso iniciado por Zenoss es el que utiliza un mayor porcentaje de memoria física con un 5.1%,

utiliza 50 MB de memoria RAM física, indica que este proceso está en un estado runing(R).

Las aplicaciones que actualmente se están ejecutando en el servidor NO-UTPL son: Nagios, CACTI, IPPLAN, MySQL y Zenoss.

	Zenoss(Down)	Zenoss(Up)
Carga media		
Load average	0.06,0.20,0.22	3.72,1.55,0.70
Tareas		
Total	126	149
Activas	1	2
Detenidas	0	0
Dormidas	125	147
Zombis	0	0
CPU		
us	0.1%	25.1%
sy	0.3%	0.4%
ni	0.3%	0.0%
id	99.4%	49.9%
wa	0.0%	24.6%
Memoria		
Total	1026328k	1026328k
Utilizada	931424k	1010800k
Libre	94904k	15528k
buffers	242400K	52824K
SWAP		
Total	1951856k	1951856k
Utilizada	134032k	134476k
Libre	1817824k	1817380k
cache	327472k	238248k

Tabla 5-4 Comparativa de las características del servidor NOC-UTPL

Según los datos recolectados el servidor del NOC-UTPL tiene un incremento de 3.66 procesos en cola con Zenoss levantado; una adicion de 26 tareas en total; el uso del CPU sin Zenoss es del 0.6%, mientras que su uso con Zenoss levantado llega al 50.1%.

5.5.2 MONITOREO DE DISPOSITIVOS

Durante el monitoreo de dispositivos, Zenoss recolecta información de los mismos, la cual es almacenada organizándola en pestañas.

Esta sección pretende realizar el monitoreo de 4 tipos de dispositivos: dispositivo Linux, dispositivo Windows, Switch[50] y Router. Para cada tipo de dispositivo se revisa las pestañas dedicadas para el monitoreo como son: status, OS[51], hardware, software, events, perf.

5.5.2.1 LINUX

Para este caso se dispone de un servidor Linux que ya se ha agregado en Zenoss con sus respectivos plugins. Para su correspondiente monitoreo, Zenoss hace uso de agente SNMP instalado en el equipo a monitorear para la obtención de información relevante del equipo.

Las diferentes características recolectadas de los dispositivos dependen en gran parte a la clase de dispositivos que un equipo es asignado debido a la disposición de templates diferentes de acuerdo a la clase.

Empezaremos ubicándonos en la página principal de status correspondiente al dispositivo.

[50] **Switch** Es el dispositivo analógico que permite interconectar redes operando en la capa 2 o de nivel de enlace de datos del modelo OSI.

[51] **OS** Operating System, es un conjunto de programas que permiten la comunicación del usuario con una computadora.

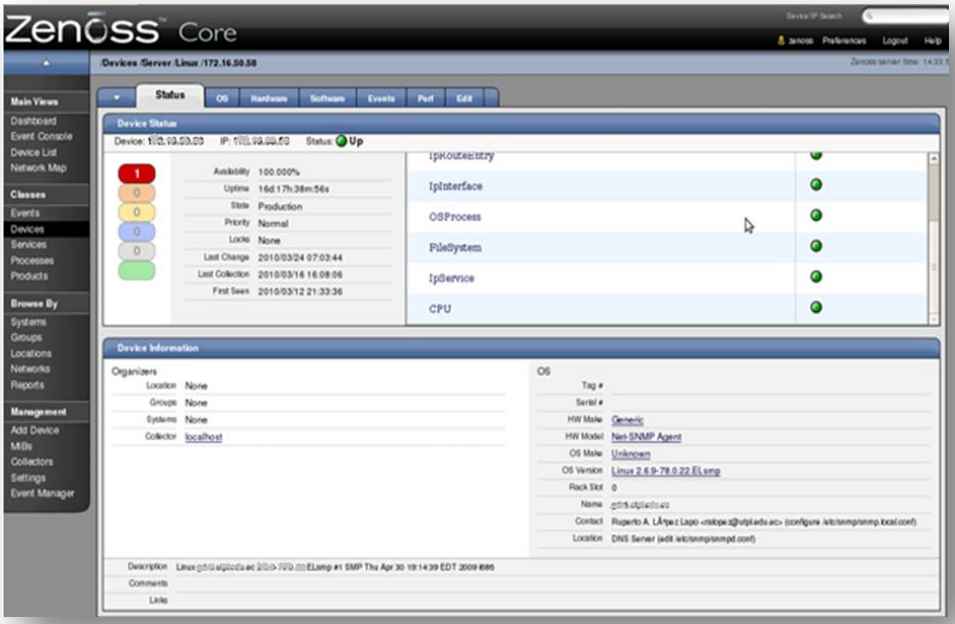

Figura 5-15 Pestaña Status del dispositivo

La página principal del dispositivo visualiza información recolectada del dispositivo organizándola mediante las siguientes pestañas.

STATUS

La pestaña Status (ver Anexo 3, Figura 0-1) se encuentra dividida en tres secciones: device status (estado del dispositivo), device information (información del dispositivo) y descripción.

La sección de device status nos permite conocer características como:

- El nombre del dispositivo, la dirección IP, el estado del dispositivo UP (prendido) o DOWN (apagado).
- El tiempo que lleva el dispositivo encendido.
- La disponibilidad, la prioridad, el estado del dispositivo entre otras características.

- Muestra el estado de los archivos del sistema, información de la CPU, procesos, servicios e interfaz, entre otros componentes.

La sección de device information se encuentra divida por organizadores y sistema operativo(OS).

- En la parte de organizadores muestra las cuatro clases que posee Zenoss con su respectiva información del dispositivo a monitorear, aquí obtendremos información a cerca de a qué grupo locación sistema y colector al que pertenece dicho dispositivo, en este caso solo mantiene información del colector al que pertenece (Localhost).
- En la parte de OS muestra información de el sistema operativo instalado, el tipo de agente instalado, el tipo de hardware, el nombre del dispositivo, información de contacto la locación, etc.

La sección de descripción muestra:

- Un resumen de las características recolectadas más importantes del dispositivo.

OS

OS (Interfaces): En el Anexo 3, **Figura 0-2** se visualiza las interfaces disponibles del dispositivo, en este caso el dispositivo cuenta con cuatro interfaces dos de las cuales se encuentra en un estado activo: la eth0 con la IP perteneciente a su red y su Mac[52] respectiva y la lo que es la loopback del dispositivo con IP 127.0.0.1.

[52] **MAC** (Media Access Control o control de acceso al medio) es una dirección que posee un identificador de 48 bits que corresponde de forma única a una Ethernet de red.

Se puede monitorear cada una de la interfaces haciendo clic en la interfaz deseada, permitiendo visualizar las siguientes gráficas por cada interfaz.

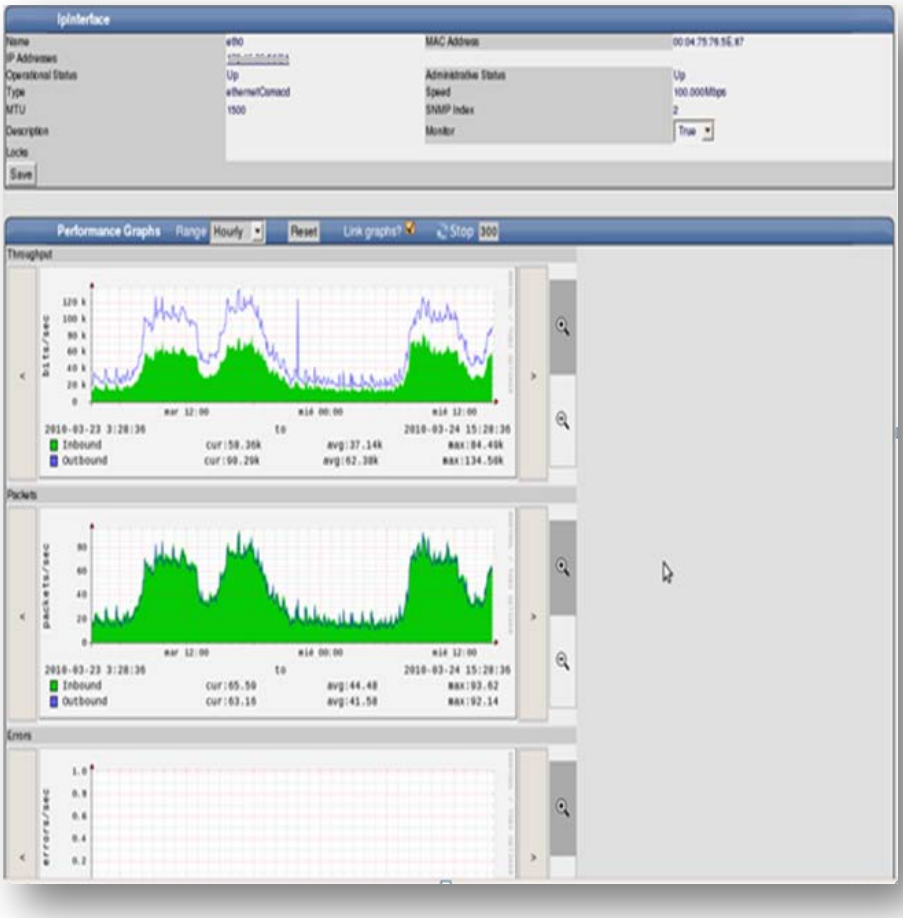

Figura 5-16 Gráficas de las interfaces de un dispositivo Linux

- En la parte de status se puede dar cuenta que la interfaz está en un estado operacional UP, estado de administración UP, la velocidad de la interfaz es de 100.000Mbps y que está siendo monitoreada (monitor – true).
- En la parte de gráficas de rendimiento se observa que existe un tráfico saliente de más de 130 K bits por segundo y un tráfico entrante de 84 K bits por segundo, de 85 a 94

paquetes por segundo entrantes, de 82 a 92 paquetes por segundo salientes y ningún error en el tráfico.

OS(OS processes): En el Anexo 3, **Figura 0-2** visualiza los procesos que se están monitoreando en nuestro caso el usr/sbin/sshd, mostrando el estado del proceso en este caso el color verde nos indica que el proceso está corriendo en el dispositivo y también nos informa que el procesos se está monitoreando.

Se puede monitorear los procesos mediante graficas dando un clic sobre el proceso.

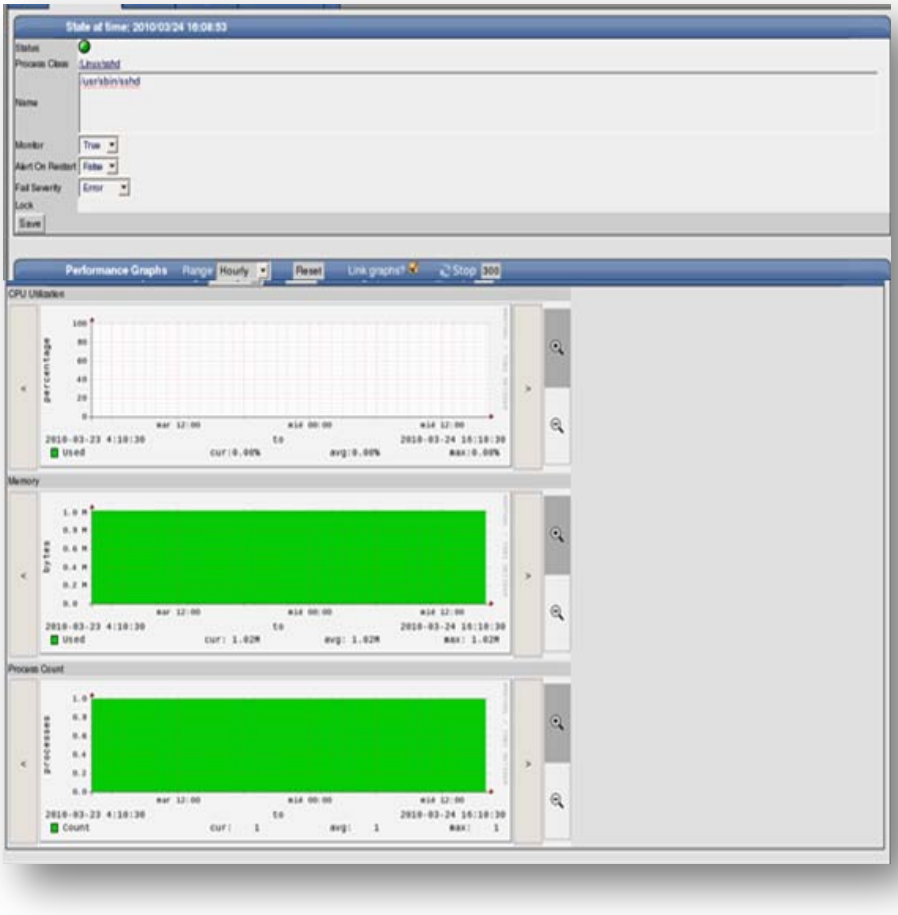

Figura 5-17 Gráficas de los procesos de un dispositivo Linux

Las gráficas nos muestran un 0 % del porcentaje de utilización de la CPU, 1.0 M de memoria utilizada, y el numero de procesos que está corriendo en el dispositivo es 1.

OS (File systems): nos muestra el tamaño total en bytes, libre, ocupado, el porcentaje utilizado de una partición o punto de montaje del equipo.
Observando la **Figura 0-2** ubicada en Anexo 3, en la sección de File systems tenemos 6 particiones: /, /backup, /boot, /tmp, /var/log, /var/named, en el directorio raíz tenemos un total de 6.5 GB, 2.1 GB usados, 4.4 GB libres y un total de 32% utilizados.

OS (Routes): muestra las rutas que toma la interfaz para salir a la red.

HARDWARE

Según la **Figura 0-3** del Anexo 3, el dispositivo tiene las siguientes características en lo que se refiere a hardware

- El dispositivo posee 1.2 GB de memoria aleatoria y 2.6 GB de memoria SWAP.
- El modelo de la unidad de procesamiento del equipo es: 2 procesadores Intel modelo Genuine Intel_ Intel(R) Pentium(R) III CPU family 1266MHz con una velocidad de 1266MHZ.

EVENTS

Observando la **Figura 0-4** del Anexo 3 se puede decir que hasta el momento se ha registrado un evento critico sobre el estado de un servicio monitoreado. El servicio Domain está bajo en otras palabras no está corriendo o funcionando en el dispositivos.

PERF

De la **Figura 0-5** del Anexo 3, se recoge los siguientes datos:

Load Average

- En un minuto tenemos 0.26 de carga de procesos.
- En 5 minutos tenemos 0.13 de carga de procesos.
- En 15 minutos tenemos 0.04 de carga de procesos.

CPU Utilization

- Porcentaje de espera de 0.0%.
- Porcentaje de usuario de 0.0%.
- Porcentaje del sistema en general 0.1%.
- Porcentaje sin utilizar 99.9%.

Memory Utilization

- Porcentaje de memoria usada 42.8 %.
- Porcentaje en el Buffer 7.4 %.
- Porcentaje de memoria cache 27.7 %.
- Porcentaje de memoria swap 0.0%.

5.5.2.2 WINDOWS

Para este caso se dispone de un equipo Windows que ya se ha agregado en Zenoss con sus respectivos plugins. Para su correspondiente monitoreo, Zenoss hace uso de agente SNMP instalado

en el equipo a monitorear para la obtención de información relevante del equipo.

