Universidad Nacional de Loja

ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y DE LOS RECURSOS NATURALES NO RENOVABLES

Carrera de Ingeniería en Sistemas

"HERRAMIENTA DE MONITOREO PARA LA RED DE LA UNIVERSIDAD NACIONAL DE LOJA."

Proyecto de tesis previa a la obtención del título de Ingeniero en Sistemas

AUTOR: **Rolando Antonio Viñan Riofrío**

DIRECTOR: **Mg.Sc. Edgar Jamil Ramón Carrión**

LOJA – ECUADOR
2010

AUTORÍA

Los conceptos, ideas, procedimientos, análisis, investigación de campo y más elementos de juicio, son de exclusiva responsabilidad del autor

Rolando Antonio Viñan Riofrío

UNIVERSIDAD NACIONAL DE LOJA
AEIRNNR

CERTIFICACIÓN

Loja, Julio del 2010

Sr. Mg.Sc. Edgar Jamil Ramón Carrión

DIRECTOR DE TESIS.

Certifica:

Que el presente trabajo de investigación titulado **"HERRAMIENTA DE MONITOREO PARA LA RED DE LA UNIVERSIDAD NACIONAL DE LOJA"**, presentado por EL Sr. Rolando Antonio Viñan Riofrío, previo a la obtención del Título de Ingeniero en Sistemas, ha sido prolijamente revisado y al haberse acogido a las sugerencias, y corregidas las partes pertinentes autorizo su presentación y sustentación.

Mg.Sc. Edgar Jamil Ramón Carrión
DIRECTOR DE TESIS

DECLARATORIA

DE AUTORÍA

El Autor declaro que el presente trabajo investigativo pasa hacer exclusividad de la unl para propósitos que estime convenientes.

Rolando Antonio Viñan Riofrío

DEDICATORIA

A mi Dios que es quien ha sabido guiarme por el camino correcto y que me ha permitido cumplir con una de mis tantas metas, a mis queridos padres los seres más sublimes, que me inculcaron la humildad, moral y sobre todo la perseverancia para poder cumplir con todo lo propuesto en mi vida.

A MIS QUERIDO PADRES

AGRADECIMIENTO

Dejo constancia de nuestro agradecimiento a la Universidad Nacional de Loja, Área de Energía y Recursos Naturales no Renovables, a sus Directivos y Docentes por los conocimientos facilitados en la trayectoria de mi Carrera, y de manera especial al Lcdo. Jamil Ramón, por su acertada dirección, quien me brindó desinteresadamente sus valiosos conocimientos en función a su destacada capacidad profesional.

De igual forma hago extensivo mi agradecimiento a los Directivos, personal y empleados de la Universidad Nacional de Loja, quienes contribuyeron con la información necesaria para el desarrollo del presente trabajo investigativo.

Finalmente agradezco a mis familiares quienes de una u otra forma colaboraron para que mi trabajo de investigación culmine con éxito.

EL AUTOR

RESUMEN

Con el avance de las conexiones y protocolos de red, se ha hecho necesario la creación de herramientas que nos permitan monitorear los equipos pertenecientes a determinada red, sus características y el tráfico que generan en la red. Esto debido a que los administradores tienen una tarea difícil al momento de querer reconocer cuales son los equipos conectados y saber si algún equipo quiere atentar contra la seguridad.

Este sistema unlmonitoreo ha sido desarrollado como una herramienta web basada en el modelo cliente servidor utilizando para ello software libre tales como: jsp, jsf, java apache Tomcat y MySQL, lo cual ha permitido obtener un software versátil y fácil de mantener.

El propósito del sistema es optimizar el monitoreo de equipos pertenecientes a determinada red permitiéndonos identificar las características de hardware de cada equipo y el trafico generado por este.

SUMMARY

Of the advance of the connections and protocols of network, one has taken control the creation of tools that allow us to monitor the equipment pertaining to certain network, their characteristics and the traffic necessary that they generate in the network. This because the administrators have a difficult task at the time of wanting to recognize as is the connected equipment and to know if some equipment wants to attempt against the security.

This unlmonitoreo system has been developed like a tool Web based on the model client servant using for it free software such as: jsp, jsf, java apache Tomcat and MySQL, which has allowed to obtain a software versatile and easy to maintain.

The intention of the system is to optimize the monitoring of equipment pertaining to certain network being allowed us to identify the characteristics of hardware of each equipment and the traffic generated by this.