Para empezar se ubica en la página principal de status correspondiente al dispositivo.

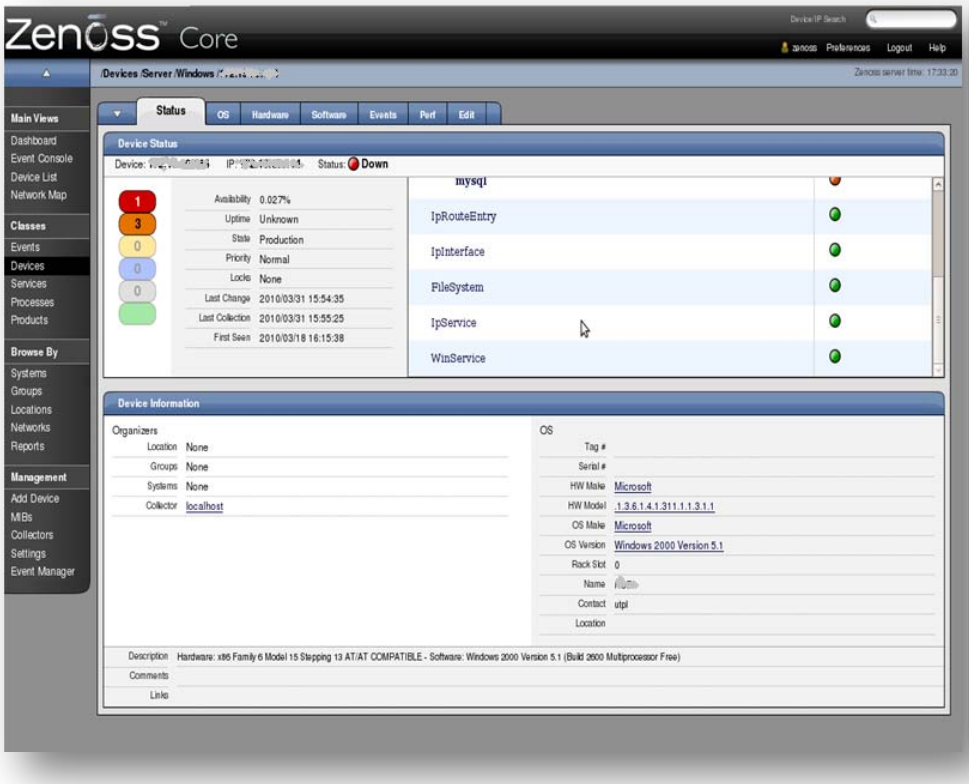

Figura 5-18 Página Status de un dispositivo

La página principal del dispositivo visualiza información recolectada del dispositivo organizándola mediante las siguientes pestañas.

STATUS

La pestaña status (ver Anexo 3, **Figura 0-6**) despliega un conjunto de las principales características recolectadas del dispositivo agregado distribuidas de la siguiente manera:

La sección de device status permite conocer características como:

- El nombre del dispositivo, la dirección IP, el dispositivo se encuentra DOWN (apagado).
- El tiempo que lleva el dispositivo encendido es de 0 seg.
- La disponibilidad, la prioridad, el estado del dispositivo entre otras características.
- Muestra el estado de algunos componentes como: archivos del sistema, información de e la CPU, procesos, servicios e interfaz, entre otros componentes.

La sección de device information se encuentra dividida por organizadores y sistema operativo(OS).

- En la parte de organizadores se muestra las cuatro clases organizadoras que posee Zenoss con su respectiva información del dispositivo a monitorear, aquí se obtiene información a cerca de a qué grupo locación sistema y colector pertenece dicho dispositivo, en este caso solo mantiene información del colector al que pertenece Localhost.
- En la parte de OS muestra información de el sistema operativo instalado; Windows 2000, versión 5.1, el tipo de agente instalado, el tipo de hardware: Microsoft, el nombre del dispositivo Nori, información de contacto la locación, etc.

En la sección de descripción muestra:

- Un resumen de las características recolectadas más importantes del dispositivo.

OS

OS (Interfaces): En el Anexo 3, **Figura 0-7** se visualiza las interfaces disponibles del dispositivo, en este caso el dispositivo cuenta con tres interfaces, dos de las cuales se encuentra en un estado activo: la Broadcom NetLink(TM) FastEthernet-minipuerto del administrador de paquetes con su IP, dirección de red y su Mac respectiva y la lo que es la lopback del dispositivo.

Se ´puede monitorear cada una de la interfaces haciendo clic en la interfaz deseada

OS(OS processes): En el Anexo 3, **Figura 0-7** visualiza los procesos que se están monitoreando en nuestro caso el MySQL, mostrando el estado del proceso (Status: color rojo que indica que el proceso no está corriendo en el dispositivo) y también informa que el proceso se está monitoreando (M: color verde que indica que el proceso si está corriendo en el dispositivo)

OS (winservice): Para los dispositivos Windows, Zenoss posee una utilidad más "Win Services" cuyo propósito es gestionar diferentes servicios que puedan correr sobre cualquier Windows cliente o servidor. En base a la **Figura 0-7** del Anexo 3, el dispositivo posee dos servicios: Aplication Management, y Event Log

OS (IP Service): basándose en la **Figura 0-7** del Anexo 3 se puede expresar que en el dispositivo se está monitoreando servicios IP tales como: Http, netbios-ssn, https, con protocolo tcp, en los puertos 80, 139,443 respectivamente, los mismos que se encuentran en estado activo.

OS (File systems): muestra el tamaño total en bytes, libre, ocupado, el porcentaje utilizado de una partición o punto de montaje del equipo.

Observando la **Figura 0-7** ubicada en Anexo 3, en la sección de File systems se tiene 2 unidades de montaje: la unidad C con un total de 97.7 y la unidad D con un total de 36.1 GB.

OS (Routes): En esta parte se puede observar las direcciones IP que tiene la interfaz y su encaminamiento en la red.

HARDWARE

En la Pestaña hardware se observa la capacidad de la memoria volátil y la swap en GB.
Según la **Figura 0-8** del Anexo 3, el dispositivo tiene las siguientes características en lo que se refiere a hardware:

- El dispositivo posee 2.0 GB de memoria aleatoria y memoria SWAP desconocida.

SOFTWARE

La **Figura 0-9** del Anexo 3 permite apreciar los nombres y los fabricantes de los programas instalados en esta máquina así como su fecha de instalación.

EVENTS

Observando la **Figura 0-10** del Anexo 3 se puede decir que hasta el momento se ha registrado un evento critico sobre el estado del dispositivo que nos dice que el dispositivo está apagado y tres eventos de peligro que corresponden a los servicios y procesos monitoreados especificando que tienen algún error o no están corriendo en el dispositivo.

PERF

De la **Figura 0-11** del Anexo 3, se recoge los siguientes datos:

CPU Utilization (utilización del CPU)

- Porcentaje de utilización de la CPU es de un 90%.

5.5.2.3 SWITCH

Para este caso se dispone de un switch que ya se ha agregado en Zenoss con sus respectivos plugins para su correspondiente monitoreo.

OS

Pestaña OS: como es un swicht tiene una serie de puertos; En la **Figura 0-12** del Anexo 3 se visualiza cada una de las interfaces: Ethernet, null y VLAN[53] que este posee con su IP, MAC y estado de actividad/inactividad correspondiente para cada una.

En Figura 0-12 del Anexo 3, se puede observar la sección de OS Processes e IP Services, en donde se puede visualizar y monitorear servicios y procesos requeridos:

[53] **VLAN** (Red de Área Local Virtual'), es un método de crear redes lógicamente independientes dentro de una misma red física.

OS processes: clase a la que pertenece, nombre del proceso, la severidad el estado de actividad/inactividad y el estado de bloqueo/desbloqueo.

IP Services: nombre del servicio, protocolo, puerto, descripción del servicio, estado de actividad/inactividad y el estado de bloqueo/desbloqueo.

Se puede monitorear el tráfico de sus puertos activos como el FastEhternet, VLAN, etc. En este caso se va a monitorear el tráfico de la VLAN 99, dando un clic sobre ella presenta la **Figura 0-12** del Anexo 3 de la cual se recoge los siguientes datos:

- Se tiene un rendimiento de entrada aproximadamente de manera constate de 650 hasta un límite de 870 Kbits/sec.
- Se tiene un rendimiento de salida aproximadamente de manera constate de 222 hasta un límite de 627 Kbits/sec.
- Ha emitido un promedio de 180 paktes /sec llegando hasta 420 paktes /sec.
- Ha recibido un promedio de 1.11 m paktes /sec llegando hasta 1.56 paktes /sec.
- Tiene 0 errores.

Load Average

- En un minuto tenemos 0.26 de carga de procesos.
- En 5 minutos tenemos 0.13 de carga de procesos.
- En 15 minutos tenemos 0.04 de carga de procesos.

CPU Utilization

- Porcentaje de espera de 0.0%.
- Porcentaje de usuario de 0.0%.
- Porcentaje del sistema en general 0.1%.
- Porcentaje sin utilizar 99.9%.

Memory Utilization

- Porcentaje de memoria usada 42.8 %
- Porcentaje en el Buffer 7.4 %.
- Porcentaje de memoria cache 27.7 %.
- Porcentaje de memoria swap 0.0%.

5.5.2.4 ROUTER

Posterior a la agregación del dispositivo, en este caso el Router en la clase específica (device/network/router/cisco), su locación y luego de haberse cerciorado que el template sea el correcto de acuerdo a los datos que deseemos obtener, modelamos el dispositivo para la recolección de estos datos dando como resultado las características del dispositivo agrupadas de la siguiente manera:

STATUS

La pestaña status (ver Anexo 3, **Figura 0-14**) presenta un conjunto de las principales características recolectadas del dispositivo agregado distribuidas de la siguiente manera:
La sección de device status nos permite conocer características como:

- El nombre del dispositivo, la dirección IP 172.16.40.10, estado; UP (el dispositivo esta encendido).
- El tiempo que lleva el dispositivo encendido: 10 d: 04 h: 08 m: 31 s.
- La disponibilidad; 100%, la prioridad; normal, el estado del dispositivo; producción, entre otras características.
- Muestra el estado de algunos componentes como: FastEthernet, VLANs, entre otros componentes.

La sección de device information se encuentra divida por organizadores y sistema operativo(OS).

- Organizadores: muestra las cuatro clases organizadoras que posee Zenoss con su respectiva información del dispositivo a monitorear, aquí se obtiene información a cerca de a qué grupo locación sistema y colector pertenece dicho dispositivo.
- En la parte de OS muestra información de el sistema operativo instalado; IOS[54] 13.4 (3i), el tipo de hardware; Cisco, el modelo de hardware; 1841, nombre del dispositivo R-CR-QUITO, información de contacto, etc.

La sección de descripción muestra:

- Un resumen de las características recolectadas más importantes del dispositivo.
- Link: un enlace a la consola del dispositivo de donde se puede realizar una mejor gestión del dispositivo.

[54] **IOS** Internetwork Operating System, Sistema operativo creado por Cisco Systems para programar y mantener equipos de interconexión de redes.

OS

OS (Interfaces): en el Anexo 3, **Figura 0-15** se visualiza las interfaces disponibles del dispositivo, en este caso el dispositivo cuenta con ocho interfaces todas en estado activo con sus respectivas direcciones IP, red a la que pertenecen esas interfaces y la dirección MAC.

También se puede ver el tráfico de dichas interfaces (ver Anexo 3, **Figura 0-16**) haciendo clic sobre una de ellas y podremos apreciar:

- En la parte de status se puede dar cuenta que la interfaz está en un estado operacional UP, estado de administración UP, la velocidad de la interfaz es de 100.000Mbps y que está siendo monitoreada (monitor – true).
- En la parte de gráficas de rendimiento se puede observar que existe un tráfico saliente de más de 1.41MB por segundo y un tráfico entrante de 761.68 K bits por segundo. De 66.69 a 305.69 paquetes por segundo salientes, de 58 a 210 paquetes por segundo entrantes y un porcentaje de error en el tráfico de 0%.
- Mantiene un estado de operación de 1.

HARDWARE

En la Pestaña hardware encontramos la capacidad de la memoria volátil y la swap en GB.

Según la **Figura 0-17** del Anexo 3, el dispositivo tiene las siguientes características en lo que se refiere a hardware:

- El dispositivo posee 73.8 MB de memoria aleatoria y memoria SWAP desconocida.

EVENTS

Observando la **Figura 0-18** del Anexo 3 se puede decir que hasta el momento se ha registrado solo eventos de DEBUG[55] el cual se identifica por el color plomo.

PERF

Esta pestaña lanza resultados gráficos de todo lo monitoreado en el router sobre el porcentaje evaluado cada 5 minutos de CPU y la cantidad de MB de la memoria disponible que le queda.

De la **Figura 0-19** del Anexo 3, se recoge los siguientes datos:

- Porcentaje de utilización de la CPU de 90%.
- Cada 5 min un porcentaje de 9.33% máximo.
- Porcentaje de memoria libre de 64.22 MB.

5.6 PROCESOS Y PROCEDIMIENTOS DE LA GESTIÓN DE RED

El manual de procesos de gestión de red del NOC-UTPL se lo realizó amparándose en el modelo presentado por la ISO por ser una de las principales opciones que se están utilizando a escala mundial, puesto que garantizan una gestión de calidad y desde el punto de vista económico reduce costes, tiempo y trabajo. Este modelo abarca: gestión del rendimiento, gestión de fallas, gestión de configuración, gestión de seguridad y gestión de contabilización quedando como entregable del presente proyecto, un manual de procesos para la gestión de la red para el NOC-UTPL.

[55] **Debug** Aplicación o herramienta que permite la ejecución controlada de un programa o un código, para seguir cada instrucción ejecutada y localizar así bugs o errores (proceso de depuración), códigos de protección, etc.

Cabe mencionar y dejar constando que del presente proyecto se presenta además del manual de procesos del NOC-UTPL, el manual de administrador y el manual de usuario correspondiente a la herramienta implementada Zenoss, cuya acta de entrega se encuentra en el anexo # 5

DISCUSIÓN

En la presente tesis se investigaron aspectos teóricos de la gestión de red, se estudiaron algunos NOC actuales, se hizo un estudio del estado actual del NOC-UTPL, también se consultó herramientas para la gestión de red llegando a realizar un análisis y selección de las mismas, teniendo como resultado la elección de Zenoss como herramienta de monitoreo para el NOC-UTPL por cumplir con los requerimientos (Anexo 1) establecidos por el NOC, y criterios de evaluación de un NMS (Tabla 3.1), finalmente se procedió a la implementación de Zenoss en un ambiente de pruebas en donde se la instaló de dos formas desde código fuente y mediante instalación binaria; posteriormente se procedió a la implementación de Zenoss en un entorno de producción, el cual durante su implementación se actualizó la versión por dos ocasiones, para permitir agregar nuevas funcionalidades a la herramienta.

Con la ayuda del NMS implementado se ha logrado atender de una mejor manera los requerimientos de monitoreo, optimizando las tareas de gestión de red y permitiendo de esta manera llevar un mejor control de la misma. Esta optimización

ha permitido disminuir en un 40 % del tiempo dedicado para las tareas del NOC (18.7 % de horas mensuales) que implica: revisión del estado de los distintos enlaces, revisión del estado de los recursos, revisión de los dispositivos de red, atención de requerimientos de monitoreo, recuperación de fallas y generación de reportes manualmente.

Con la realización de la presente tesis se puede expresar lo siguiente: para contar con un NOC estandarizado y sólido en la UTPL, es factible llevar una gestión de red haciendo uso en forma paralela tanto del manual de procesos como del NMS, aportando para ello al NOC.UTPL de un manual de procesos de gestión de red, la implementación de un Sistema de gestión de red (ZENOSS) en el servidor del NOC-UTPL, un manual de administrador y un manual de usuario correspondiente a la herramienta implementada.

Durante el desarrollo del presente proyecto se presentaron una serie de inconvenientes como: el servidor de pruebas asignado poseía características limitadas comparadas con las características que Zenoss especificaba para su implementación, constantes cambios de versiones; en el transcurso del proyecto luego de tener instalado la herramienta y en su mayor parte configurado el ambiente de pruebas, se lanza una versión actual de la herramienta por lo cual se utilizó la herramienta con la última versión después de haber realizado las evaluaciones correspondientes entre versiones. Además durante la configuración de la herramienta se presentaron errores en la aplicación los cuales fueron resueltos y debidamente documentados (Anexo 2).

TRABAJOS FUTUROS

Implementación de un servidor SNPP[56] (Simple Network Paging Protocol) para permitir el envío de mensajes de alerta mediante el uso de un módem y evitar así que el servidor no envié mensajes de alerta cuando no se disponga de internet por diferentes circunstancias inesperadas.

Construcción de Zenpacks específicos para la creación de gráficas que permitan visualizar procesos y así obtener gráficas y datos específicos para su monitoreo.

[56] **SNPP:** es un protocolo que define un método por el cual un localizador puede recibir un mensaje en el Internet.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

Con el estudio realizado del estado actual de los NOC, se definió: la estructura de trabajo, roles y responsabilidades, tareas a realizar para la conformación de un NOC, y las áreas funcionales de la gestión de red, otorgando una visión más clara de cada una de las actividades que se debe realizar para cada proceso de la gestión de red.

La gestión de red se debe llevar a cabo mediante la creación u conformación de un centro de gestión de redes (NOC), el mismo que debe constar de tres recursos principales: procesos de gestión, recursos humanos y herramientas de apoyo.

El NOC-UTPL realiza una gestión centralizada, cuenta con el proceso de gestión de fallas aunque no está estandarizado, posee herramientas de monitoreo, tales como: Nagios, Cacti, IPplan, Netflow Analizer. A pesar de no estar formalizado, el NOC-UTPL actualmente monitorea 173 dispositivos aproximadamente y cuenta con un servidor dedicado a la gestión de red.

En la actualidad existen una infinidad de herramientas de código abierto dedicadas a la gestión de redes, en base a los requisitos especificados en el presente trabajo, se destacan las siguientes herramientas: Zenoss, Cacti, Nagios, JFFNMS, Ossim, Hyperic.

La elección de la herramienta se realizo en base a los criterios de evaluación para el NMS (mecanismo de notificación, generación de reportes, extensión de funcionalidades, licencia, grafica de datos de rendimiento, integración de herramientas, etc.), quedando Zenoss como la mejor, destacándose por ser la aplicación que cumple con todos los criterios y además ser extensible en sus funcionalidades gracias a los Zenpacks.

El sistema de gestión de red (NMS), permite al administrador gestionar la red de una manera rápida fácil y sencilla, minimizando recursos, tiempo y costos.

Un NMS ayuda al operador de red a recolectar datos estadísticos de los dispositivos, detectar y diagnosticar los problemas de red logrando mantener la disponibilidad de la red el mayor tiempo posible.

Con la implementación de Zenoss como herramienta de gestión de red en el NOC-UTPL se logró visualizar todas las subredes conjuntamente con sus dispositivos, se monitoreó algunos de los dispositivos más relevantes de la red obteniendo gráficas de datos de dichos dispositivos, entre otras características.

Zenoss brinda un entorno gráfico agradable e intuitivo pudiéndose considerar como el mejor entorno gráfico en el ambiente Open Source.

Los reportes son una de las funcionalidades más fuertes de Zenoss debido a que cuenta con una extensa variedad en reportes.

Zenoss está integrado con otras herramientas como Cacti y Nagios (ambas open Source) que permite a los administradores tener un control completo sobre la infraestructura de red.

Para tener un mejor control de la red es factible realizar la gestión de red haciendo uso en forma paralela tanto del manual de procesos como del NMS.

Actualmente, con la implementación de Zenoss como herramienta de monitoreo se logró reducir en un 40% del tiempo dedicado a las tareas correspondientes a la gestión de red.

RECOMENDACIONES

Documentar toda la información relevante del NOC-UTPL. Debido a que no cuenta con una documentación detallada y apreciable para la gestión de la red.

Establecer formalmente el equipo NOC –UTPL.

Gestionar la red rigiéndose siempre a los procesos de la gestión establecidos en el manual de procesos y cumpliendo con las normas y políticas establecidas.

Configurar y comprobar el adecuado funcionamiento del agente SNMP en los dispositivos gestionados, verificando permisos y firewalls[57] para permitir la recolección de información mediante SNMP.

De las diversas formas de instalación de Zenoss se recomienda la instalación desde repositorios por ser la más fácil, rápida y menos propensa a fallos.

El servidor destinado para el funcionamiento de Zenoss debe de contar con las características esenciales de acuerdo a la cantidad de dispositivo que se requiere monitorear y basarse en a las especificaciones de hardware otorgadas por Zenoss.

Incrementar la capacidad de memoria RAM y disco del servidor actual del NOC-UTPL, considerando que en él se encuentra actualmente instaladas otras herramientas de monitoreo.

Se deberá crear alertas en los dispositivos más importantes y con un nivel de severidad alto, para evitar la saturación del mail.

[57] **Firewall**s es un elemento de software o hardware utilizado en una red para prevenir algunos tipos de comunicaciones prohibidos.

BIBLIOGRAFÍA

- ADVENTNET. (2008, abril): *User Guide Manage Engines OpUtils 5*.(en línea). (citado 14 de octubre 2010). Disponible en http://www.manageengine.com/products/oputils/AdventNet_ME_OpUtils.pdf

- ADVENTNET INC 5200, F. (2005.): *Free Network Monitoring Software for small Network.* .(en línea). USA: California(citado 09 de junio 2010). Disponible en http://zma.com.ar/contenidos/images/image/OpManager/Documentos/Free-Network-Monitoring-Tool.pdf

- ARBOLEDA MARTÍNEZ, A., & LÓPEZ TORRES, A. (2006.): *Sistema de monitoreo zenoss en Ubuntu 8.04.* (en línea). (citado 10 de octubre 2010). Disponible en http://www.scribd.com/doc/8872031/Sistema-de-Monitoreo-Zenoss-en-Ubuntu-8

- AYUQUEO, S. (2009): *Monitoreo y analisis de red con Nagios.* .(en línea). Argentina (citado 19 de febrero 2010). Disponible en http://cayu.com.ar/files/manual-nagios-2009.pdf.

- BOYER, Y. (2006, enero): *Aplicaciones Open Source para el monitoreo de redes IP.* .(en línea). Venezuela: Caracas(citado 4 de diciembre 2009). Disponible en http://neutron.ing.ucv.ve/comunicaciones/Asignaturas/DifusionMultimedia/Tareas%202006-1/Aplicaciones%20Open%20Source%20para%20el%20monitoreo%20de%20redes%20IP_Yubaira_.pdf

- CACTI. (2009). *CACTI.* .(en línea). (citado 5 de enero 2010). Disponible en http://www.cacti.net/features.php

- CHEMARY. (2006). *Introducción a la gestión de redes* (en línea).(citado 20 de enero 2010). Disponible en http://www.chemary.com/apuntes/gx.pdf

- CLARA. (2008): *Cooperacion Latinoamericana de redes avanzadas*. .(en línea). (citado 8 de enero 2010). Disponible en http://www.redclara.net/

- CONTRERAS, C., ARICAPA ARAQUE, D., RESTREPO VANEGAS, J., MUÑOS CASTAÑO, C. Y., RUA PULGARÍN, C. A., QUINTERO, L. T., y otros. (2008). *Administracion de redes y computadores.* Colombia: Medillin (citado 22 de julio 2010). Disponible en http://www.scribd.com/doc/8752314/Proyecto

- CORED. (2007). *Centro de operaciones de la red CORED*.(en línea). Mexico (citado 9 de enero 2010) Disponible en http://www.cored.df.gob.mx/

- CUBELLS, N. V. (2005): *Gestion de redes y servicios IP.* .(en línea). Mexico(citado 17 de junio 2009). Disponible en http://colaboracion.uat.edu.mx/ce/ehuerta/Shared%20Documents/Gesti%C3%B3n%20de%20redes%20y%20servicios%20IP.pdf

- DOMINGUEZ DORADO, M., & ZARANDIETA MORAN, J. (2003): *Evaluacion de Nagios para Linux.* .(en línea). España: Cáceres(citado 10 de febrero 2010). Disponible en http://nagios.sourceforge.net/download/contrib/documentation/misc/Nagios_spanish.pdf

- GARZON, J. (2006.): *Monitorizacion Grafica del Trafico de Red y otros parametros del Sistema.* .(en línea). (citado 5 de enero 2010). Disponible en http://www.redes-linux.com/manuales/Monitorizacion_redes/mrtg.pdf

- GONZALES, E. (2005, febrero): *Herramientas libres de monitorizacion: Nagios.* (en línea). España(citado 14 de enero 2010). Disponible en http://www.e-ghost.deusto.es/docs/2005/cursos/charlaNagios_20050224.pdf

- HERVEY, A. (2008, noviembre). *Estadísticas de red y servidores con CACTI* (en línea). Venezuela: Merida(citado 20 de enero 2010). Disponible en http://www.nsrc.org/workshops/2008/walc/presentaciones/Cacti.pdf

- *Introduccion a la gestion de redes.* (s.f.). Obtenido de http://www.chemary.com/apuntes/gx.pdf

- LIGUORI DE GOTTIG, A. M. (2009, febrero). *Security hack*. . (en línea). Argentina (citado 2 de enero 2010). Disponible en http://sechack.blogspot.com/2009/02/una-mirada-por-hyperic.html

- LINUX- MAGAZINE. (2009). "Inseguridades*". Magazine.* Rev. De Linux. Vol. 36 8-9

- LÓPEZ, D., VERGARA, J. E., BELLIDO, L., & FERNÁNDEZ, D. (2004). *Monitorización de una red académica mediante Netflow.* (en línea). España: Madrid(citado 22 de febrero 2010). Disponible en http://www.ii.uam.es/esp/investigacion/memoria_2004.pdf

- LUCA, D. (2008, mayo). *Open Source in Network Administration: The NTOP Project.* (en línea). (citado 08de marzo 2010). Disponible en http://www.ntop.org/OpenSourceConf_Athens2008.pdf

- LUDWING, R. C. (2004). *Redclara.* (en línea). Mexico: Veracruz(citado 14 de octubre 2010). Disponible en Operacione Red CLARA: http://www.redclara.net/noc/doc/OperacionRedCLARA.pdf

- MADRID MOLINA, J., MONTOYA GONZALES, C., OSORIO BETANCUR, J., VASQUEZ, A., CARDENAS, L. E., MUNERA SALAZAR, L., Y OTROS. (2008.). *Implementacion y Mejora de la Consola de Seguridad Informatica OSSIM en el Entorno Colombiano.* (en línea). Colombia: Barranquilla(citado 14 de octubre 2010). Disponible en http://memcontic1.googlepages.com/Conredes08_PhD_Juan_Manuel_Madrid_Im.pdf

- NANCE, B., & WORLD, N. (2007, agosto): *Zenoss Core.* (en línea). (citado 20 de enero 2010). Disponible en

http://www.google.com.ec/url?sa=t&source=web&cd=3&ved=0CCIQFjAC&url=http%3A%2F%2Fwww.boulderventures.com%2Fnews%2F061807_Rollout_Zenoss_Core_Review.doc&rct=j&q=Hyperic++filetype%3Adoc&ei=Jb4XTKCbJ4SKlwepg8HCCw&usg=AFQjCNH8PUq2VPvBTs64xFkIEn-CxTmr-A&sig2=K

- OPPENHEIMER, P., & PRESS, C. (2007). *Diseño de redes coorporativas: una metodologia decente*.Desarrollo de estrategias de gestión de red (en línea). (citado 12 de febrero 2010). Disponible en http://www.ldc.usb.ve/~emilio/CursosUSB/Redes3/TopDownDesign/Capitulo09.ppt

- OZ, M., & E&M, C. (2009, junio): *Cacti Complete Ntework graphing Solution.* (en línea). (citado 18 de marzo 2010). Disponible en http://www.emet.co.il/uploads/09_cacti.pdf

- PADILLA JAUME, L., ARIS, A., & FIBLA, X. (2007, junio): *Herramientas de Monitorizacion.* (en línea). España: Barcelona(citado 10 de abril 2010). Disponible en http://www.google.com.ec/url?q=http://c-new-car.googlecode.com/files/Herramientas%2520de%2520monitorizacion.pdf&ei=chJEStOFL6C_twey6tCyAg&sa=X&oi=spellmeleon_result

- SANZ TAPIA, R., SANCHEZ CID, L., & ALMORZA DAZA, J. (2006): *Monitorizacion y gestion de dispositivos, servicios y aplicaciones.* (en línea). España: Andalucia (citado 14 de octubre 2010). Disponible en Obtenido de http://www.csi.map.es/csi/tecnimap/tecnimap_2006/01T_PDF/monitorizacion.pdf

- T.MAGEDANZ, T. S. (1996): *From Networks and Network Management into Service and Service Management* (Vol. 4).

- TOARES, ROBERT. (2010): *Componentes de redes IP* .(en línea). Argentina. (citado 23 de febrero 2010). Disponible en http://www.robertoares.com.ar/wp-content/uploads/2010/06/Seccion-3.pdf

- UNAM. (1996): *Centro de Monitoreo de la red UNAM*. (en línea). Mexico(citado 14 de octubre 2010). Disponible en http://www.noc.unam.mx/

- UTPL. (2009): *Network Operation Center.*Universidad Tecnica Particular de Loja.

- VEGA TIRADO, R. E., HENAO ALVAREZ, J. A., LOAIZA GARCIA, J. A., PINEDA GONZALES, C. A., & MARTINEZ ALZATE, L. M. (2008): *Monitoreo y Gestion de Red*. (en línea). Colombia:Medillin (citado 19 de abril 2010). Disponible en http://www.scribd.com/doc/8422802/Proyecto-Monitoreo-Y-Gestion-de-Red-Con-Correcciones-a-los-comentarios-del-profe

- VELOSO, BRYAN. (2006): *Computación para ingenieros* .(en línea). (citado 19 de febrero 2010). Disponible en http://compuingenieros.wordpress.com/

- ZENOSS. (2010). *Zenoss Core Installation.* (en linea). USA: Annapolis (citado 25 marzo 2010). Disponible en http://www.scribd.com/doc/37271836/Zenoss-Core-Installation-04-072010-3-0-v02

ANEXOS

Anexo 1. Plantilla de Especificación de requerimientos

Especificación de requisitos de Software

Proyecto: Sistema de Gestión de redes NOC con herramientas de código abierto

Revisión 1.0

marzo de 2011

CONTROL DE CAMBIOS

Registro del control de cambios en el documento SRS del Sistema de Gestión de Redes NOC con herramientas de Código Libre

Fecha	Versión	Descripción	Autor
17-junio-2009	1.0	Versión inicial previa aprobación.	Veronica Ramírez

COMPILACIÓN Y AUTOMATIZACIÓN DE UN SISTEMA DE GESTIÓN DE REDES NOC CON HERRAMIENTAS DE CÓDIGO ABIERTO.