UNIVERSIDAD NACIONAL DE LOJA
AEIRNNR

ÍNDICE

MARCO TEORICO

CAPITULO II: ANALISIS DEL SISTEMA "HERRAMIENTA DE MONITOREO PARA LA RED DE LA UNL"

CAPITULO III. DISEÑO Y MODELADO DE LA APLICACIÓN

CAPITULO IV. IMPLEMENTACION DE LA APLICACIÓN

INDICE

FIGURAS GRAFICOS y TABLAS

TABLAS

INTRODUCCIÓN

En la actualidad existen muchas formas de acceder a la información, entendiendo por este término como la unión de un conjunto de datos, que al ser transformada individualmente hace surgir al conocimiento. Tomando conciencia que el conocimiento no es privilegio de un solo individuo y que es un deber la difusión de la información, se debe buscar herramientas idóneas que permitan el acceso rápido a la información, otorgando una mayor facilidad a las personas que acceden a ella.

Para lo cual la comunicación y las telecomunicaciones en sus inicios surgen por la necesidad de intercambiar información a grandes distancias a través de un medio o canal de comunicación por medio de señales.

Es por esto y por los avances tecnológicos que se ha permitido una conectividad entre diferentes dispositivos de hardware de distintas marcas y proveedores y que convivan en un escenario computacional, compartiendo acceso a la base de datos, programas internos y aplicaciones gracias a la cual surge un gran campo denominado redes de comunicación, estas redes permiten la interconexión de varios puntos, en este caso computadoras enlazadas entre sí mediante cables.

Un aspecto relevante a tomar en cuenta, es el software para redes ya que es necesario implementar varias herramientas destinadas a solucionar problemas de seguridad, confiabilidad y protección de datos. Tal es el caso de la Herramienta de monitoreo para la red de la Universidad Nacional de Loja.

El software unlmonitoreo permite monitorear las características de hardware y tráfico de la red en equipos conectados a determinada red, además se puede crear comunidades ya sea de carácter público o privada y distinguir entre host, routers y servidores. También con las propiedades OID (Identificador de Objetos) podemos conocer el tipo de objeto a monitorear siempre y cuando estos objetos tengan habilitado el servicio de SNMP (Protocolo Simple de Administración de Red).

METODOLOGÍA

El desarrollo del software se inicia con la determinación de los requerimientos o las necesidades específicas que tiene el personal que Administra la Red de la Universidad Nacional de Loja, es decir se obtienen las dificultades a las que se enfrentan cuando realizan el monitoreo. Para ello se utilizaran técnicas como la encuesta, la entrevista y la observación directa.

Una vez efectuadas estas actividades, se realiza una descripción de la situación actual y de los requerimientos que se utilizaran para elaborar el software.

Se aplicará el método analítico y sintético que permitirá descomponer la información recolectada, desintegrarla en partes y llegar a una conclusión luego de un minucioso análisis, la misma que servirá de guía para poder establecer los diferentes modelos y las posibles soluciones a la problemática planteada.

Para el presente desarrollo de software se utilizo Metodología Orientada a Objetos (aplicando el método ICONIX) el cual permitirá construir un modelo de dominio. Esta metodología usa UML el cual genera un sistema de diagramas tanto de la parte estática (modelo de dominio y diagrama de clases), como de la parte dinámica (casos de uso, diagramas de robustez y diagramas de secuencia) y algunas técnicas que nos permitirá programar de forma rápida y eficiente.

“Los pasos básicos en el proceso ICONIX son: análisis de requerimientos, análisis y diseño preliminar, diseño, desarrollo e implementación.

1. ***Análisis de requerimientos***

- Identificar los objetos y las relaciones de agregación y generalización en el modelo de dominio(Diagrama de Clases)
- Identificar casos de uso
- Organizar casos de uso en grupos (paquetes)
- Asociar requerimientos funcionales a casos de uso y objetos del dominio

2. Análisis y diseño preliminar

- Describir los caso de uso como flujo de acciones:
  - Acciones alternos y de excepción
- Análisis de robustez
  - Identificar los objetos que participan en los casos de uso
- Actualizar los diagramas de clase con las clases y atributos

3. Diseño

- Identificar los mensajes que deben ser enviados entre objetos, además los objetos y asociaciones que deben ser invocados. Secuencia de Diagramas. Además se muestra en el diagrama de colaboración los principales procesos entre objetos.
- Terminar modelo estático
- Verificar el cumplimiento de los requerimientos