SRS-ESPECIFICACIÓN DE REQUERIMIENTOS

Código	Nombre	Fecha	Grado Necesidad
REQ-01	Monitoreo de servicios	17-julio-2009	alto
Descripción	El sistema deberá Monitorear los servicios (HTTP, POP3, SNMP, FTP).		

Código	Nombre	Fecha	Grado Necesidad
REQ-02	Monitoreo de recursos	17-julio-2009	medio
Descripción	El sistema deberá Monitorear los recursos de los dispositivos(disco, memoria, CPU)		

Código	Nombre	Fecha	Grado Necesidad
REQ-03	Monitoreo de conectividad	17-julio-2009	medio
Descripción	El sistema deberá Monitorear el estado del dispositivo (up, down)		

Código	Nombre	Fecha	Grado Necesidad
REQ-04	Alertas y Notificaciones	17-julio-2009	alto
Descripción	El sistema deberá permitir la Generación de alertas basado en notificaciones que nos informen el estado de los servicios y dispositivos a través de un medio de comunicación		

Código	Nombre	Fecha	Grado Necesidad
REQ-05	Descubrimiento de red	17-julio-2009	medio

Descripción	El sistema permitirá el Descubrimiento y visualización de la red junto con los dispositivos que la conforman		
Código	**Nombre**	**Fecha**	**Grado Necesidad**
REQ-06	Reportes	17-julio-2009	alto
Descripción	El sistema deberá permitir la generación de reportes		
Código	**Nombre**	**Fecha**	**Grado Necesidad**
REQ-07	Eventos	17-julio-2009	alto
Descripción	El sistema deberá permitir la gestión de eventos		
Código	**Nombre**	**Fecha**	**Grado Necesidad**
REQ-08	Herramienta integrada	17-julio-2009	alto
Descripción	El sistema deberá estar integrado con otras herramientas de gestión sobresalientes en este campo.		
Código	**Nombre**	**Fecha**	**Grado Necesidad**
REQ-09	Extensión de funcionalidades	17-julio-2009	medio
Descripción	La herramienta debe permitir extender sus funcionalidades, permitir la creación de paquetes para incrementar sus características.		

Anexo 2.-Errores

ERROR DE GOOGLE MAPS

Type: KeyError
Value: '172.16.1.8'

Traceback (innermost last):

* Module ZPublisher.Publish, line 114, in publish
* Module ZPublisher.mapply, line 88, in mapply
* Module ZPublisher.Publish, line 40, in call_object
* Module Shared.DC.Scripts.Bindings, line 311, in __call__
* Module Shared.DC.Scripts.Bindings, line 348, in _bindAndExec
* Module Products.CMFCore.FSPageTemplate, line 195, in _exec
* Module Products.CMFCore.FSPageTemplate, line 134, in pt_render
* Module Products.PageTemplates.PageTemplate, line 104, in pt_render
  <FSPageTemplate at /zport/locationGeoMap used for /zport/dmd/Locations>
* Module TAL.TALInterpreter, line 206, in __call__
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 711, in do_useMacro
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 426, in do_optTag_tal
* Module TAL.TALInterpreter, line 411, in do_optTag
* Module TAL.TALInterpreter, line 406, in no_tag
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 711, in do_useMacro
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 426, in do_optTag_tal
* Module TAL.TALInterpreter, line 411, in do_optTag
* Module TAL.TALInterpreter, line 406, in no_tag
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 734, in do_defineSlot
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 426, in do_optTag_tal
* Module TAL.TALInterpreter, line 411, in do_optTag
* Module TAL.TALInterpreter, line 406, in no_tag
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 734, in do_defineSlot
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 426, in do_optTag_tal
* Module TAL.TALInterpreter, line 411, in do_optTag
* Module TAL.TALInterpreter, line 406, in no_tag
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 677, in do_condition
* Module TAL.TALInterpreter, line 250, in interpret
* Module TAL.TALInterpreter, line 426, in do_optTag_tal

```
    * Module TAL.TALInterpreter, line 411, in do_optTag
    * Module TAL.TALInterpreter, line 406, in no_tag
    * Module TAL.TALInterpreter, line 250, in interpret
    * Module TAL.TALInterpreter, line 501, in do_insertText_tal
    * Module Products.PageTemplates.TALES, line 227, in evaluateText
    * Module Products.PageTemplates.TALES, line 221, in evaluate
      URL: file:ZenWidgets/skins/zenui/locationGeoMap.pt
      Line 17, Column 4
      Expression: string:"\n var geocodecache;\n var nodedata =
${here/getChildGeomapData};\n var linkdata = ${here/getChildLinks};\n var
cachestring = '${here/dmd/getGeoCache}';\n if (cachestring) geocodecache =
evalJSON(cachestring);\n "
      Names:

      {'container': <ZentinelPortal at /zport>,
       'context': <Location at /zport/dmd/Locations>,
       'default': <Products.PageTemplates.TALES.Default instance at 0xb74a3b6c>,
       'here': <Location at /zport/dmd/Locations>,
       'loop': <Products.PageTemplates.TALES.SafeMapping object at 0x9021d4c>,
       'modules': <Products.PageTemplates.ZRPythonExpr._SecureModuleImporter
instance at 0xb74aabcc>,
       'nothing': None,
       'options': {'args': ()},
       'repeat': <Products.PageTemplates.TALES.SafeMapping object at 0x9021d4c>,
       'request': <HTTPRequest,
URL=http://10.20.6.23:8080/zport/dmd/Locations/locationGeoMap>,
       'root': <Application at >,
       'template': <FSPageTemplate at /zport/locationGeoMap used for
/zport/dmd/Locations>,
       'traverse_subpath': [],
       'user': mawhi}

    * Module Products.PageTemplates.Expressions, line 224, in __call__
    * Module Products.PageTemplates.Expressions, line 185, in __call__
    * Module Products.PageTemplates.Expressions, line 180, in _eval
    * Module Products.PageTemplates.Expressions, line 85, in render
    * Module Products.ZenModel.Location, line 110, in getChildLinks
    * Module Products.ZenModel.LinkManager, line 171, in getChildLinks
    * Module OFS.Traversable, line 221, in unrestrictedTraverse
      __traceback_info__: ([], '12.87.196.92')
    * Module OFS.ObjectManager, line 713, in __getitem__

KeyError: '172.16.1.8'
```

SOLUCIÓN

http://forums.zenoss.com/viewtopic.php?p=22959

In this post i find all i need
http://community.zenoss.com/forums/viewtopic.php?p=22878#22878
[root@localhost zenoss]# su zenoss
[zenoss@localhost zenoss]$ zendmd
Welcome to zenoss dmd command shell!
use zhelp() to list commands
>>> dmd.ZenLinkManager.layer3_catalog.manage_catalogClear();
>>> commit();
>>>
[zenoss@localhost zenoss]$ zenmigrate --step=ReindexIpAddressNetworkIds
INFO:zen.migrate:Installing ReindexIpAddressNetworkIds (2.2.1)
INFO:zen.migrate:committing
INFO:zen.migrate:Migration successful

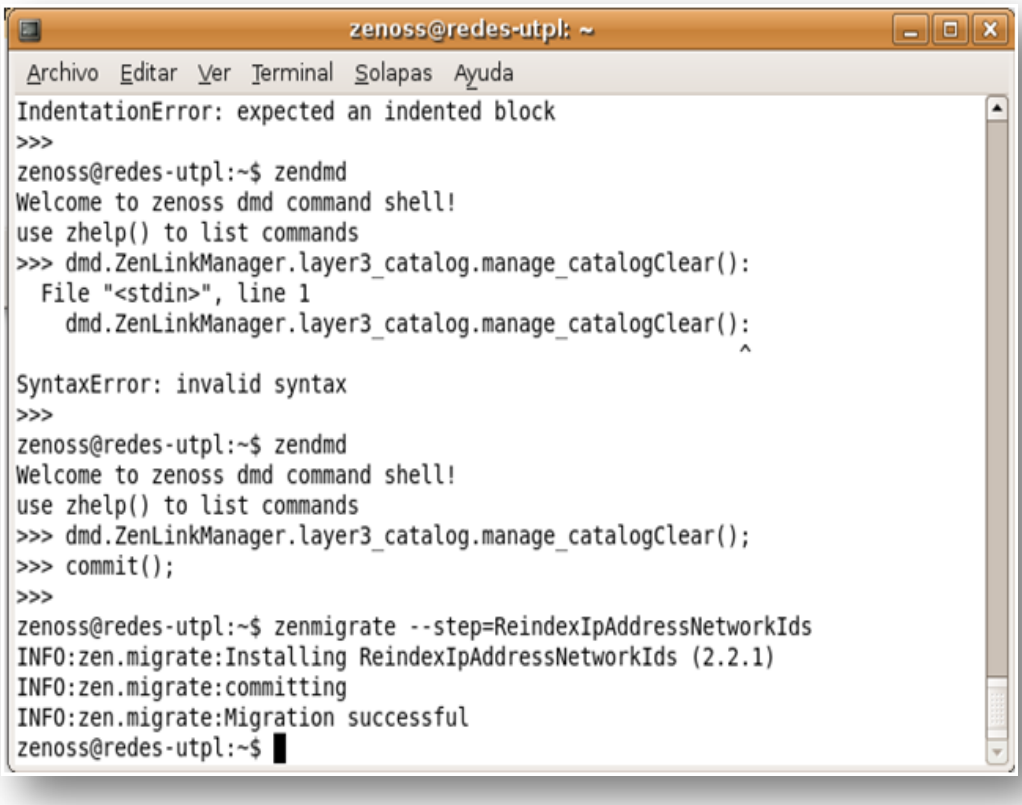

Figura 0-1 Solución del error de google Maps

PROBLEMA CON LA INTERFAZ WEB DE ZENOSS

La interfaz web de Zenoss no muestra absolutamente ninguna información, no realiza la conexión con el servidor, al momento de digitar la URL no visualiza nada se queda en blanco.

SOLUCIÓN

http://www.sysadminwiki.net/wiki/index.php?title=Common_Zenoss_Errors_-_Post_Install

Connection Refused / Zope fails to start

This happened to me after I upgraded my kernel, and rebooted the server. While collections and monitoring were still working finest kind, I could not connect to the web interface on port 8080. Luckily I found a thread where Chet Luther explains the fix. The upgrade apparently corrupted my zope cache. This fixed it:

1. Log in as the zenoss user.
2. Clear the zope cache.

```
zopectl stop
rm $ZENHOME/var/zeo1-1.zec
zeoctl start
```

INSTALACIÓN DE ZENPACKS VIA UI (versión 2.5.2)

En la versión actual de Zenoss (2.5.2) no se puede instalar zenpacks desde la interfaz web.

Actualmente no hay solución, la comunidad de Zenoss se encuentra resolviendo el problema pero hasta la fecha no se presenta solución pero dan una alternativa instalar el Zenpack desde consola.

ALTERNATIVA:

El ticket creado para este problema está en el siguiente enlace:

http://dev.zenoss.com/trac/ticket/6250

Instalación manual del Zenpack utilizando SSH:

1. Loguearse en el servidor como un usuario mediante ssh
2. Cambiarse de usuario a usuario Zenoss haciendo:
3. Moverse al lugar en donde descargamos los Zenpacks
4. Usamos el siguiente comando para instalar los Zenpacks
5. Log in on your system using the ssh client.

Para asegurarnos de la instalación de Zenpack, nos ubicamos en la consola web de Zenoss, nos ubicamos en Settings→Zenpacks y podemos visualizar el Zenpack instalado. Si vemos que el Zenpack es reportado como Broken o missing es porque necesita que el servidor se reinicie para tomar los cambios efectuados. Para esto nos conectamos con el servidor vía ssh y como usuario root digitamos lo siguiente:

/etc/init.d/zenoss-stack stop

Anexo 3.-Pantallas del monitoreo de dispositivos

LINUX

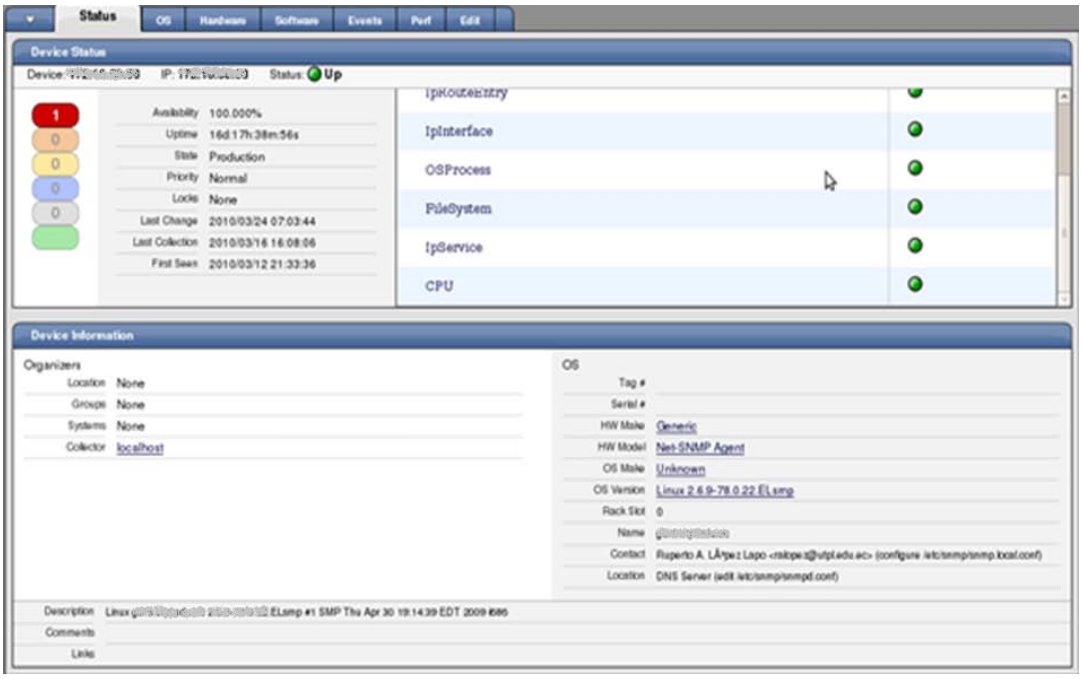

Figura 0-2 Pestaña status (Linux)

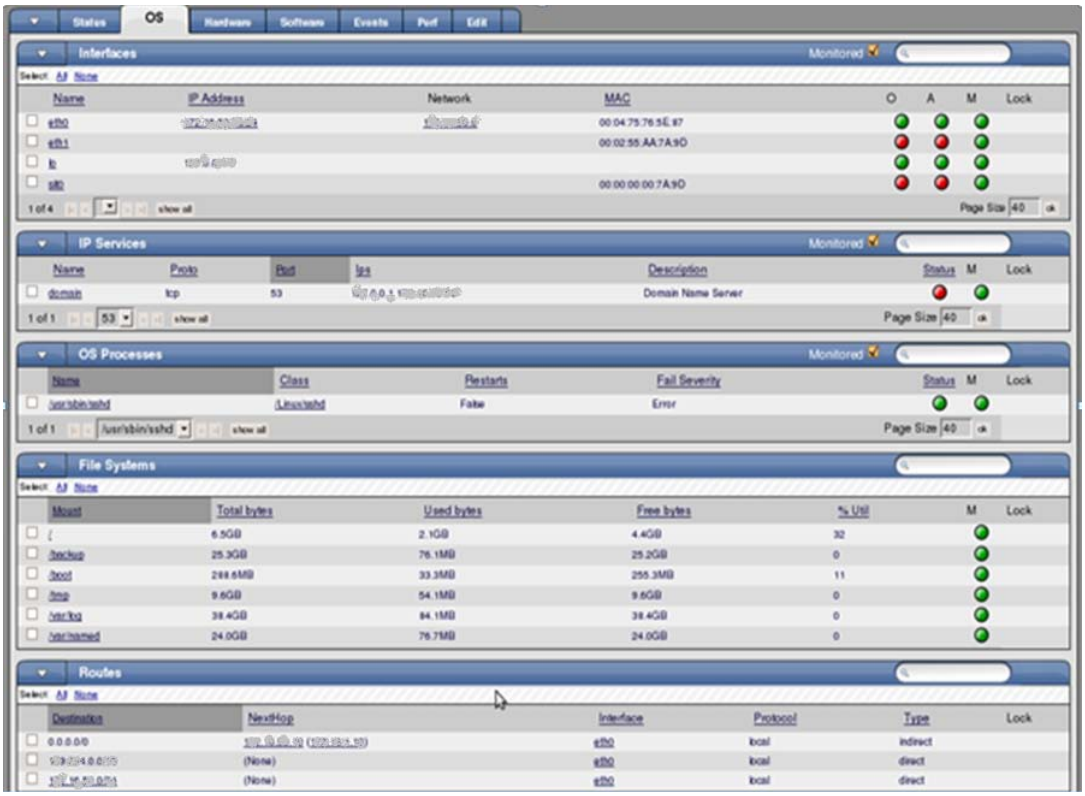

Figura 0-3 Pestaña OS (Linux)

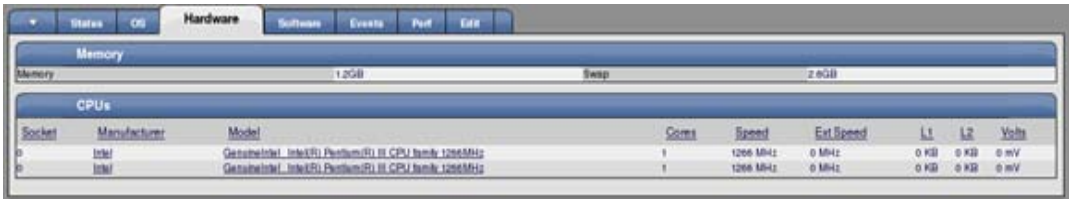

Figura 0-4 Pestaña hardware (Linux)

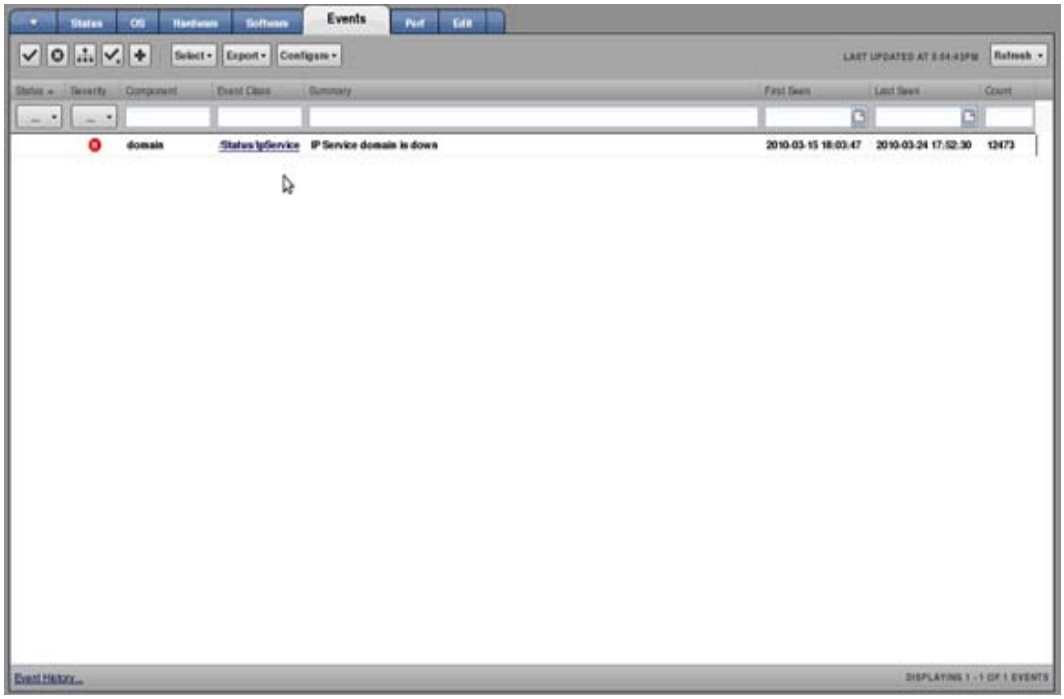

Figura 0-5 Pestaña de events (Linux)

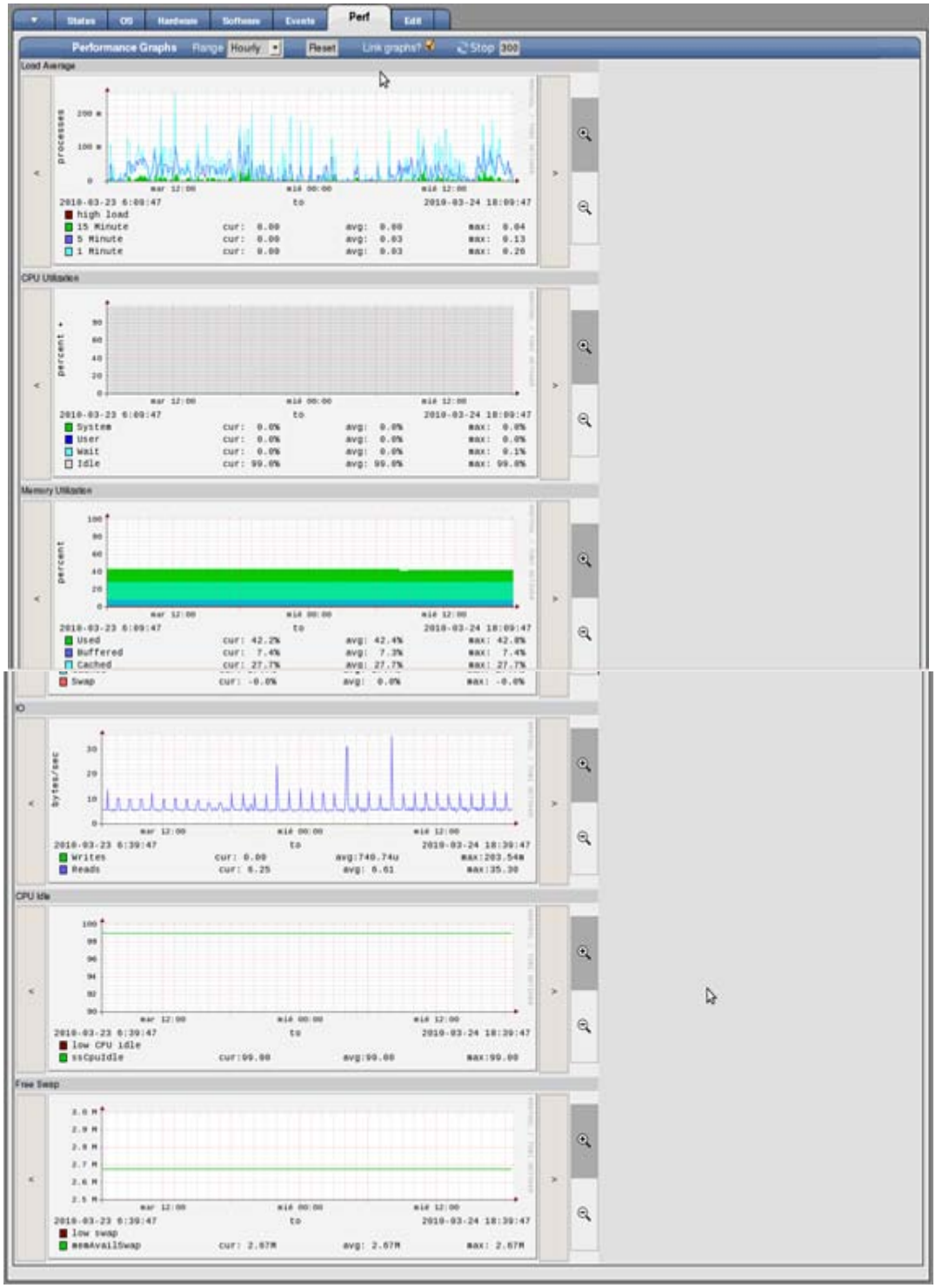

Figura 0-6 Pestaña Perf (Linux)

WINDOWS

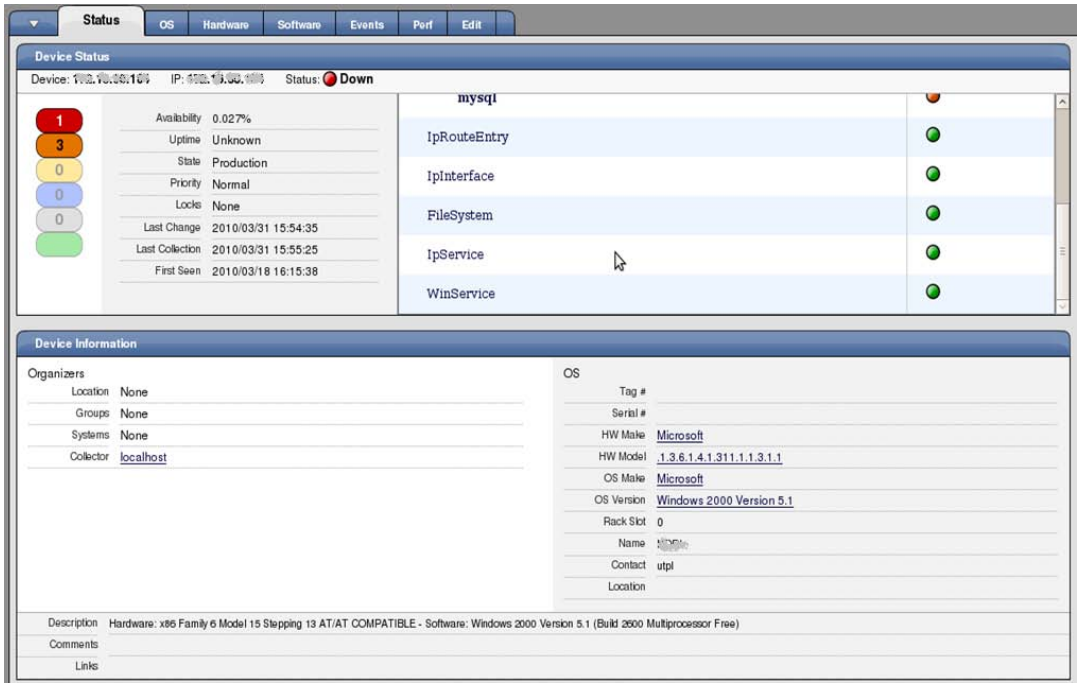

Figura 0-7 Pestaña Status (Windows)

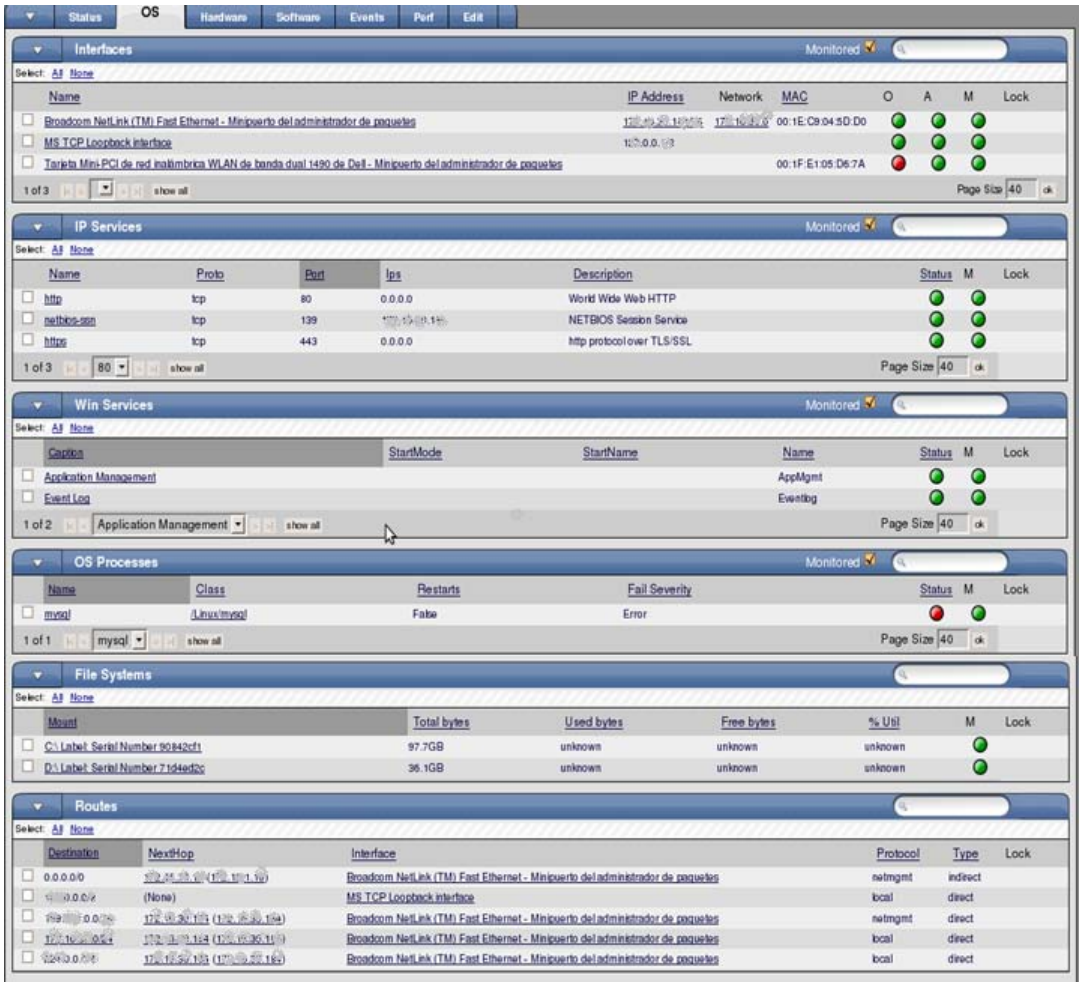

Figura 0-8 Pestaña OS (Windows)

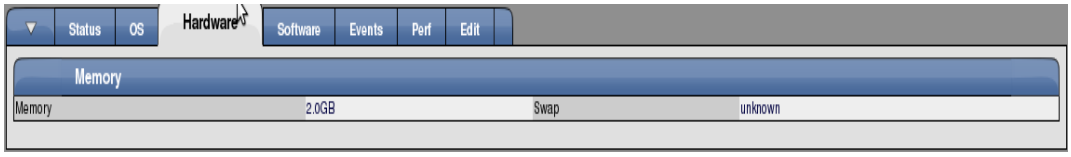

Figura 0-9 Pestaña Hardware (Windows)