4. Implementación

- Utilizar el diagrama de componentes que describe los elementos físicos y sus relaciones en el entorno de la realización.
- Escribir el código
- Hacer uso de los componentes en diferentes aplicaciones
- Modificar con facilidad el software
- Pruebas de sistema y aceptación basadas en casos de uso
- Test de verificación con los usuarios."[1]

---

1 FERNÁNDEZ Juan y SUMANO María, 2004. ICONIX Notas del método con ampliaciones y mejoras. [Diapositivas]. [http:// www.uv.mx/jfernandez/cursos_archivos/CICONIX.PPT] http://www.portalhuarpe.com.ar/Seminario09/archivos/MetodologiaICONIX.pdf

MARCO TEORICO

CAPITULO I

PROTOCOLO TCP/IP Y SNMP

1. PROTOCOLO TCP/IP

1.1 HISTORIA YCONCEPTO ARQUITECTURA DE TCP/IP

El Protocolo de Internet (IP) y el Protocolo de Transmisión (TCP), fueron desarrollados inicialmente en 1973 por el informático estadounidense Vinton Cerf como parte de un proyecto dirigido por el ingeniero norteamericano Robert Kahn y patrocinado por la Agencia de Programas Avanzados de Investigación (ARPA, siglas en inglés) del Departamento Estadounidense de Defensa. Internet comenzó siendo una red informática de ARPA (llamada ARPANET) que conectaba redes de ordenadores de varias universidades y laboratorios en investigación en Estados Unidos. World Wibe Web se desarrolló en 1989 por el informático británico Timothy Berners-Lee para el Consejo Europeo de Investigación Nuclear (CERN, siglas en francés).

TCP/IP es el protocolo común utilizado por todos los ordenadores conectados a Internet, de manera que éstos puedan comunicarse entre sí. Hay que tener en cuenta que en Internet se encuentran conectados ordenadores de clases muy diferentes con *hardware* y *software* incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Aquí se encuentra una de las grandes ventajas del TCP/IP, pues este protocolo se encargará de que la comunicación entre todos sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de *hardware*.

TCP/IP no es un único protocolo, sino que es en realidad lo que se conoce con este nombre es un conjunto de protocolos que cubren los distintos niveles del modelo OSI. Los dos protocolos más importantes son el TCP (Transmission Control Protocol) y el IP (Internet Protocol), que son los que dan nombre al conjunto.

1.1.1.1. Presentación del modelo OSI

OSI significa Interconexión de sistemas abiertos. Este modelo fue establecido por ISO para implementar un estándar de comunicación entre equipos de una red, esto es, las reglas que administran la comunicación entre equipos. De hecho, cuando surgieron las redes, cada fabricante contaba con su propio sistema (hablamos de un sistema patentado), con lo cual coexistían diversas redes incompatibles. Por esta razón, fue necesario establecer un estándar.

La función del modelo OSI es estandarizar la comunicación entre equipos para que diferentes fabricantes puedan desarrollar productos (software o hardware) compatibles (siempre y cuando sigan estrictamente el modelo OSI).

- **La importancia de un sistema de capas:** El objetivo de un sistema en capas es dividir el problema en diferentes partes (las capas), de acuerdo con su nivel de abstracción. Cada capa del modelo se comunica con un nivel adyacente (superior o inferior). Por lo tanto, cada capa utiliza los servicios de las capas inferiores y se los proporciona a la capa superior.

1.1.1.2. EL MODELO OSI

El modelo OSI es un modelo que comprende 7 capas, mientras que el modelo TCP/IP tiene sólo 4. En realidad, el modelo TCP/IP se desarrolló casi a la par que el modelo OSI. Es por ello que está influenciado por éste, pero no sigue todas las especificaciones del modelo OSI. Las capas del modelo OSI son las siguientes:

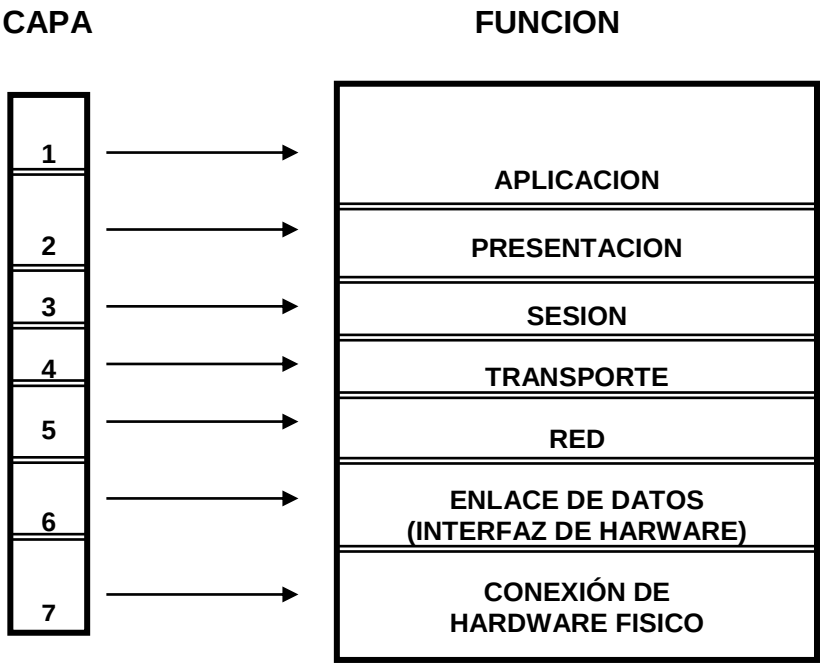


FIGURA 1.1 MODELO OSI

- **"La capa física** define la manera en la que los datos se convierten físicamente en señales digitales en los medios de comunicación (pulsos eléctricos, modulación de luz, etc.).

- **La capa de enlace de datos** define la interfaz con la tarjeta de interfaz de red y cómo se comparte el medio de transmisión.

- **La capa de red** permite administrar las direcciones y el enrutamiento de datos, es decir, su ruta a través de la red.

- **La capa de transporte** se encarga del transporte de datos, su división en paquetes y la administración de potenciales errores de transmisión.

- **La capa de sesión** define el inicio y la finalización de las sesiones de comunicación entre los equipos de la red.

- **La capa de presentación** define el formato de los datos que maneja la capa de aplicación (su representación y, potencialmente, su compresión y cifrado) independientemente del sistema.

- **La capa de aplicación** le brinda aplicaciones a la interfaz. Por lo tanto, es el nivel más cercano a los usuarios, administrado directamente por el software."[2]

1.1.2. ARQUITECTURA DE TCP/IP

La arquitectura del TCP/IP consta de cuatro niveles o capas en las que se agrupan los protocolos, y que se relacionan con los niveles OSI de la siguiente manera:

Aplicación: Se corresponde con los niveles OSI de aplicación, presentación y sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de ficheros (FTP), conexión remota (TELNET) y otros más recientes como el protocolo HTTP (*Hypertext Transfer Protocol*).

[2] El modelo Osi. [http://es.kioskea.net/contents/internet/tcpip.php3]

Transporte: Coincide con el nivel de transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP, se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos.

Internet: Es el nivel de red del modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. Es utilizado con esta finalidad por los protocolos del nivel de transporte.

Acceso de Red: Es la interfaz de la red real. TCP/IP no especifica ningún protocolo concreto, así es que corre por las interfaces conocidas, como por ejemplo: 802.2, CSMA/CD, X.25, etc.

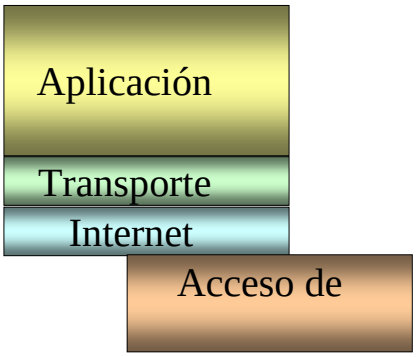


FIGURA 1.1.2 Arquitectura tcp/ip

El TCP/IP necesita funcionar sobre algún tipo de red o de medio físico que proporcione sus propios protocolos para el nivel de enlace de Internet. Por este motivo hay que tener en cuenta que los protocolos utilizados en este nivel pueden ser muy diversos y no forman parte del conjunto TCP/IP. Sin embargo, esto no debe ser problemático puesto que una de las funciones y ventajas principales del TCP/IP es proporcionar una abstracción del medio de forma que sea posible el intercambio de información entre medios diferentes y tecnologías que inicialmente son incompatibles.

Para transmitir información a través de TCP/IP, ésta debe ser dividida en unidades de menor tamaño. Esto proporciona grandes ventajas en el manejo de los datos que se

transfieren y, por otro lado, esto es algo común en cualquier protocolo de comunicaciones. En TCP/IP cada una de estas unidades de información recibe el nombre de "datagrama" (*datagram*), y son conjuntos de datos que se envían como mensajes independientes.