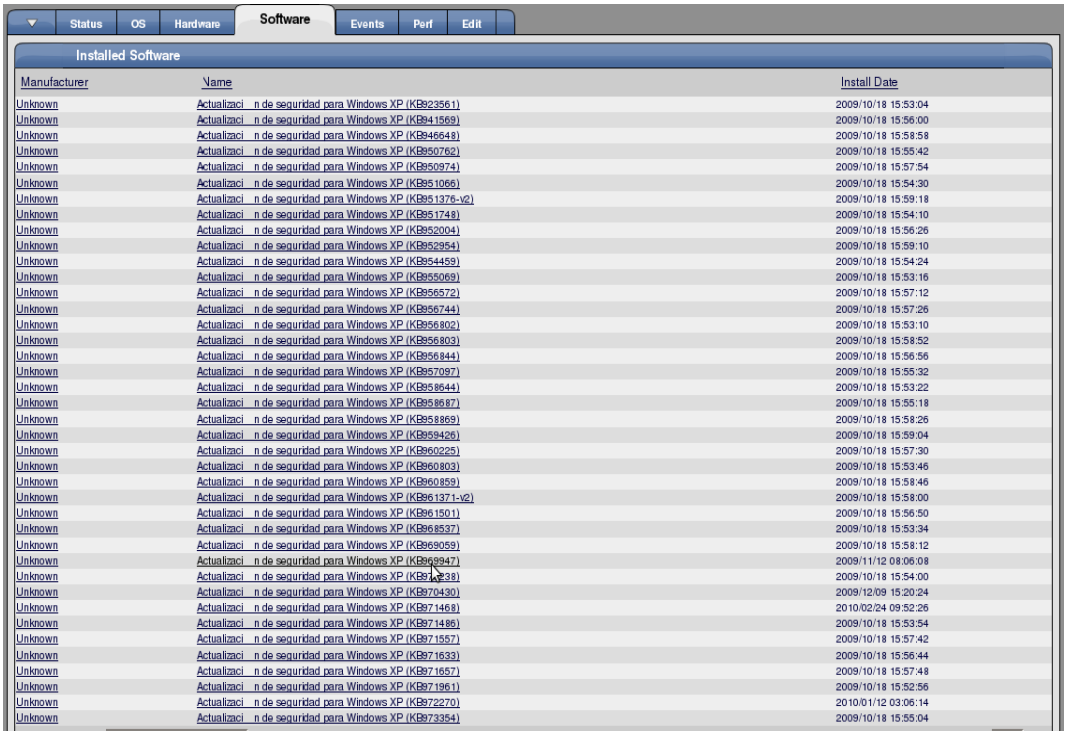

Figura 0-10 Pestaña Software (Windows)

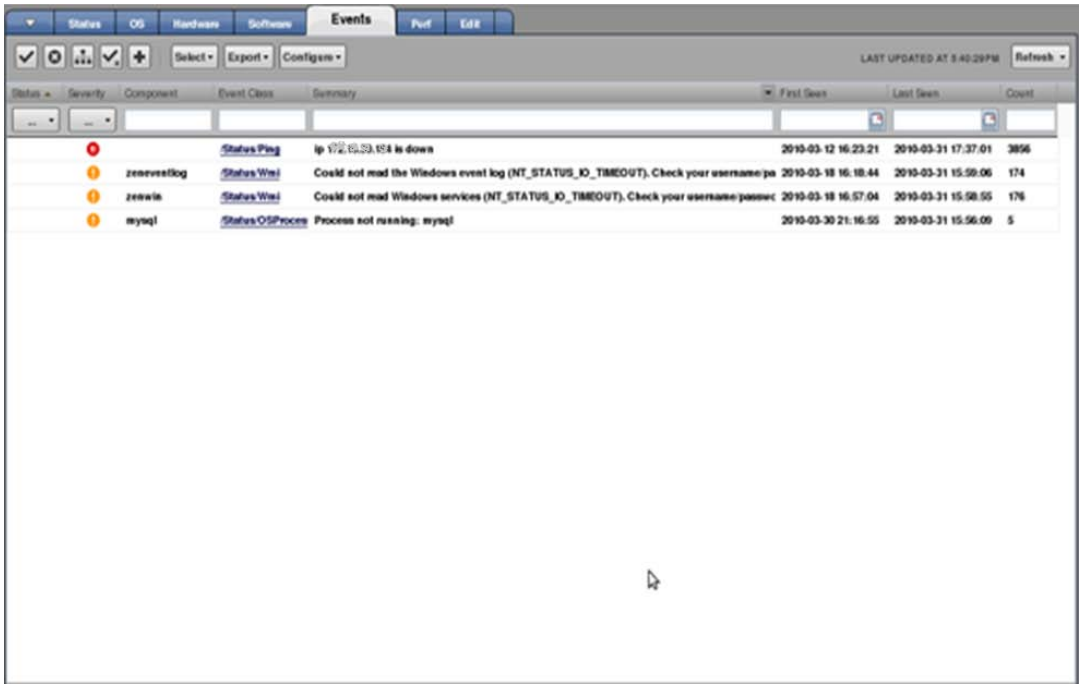

Figura 0-11 Pestaña Events (Windows)

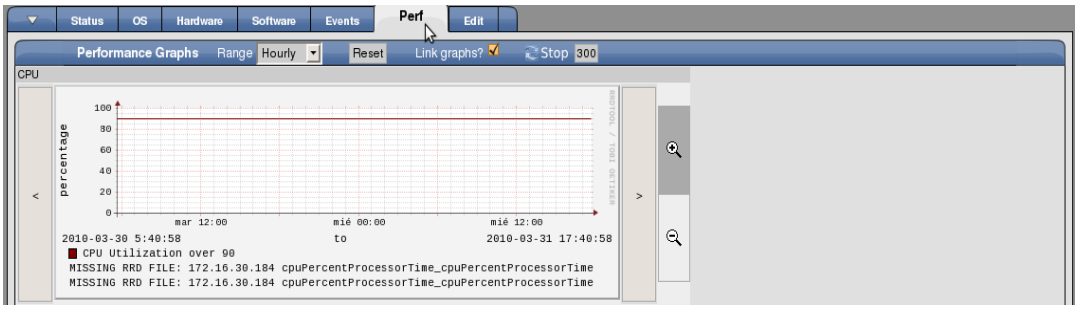

Figura 0-12 Pestaña Perf (Windows)

SWICTH

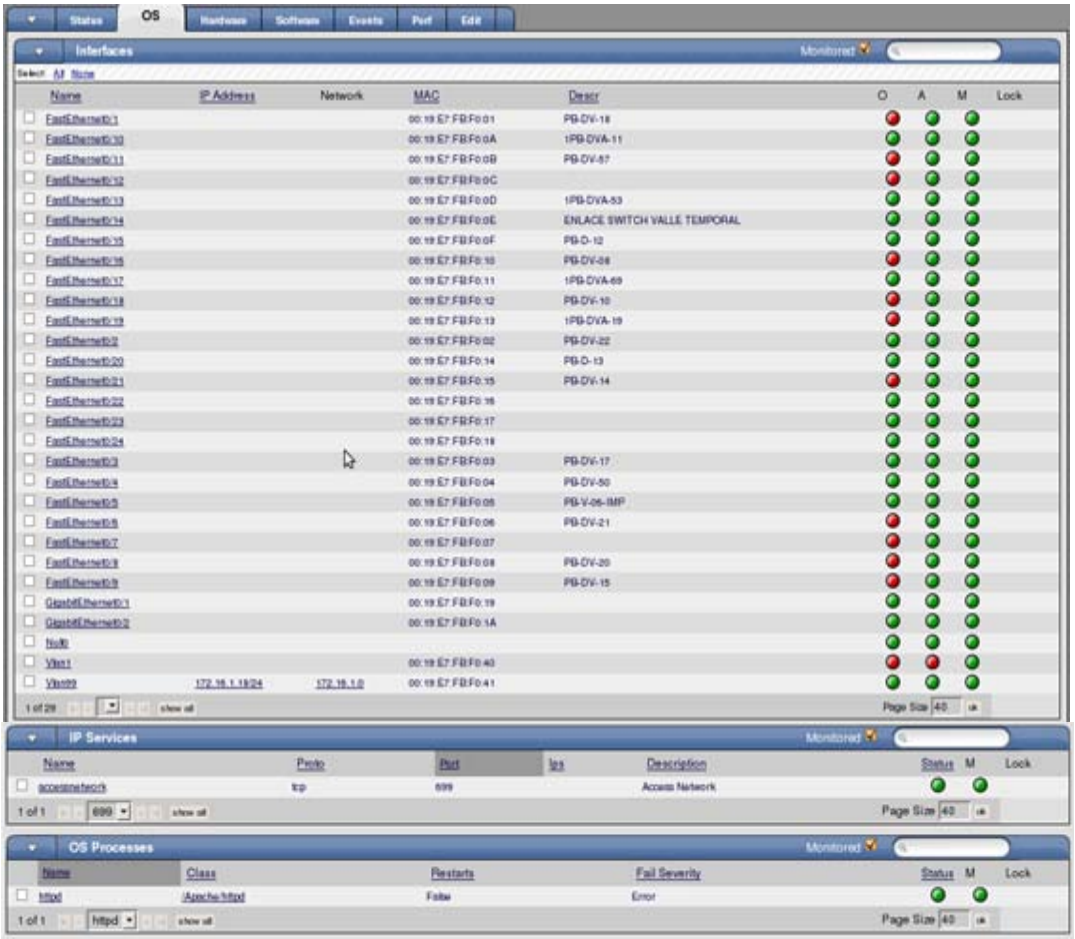

Figura 0-13 Pestaña OS (Switch)

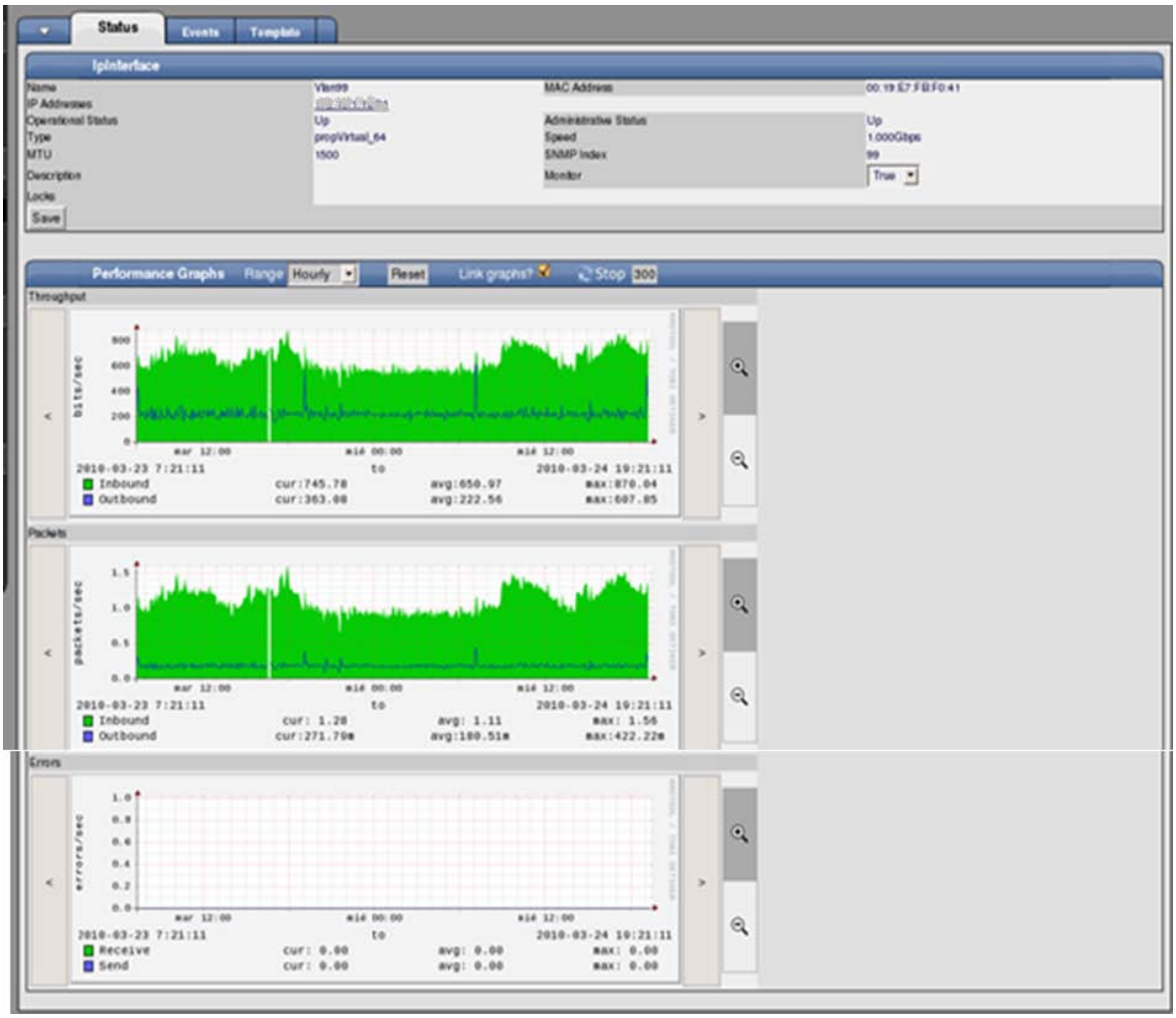

Figura 0-14 Gráficas de la VLAN (Switch)

ROUTER

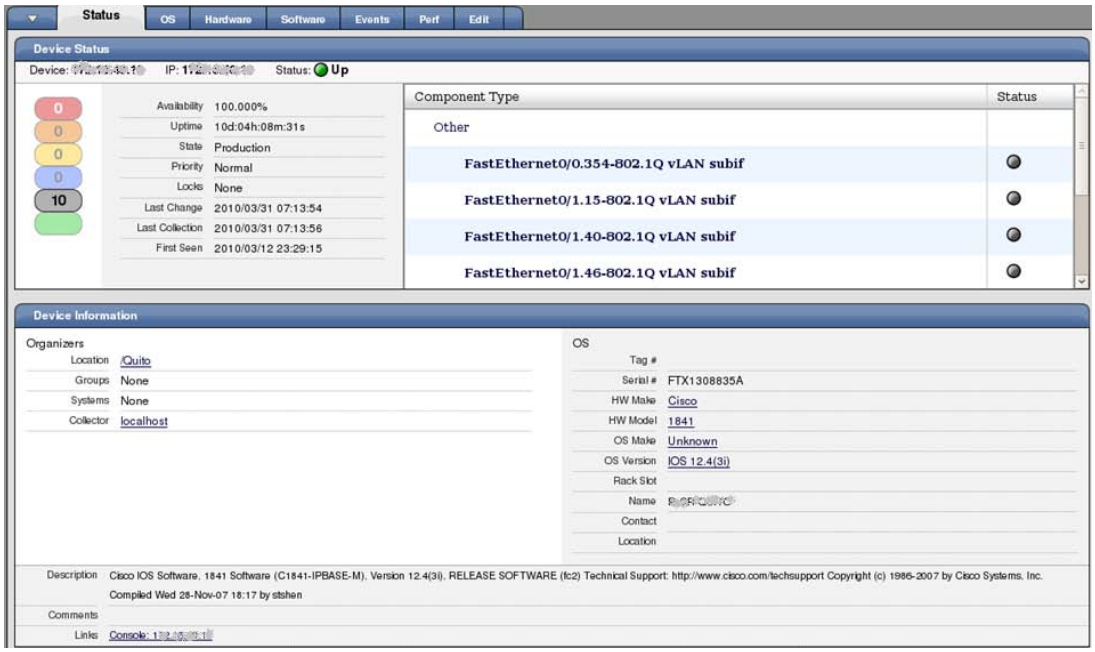

Figura 0-15 Pestaña Status (Router)

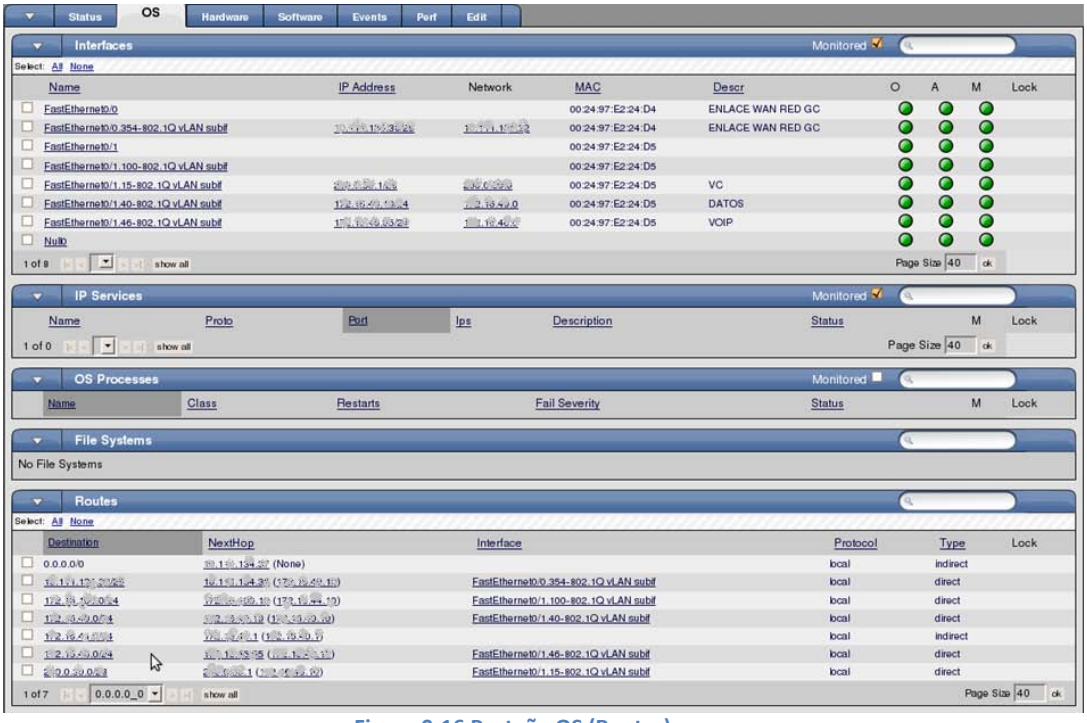

Figura 0-16 Pestaña OS (Router)

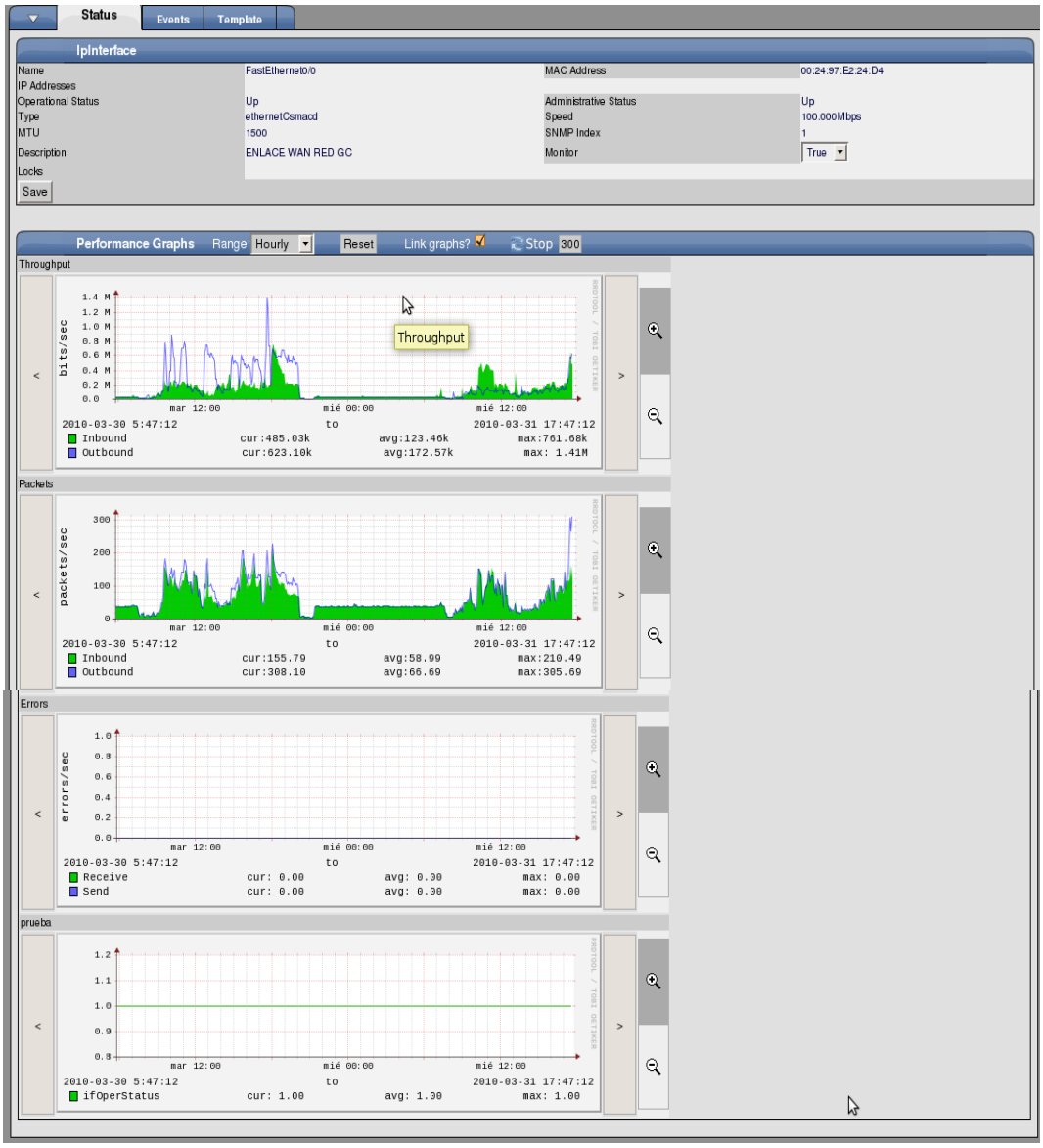

Figura 0-17 Gráficas de la interfaz del Router

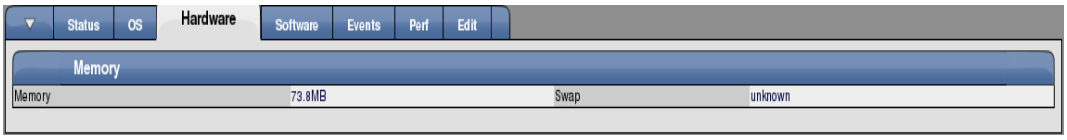

Figura 0-18 Pestaña Hardware (Router)

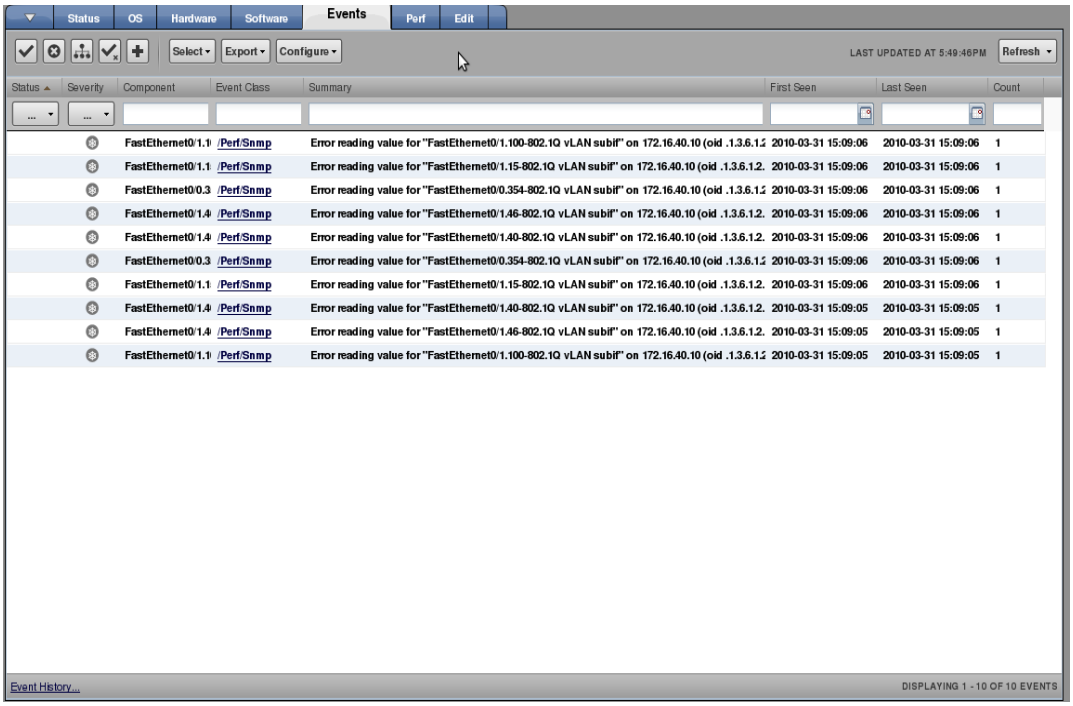

Figura 0-19 Pestaña Events (Router)

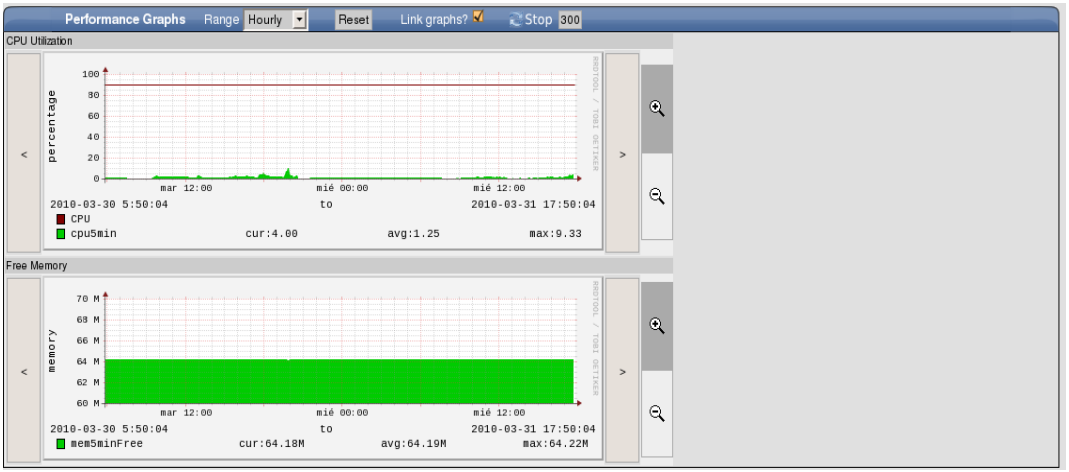

Figura 0-20 Pestaña Perf (Router)

Anexo 4.-Demonios de Zenoss

DEMONIOS	CARACTERISTICAS
• ZenrRRD	• Reúne series cronológicas de datos y actúa como un RRDtool.
• Zenevents	• Interactúa con la base de datos MySQL Eventos.
• Zenmodel	• Configuración del modelo de Zope (objeto de base de datos)
• Zenhub	• Broker de información entre la capa de datos y la recogida de los demonios.
Automatizado de Modelado	
• Zendisc	• Encargado de descubrir todas las redes activas, para encontrar direcciones ip y dispositivos.
• ZenwinModeler	• Se utiliza para el auto-descubrimiento de Windows Servicios (WMI) se ejecuta en un cuadro de las ventanas.
• ZenModeler	• Se utiliza para alto rendimiento, modelo que utiliza SNMP, SSH, Telnet
Disponibilidad de modelos	
• Zenping	• Supervisión del estado del ping para ICMP
• Zenstatus	• Realiza pruebas de conexión TCP remoto de los demonios
• Zenprocess	• Permite la supervisión de proceso utilizando los recursos de acogida

	SNMP MIB.
Evento de colección	
• Zensyslog	• Es la recogida y clasificación de syslog de eventos
• Zeneventlog	• Se utiliza recoger (WMI) de registro de eventos de eventos.
• Zentrap	• Recoge trampas SNMP. Recibe las trampas y los convierte en los acontecimientos.
Monitoreo de desempeño	
• ZenperfSNMP	• Realiza un alto rendimiento asincrónico SNMP.(Rendimiento de colección)
• ZenperfxMLrpc	• Se utiliza para XML RPC colección
• Zencommand	• Utiliza para XML RPC, permite el funcionamiento de Nagios y Cacti y plug-ins local o remotamente a través de ssh
Respuesta automática	
• Zenactions (Arboleda Martinez, Lopez Torres, & Garcia, 2006)	• Se utiliza para alertas (SMTP, SNPP y mantenimiento Windows)

Tabla 0-1 Demonios de Zenoss (Vega Tirado, Henao Alvarez, & Loaiza Garcia, 2008)

Anexo 5.-Acta de Entrega

ACTA DE ENTREGA - RECEPCIÓN

En la ciudad de Loja a los dieciocho días del mes de Enero del año 2011 se reunieron en la UTPL. El Ing. Carlos Aguilar Administrador del NOC-UTPL y la Señorita Verónica Ramírez tesista del tema "COMPILACIÓN Y AUTOMATIZACIÓN DE UN SISTEMA DE GESTIÓN DE REDES NOC CON HERRAMIENTAS DE CÓDIGO ABIERTO "con la finalidad de proceder con la entrega de los siguientes documentos:

- Manual de procesos de gestión de red
- Manual de administrador de Zenoss
- Manual de usuario de Zenoss

_____	_____
ENTREGA	**RECIBE**
Verónica Leonor Ramírez	**Ing. Carlos Aguilar**

Anexo 6.-Manuales

En el presente proyecto se obtuvo los siguientes manuales:

- Manual de procesos de gestión de red (ver CD)
- Manual de administrador de ZENOSS (ver CD)
- Manual de usuario de ZENOSS (ver CD)

Anexo 7.-Paper

Del presente proyecto se obtiene un paper (ver CD), el cual destaca los aspectos más importantes de la investigación.

GLOSARIO DE TÉRMINOS

AJAX: Asynchronous JavaScript y XML, es una técnica de desarrollo web para crear aplicaciones web que se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

CGIs: (Common Gateway Interface). Tecnología de la WWW que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa.

CLI: Interfaz de línea de comandos que permite manipular un programa o sistema operativo con instrucciones escritas.

CMIP: (Común de Información de Gestión Protocolo), El protocolo de gestión de la información común (CMIP) es un protocolo OSI utilizado para la gestión de redes . Apoya el intercambio de información entre las aplicaciones de gestión de redes y agentes de administración.

CPU: Central Processing Unit (unidad de proceso central, es el cerebro del ordenador. A veces es referido simplemente como el procesador o procesador central, la CPU es donde se producen la mayoría de los cálculos. En términos de potencia del ordenador, la CPU es el elemento más importante de un sistema informático.

Debug: Aplicación o herramienta que permite la ejecución controlada de un programa o un código, para seguir cada instrucción ejecutada y localizar así bugs o errores (proceso de depuración), códigos de protección, etc.

DHCP: (Dynamíc Host Confíguration Protocol). Protocolo de configuración dinámica de host. Protocolo que usan las computadoras para obtener información de configuración. El DHCP permite asignar una dirección IP a una computadora sin requerir que un administrador configure la información sobre la computadora en la base de datos de un servidor.

DNS: (Domain Name System) Sistema de Nombres de Dominio. Conjunto de protocolos y servicios para la identificación/conversión de una dirección de internet expresada en lenguaje natural por una dirección IP.

Ethernet: Es un estándar de red para la transmisión de datos mediante cable de par trenzado o coaxial. In the LAN (local area network) world, Ethernet is the most widely used standard.

Firewalls: Un cortafuegos o firewall, es un elemento de software o hardware utilizado en una red para prevenir algunos tipos de comunicaciones prohibidos según las políticas de red que se hayan definido en función de las necesidades de la organización responsable de la red.

Front-end: De forma general, front-end hace referencia al estado inicial de un proceso. Contrasta con back-end, que se refiere al estado final de un proceso. Es responsable de recoger entradas de los usuarios, y ser procesadas de tal manera que cumplan las especificaciones para que el back-end pueda usarlas.

FTP: File Transfer Protocol, el protocolo para intercambiar archivos en Internet. Su misión es permitir a los usuarios recibir y enviar ficheros de todas las máquinas que sean servidores FTP.

GPL: General Public License (Licencia Pública General). Licencia creada por la Free Software Foundation y orientada principalmente a los términos de distribución, modificación y uso de software libre.

Helpdesk: Conjunto de recursos técnicos y humanos que permiten dar soporte a diferentes niveles de usuarios informáticos de una empresa, es un recurso de información y asistencia para resolver problemas con computadoras y productos similares, las corporaciones a menudo proveen soporte(helpdesk) a sus consumidores vía número telefónico totalmente gratuito, website o e-mail. También hay soporte interno que provee el mismo tipo de ayuda para empleados internos solamente.

HTML: Hyper Text Markup Language, es un lenguaje de programación muy sencillo que se utiliza para crear los textos y las páginas web. Si se hace la traducción de su nombre del inglés al castellano, sería “Lenguaje de Marca de Hipertextos”, ya que es justamente un lenguaje que se basa en las marcas para crear los hipertextos.

HTTP: HyperText Transfer Protocol (Protocolo de transferencia de hipertexto) es el método más común de intercambio de información en la world wide web, el método mediante el cual se transfieren las páginas web a un ordenador.

ICMP: (Internet Control Message Protocol - Protocolo de Control de Mensajes de Internet). Subprotocolo de diagnóstico y notificación de errores del Protocolo de Internet (IP). Es utilizado para enviar mensajes de errores cuando un servicio no está disponible o cuando un host no puede ser encontrado, etc.

Instalación Monolítica: Instalar software así ya esté instalado en la maquina.

IOS: Internetwork Operating System, (Sistema Operativo de Interconexión de Redes). Sistema operativo creado por Cisco Systems para programar y mantener equipos de interconexión de redes informáticas como switches (conmutadores) y routers (enrutadores).

ISO: (International Organization for Standardization) es la Organización Internacional para la Estandarización. Su nombre ISO significa "igual" en griego. Fue fundada en el año 1946 y unifica a más de cien países. Se encarga de crear estándares o normas internacionales.

LAN: (Red de área local), es una red informática que abarca un área de físico pequeño, como una casa, oficina, o pequeños grupos de edificios, como una escuela, o un aeropuerto.

Logs: Archivo que registra movimientos y actividades de un determinado programa (log file). En un servidor web, se encarga de guardar todos los requerimientos ("requests") y servicios entregados desde él, por lo que es la base del software de estadísticas de visitas.

MAC: (Media Access Control o control de acceso al medio) es una dirección que posee un identificador de 48 bits que corresponde de forma única a una Ethernet de red. Se conoce también como la dirección física en cuanto a identificar dispositivos de red. Cada dispositivo tiene su propia dirección MAC.

MIB: (Management Information Systems o base de información de gestión), es una base de datos virtual que se usa para la gestión de las entidades en una red de comunicaciones.

MySql: MySQL es un sistema de gestión de bases relacionales (RDBMS), basado en SQL (Structured Query Language).

NMS: Sistema de gestión de redes (Network Management System), es un software utilizados para supervisar y administrar una red.

NOC: Centro de operación de red (Network Operation Center) donde se analiza el funcionamiento y operación de todos los equipos que componen la red y el Centro de Datos.

Nodo: Punto de conexión, ya sea un punto de redistribución o un punto final de la comunicación.

NRPE: Nagios Remote Plug-in Executor, le permite ejecutar remotamente plugins de Nagios en otros equipos Linux / Unix. Esto le permite controlar las estadísticas remotas de la máquina (el uso de disco, CPU, etc.)

Open Source: Software libre y de código abierto (también conocido como FOSS o FLOSS, por free/libre and open Source software, en inglés) es el software que está licenciado de tal manera que los usuarios pueden estudiar, modificar y mejorar su diseño mediante la disponibilidad de su código fuente.

OS: Operating System (Sistema Operativo). Es un conjunto de programas que permiten la comunicación del usuario con una computadora. El OS comienza a trabajar cuando se enciende el computador gestionando el hardware y software de la misma.

OSI: (Open System Interconnection) un estándar ISO para las comunicaciones en todo el mundo que define un marco para la aplicación de protocolos de redes en siete capas. El control se pasa de una capa a la otra, empezando por el nivel de aplicación en una estación, de proceder a la capa inferior, a través del canal a la siguiente estación y respaldo de la jerarquía.

Pager: pequeño dispositivo de telecomunicación donde se reciben mensajes que aparecen escritos en un display. La comunicación se establece por teléfono y también por e-mail y es de una sola vía: el usuario debe responder el llamado comunicándose por otro medio. Varias firmas han anunciado que ofrecerán un servicio de doble vía, es decir, con posibilidad de respuesta.

PDU: (Protocolo de unidad de datos) una unidad de datos que se especifica en un protocolo de un determinado nivel y que consiste en el control de la información de protocolo de la capa dada.

PHP: Permite a los desarrolladores web, crear contenido dinámico que interactúa con bases de datos. Aplicaciones PHP se encuentran normalmente en los servidores de Linux y en relación con bases de datos MySQL
Rrdgraph: Round Robin funciones de base de datos herramienta de representación gráfica.

Plugins: Software del módulo que añade una específica función o servicio a un sistema más grande.

POP3: (Post Office Protocol 3 - Protocolo 3 de Correo). Es un protocolo estándar para recibir mensajes de e-mail. Los mensajes de e-mails enviados a un servidor, son almacenados por el servidor pop3. Cuando el usuario se conecta al mismo (sabiendo la dirección POP3, el nombre de usuario y la contraseña), puede descargar los ficheros.

Pop-ups: Son ventanas no abiertas por el usuario que aparecen al acceder a una página. Normalmente aparece el parte superior de la página. El usuario puede cerrar este tipo de ventanas. El pop under aparece al cerrar una ventana del navegador. Normalmente son utilizados con fines publicitarios.

Portlets: Un portlet es un componente Web hecho en Java y manejado a través de un contenedor de portlets que procesa las peticiones de los clientes y produce contenido dinámico.

PostgreSQL: Es la base de datos open source más avanzada del mundo. Usada en empresas, universidades, instituciones públicas y privadas.

Protocolo: Es una descripción formal de formatos de mensaje y las normas para el intercambio de los mensajes.

Proxy: Un proxy web es utilizado para interceptar la navegación de páginas web por motivos de seguridad, anonimato, rendimiento, etc.

Python: Lenguaje de programación orientado-objeto desarrollado por Guido van Rossum.

RMON: Es un protocolo que permite a varios monitores de red y sistemas de la consola para intercambiar los datos de control de la red, proporcionar información relacionada con errores de red y su utilización.

Routers: Dispositivo hardware o software para interconexión de redes de computadoras que opera en la capa tres (nivel de red) del modelo OSI. El router interconecta segmentos de red o redes enteras. Hace pasar paquetes de datos entre redes tomando como base la información de la capa de red.

RRA: Round Robin Archives, son archivos de datos de una base de datos RRD.

RRDTOOL: Sistema para almacenar y presentar datos en series temporales. Su propósito principal es crear una representación gráfica agradable, pero también puede generar un informe numérico.

Scripts: Conjunto de instrucciones generalmente almacenadas en un archivo de texto que deben ser interpretados línea a línea en tiempo real para su ejecución, se distinguen de los programas, pues deben ser convertidos a un archivo binario ejecutable para correrlos.

SHA: (Secure Hash Algorithm) Un popular algoritmo hash unidireccional que se utiliza para crear firmas digitales. SHA fue desarrollado por el NIST, y SHA-1 es una revisión a la norma lanzado en 1994. SHA-1 es similar a la MD4 y

MD5 algoritmos desarrollados por Rivest, pero es ligeramente más lento y más seguro.

SMS: (Short Message Service) Servicio de Mensaje Corto. Es un servicio de mensajería por teléfonos celulares. Con este sistema se puede enviar o recibir mensajes entre celulares y otros dispositivos electrónicos, e incluso utilizando internet.

SMTP: (Simple Mail Transfer Protocol) es un protocolo TCP / IP protocolo utilizado para enviar y recibir correo electrónico. Se define un formato de mensaje y un procedimiento para enrutar los mensajes a través de Internet desde el origen al destino a través de servidores de correo electrónico.

SNMP: Protocolo simple de administración de red (Simple Network Management Protocol). Es un protocolo que les permite a los administradores de red administrar dispositivos de red y diagnosticar problemas en la red. Es parte de la familia de protocolos TCP/IP.

SQL: (Structured Query Language) Lenguaje utilizado para base de datos, SQL es un lenguaje de definición de datos (LDD), un lenguaje de definiciones de vistas (LDV) y un lenguaje de manipulación de datos (LMD), que posee también capacidad para especificar restricciones y evolución de esquemas.

SSH: (Secure SHell). Protocolo seguro que sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, y también puede redirigir el tráfico de X para poder ejecutar programas gráficos si tenemos un Servidor X arrancado. Fue diseñado desde el principio para ofrecer un máximo de seguridad y permitir el acceso remoto a servidores de forma segura.

SWAP: Técnica que permite que una computadora simule más memoria principal de la que posee. La técnica es usada por la mayoría de los sistemas operativos actuales. Ver memoria virtual.

Switch: Es el dispositivo analógico que permite interconectar redes operando en la capa 2 o de nivel de enlace de datos del modelo OSI u Open Systems Interconnection.

Syslogs: Es un servicio de registro de datos que es muy frecuentemente usado en entornos Linux y Unix. El concepto tras Syslogs es que el registro de sucesos e información es enteramente manejado por un servidor dedicado llamado 'Servidor Syslog'.

TCP: (Transfer control protocol) Es un protocolo de transporte fiable en el protocolo TCP / IP. TCP garantiza que los datos llegan con precisión y el 100% intacta en el otro extremo.

TCP/Ip: Conjunto de protocolos de red en los que se basa Internet y que permiten la transmisión de datos entre redes de computadoras. Los dos protocolos más importantes que la componen: Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP).

Telnet: Telnet es un comando de usuario y una subyacente TCP / IP de protocolo de acceso a equipos remotos. A través de Telnet, un administrador u otro usuario pueden acceder a algún otro equipo de forma remota.

Throughput: Volumen de trabajo o de información que fluye a través de un sistema. Así también se le llama al volumen de información que fluye en las redes de datos.

TMN: (Telecommunication managment network, red de gestión de las telecomunicaciones), es un modelo de protocolo para la gestión de los sistemas abiertos en una red de comunicaciones.

Token Ring: Es una arquitectura de red desarrollada por IBM en los años 1970 con topología lógica en anillo y técnica de acceso de paso de testigo.

Trap: Es un código o una señal diseñada para capturar los errores y poner de manifiesto dónde están, es un tipo de PDU empleados para informar de un evento o otros asincrónica de alerta acerca de un subsistema de gestión.

UDP: Protocolo de Datagrama de Usuario (en inglés User Datagram Protocol) un protocolo sin conexión que, funciona en redes IP, no establece un diálogo previo entre las dos partes, ni tampoco mecanismos de detección de errores.

Umbral: El umbral es la cantidad mínima de señal que ha de estar presente para ser registrada por un sistema.

URLS: Son las siglas de Localizador de Recurso Uniforme (en inglés Uniform Resource Locator), la dirección global de documentos y de otros recursos en la World Wide Web.

Vlan: (Virtual LAN, 'Red de Área Local Virtual') es un método de crear redes lógicamente independientes dentro de una misma red física. Varias VLANs pueden coexistir en un único conmutador físico o en una única red física. Una VLAN consiste en una red de ordenadores que se comportan como si estuviesen conectados al mismo conmutador, aunque pueden estar en realidad conectados físicamente a diferentes segmentos de una red de área local.

VPN: Red Privada Virtual (RPV) o Virtual Private Network (VPN) supone una tecnología de red que, por razones de costo y comodidad, brinda la posibilidad

de conectarse a una red pública generando una extensión a nivel de área local. Por caso, este tipo de redes se utilizan a la hora de conectar dos o más oficinas de una empresa a través de Internet. Esto facilita la conexión y el intercambio a un bajo costo económico, y permite que miembros de un mismo equipo se conecten entre sí desde locaciones remotas.

WAN: (Wide Area Network - Red de Área Extensa). WAN es una red de computadoras de gran tamaño, generalmente dispersa en un área metropolitana, a lo largo de un país o incluso a nivel planetario.

WMI: (Windows Management Instrumentation). Una interfaz de programación de Windows que permite que el sistema y los dispositivos de red puedan ser configurados y administrados.

WMware player: Programa de software desarrollado por la empresa VMware Inc. (disponible de forma gratuita), para ejecutar máquinas virtuales huéspedes (guest) producidas por otros productos de VMware.

XML: Son las siglas de Extensible Markup Language, una especificación/lenguaje de programación desarrollada por el W3C. XML es una versión de SGML, diseñado especialmente para los documentos de la web. Permite que los diseñadores creen sus propias etiquetas, permitiendo la definición, transmisión, validación e interpretación de datos entre aplicaciones y entre organizaciones.

XSS: Cross-Site-Scripting Problema de seguridad en las páginas web, generalmente por vulnerabilidades en el sistema de validación de datos entrantes. Un ataque XSS consiste en enviar un script malicioso a la página, ocultándolo entre solicitudes legítimas.

Zenpacks: Los ZenPacks son grupos empaquetados de funciones y modelos de plantillas para tipos específicos de dispositivos.

Zope: “Z Object Publishing Enviroment", un entorno para publicar objetos. Es un servidor de aplicaciones libre, integra un servidor Web, un lenguaje de srcipt (Python) y un servidor de bases de datos.