1.1.3. CARACTERÍSTICAS DE TCP/IP

Dentro de un sistema TCP/IP los datos transmitidos se dividen en pequeños paquetes, éstos resaltan una serie de características.

- La tarea de IP es llevar los datos los paquetes de un sitio a otro. Las computadoras que encuentran las vías para llevar los datos de una red a otra (denominadas enrutadores) utilizan IP para trasladar los datos. En resumen *IP mueve los paquetes de datos a grandes cantidades, mientras TCP se encarga del flujo y asegura que los datos estén correctos.*

- Las líneas de comunicación se pueden compartir entre varios usuarios. Cualquier tipo de paquete puede transmitirse al mismo tiempo, y se ordenará y combinará cuando llegue a su destino. Compare esto con la manera en que se transmite una conversación telefónica. Una vez que establece una conexión, se reservan algunos circuitos para usted, que no puede emplear en otra llamada, aun si deja esperando a su interlocutor por veinte minutos.

- Los datos no tienen que enviarse directamente entre dos computadoras.
  Cada paquete pasa de computadora en computadora hasta llegar a su destino.
  Éste, es el secreto de cómo se pueden enviar datos y mensajes entre dos computadoras aunque no estén conectadas directamente entre sí. Lo que realmente sorprende es que sólo se necesitan algunos segundos para enviar un archivo de buen tamaño de una máquina a otra, aunque estén separadas por miles de kilómetros y pese a que los datos tienen que pasar por múltiples computadoras.
  Una de las razones de la rapidez es que, cuando algo anda mal, sólo es necesario volver a transmitir un paquete, no todo el mensaje.

- Los paquetes no necesitan seguir la misma trayectoria. La red puede llevar cada paquete de un lugar a otro y usar la conexión más idónea que esté

disponible en ese instante. No todos los paquetes de los mensajes tienen que viajar, necesariamente, por la misma ruta, ni necesariamente tienen que llegar todos al mismo tiempo.

- La flexibilidad del sistema lo hace muy confiable. Si un enlace se pierde, el sistema usa otro. Cuando usted envía un mensaje, el TCP divide los datos en paquetes, ordena éstos en secuencia, agrega cierta información para control de errores y después los lanza hacia fuera, y los distribuye. En el otro extremo, el TCP recibe los paquetes, verifica si hay errores y los vuelve a combinar para convertirlos en los datos originales. De haber error en algún punto, el programa TCP destino envía un mensaje solicitando que se vuelvan a enviar determinados paquetes.

1.1.4. PROTOCOLOS TCP/IP

APLICACIÓN →	FTP, SMTP, TELNET	SNMP, X-WINDOWS RPC, NFS
TRANSPORTE →	TCP	UDP
INTERNET →	IP, ICMP, 802.2, X.25	
ACCESO DE RED →	ETHERNET, IEEE 802.2, X.25	

FIGURA 1.1.3 Protocolos

1.1.4.1. Nivel de Aplicación

Constituye el nivel más alto de la torre tcp/ip. A diferencia del modelo OSI, se trata de un nivel simple en el que se encuentran las aplicaciones que acceden a servicios disponibles a través de Internet. Estos servicios están sustentados por una serie de protocolos que los proporcionan.

- **FTP (File Transfer Protocol).** Se utiliza para transferencia de archivos.

- **SMTP (Simple Mail Transfer Protocol)**. Es una aplicación para el correo electrónico.

- **TELNET:** Permite la conexión a una aplicación remota desde un proceso o terminal.

- **RPC (Remote Procedure Call).** Permite llamadas a procedimientos situados remotamente. Se utilizan las llamadas a RPC como si fuesen procedimientos locales.
- **SNMP (Simple Network Management Protocol).** Se trata de una aplicación para el control de la red.

- **NFS (Network File System).** Permite la utilización de archivos distribuidos por los programas de la red.

- **XWindows**. Es un protocolo para el manejo de ventanas e interfaces de usuario.

1.1.4.2. Nivel de Transporte

Este nivel proporciona una comunicación extremo a extremo entre programas de aplicación. La maquina remota recibe exactamente lo mismo que le envió la maquina origen. En este nivel el emisor divide la información que recibe del nivel de aplicación en paquetes, le añade los datos necesarios para el control de flujo y control de errores, y se los pasa al nivel de red junto con la dirección de destino.

En el receptor este nivel se encarga de ordenar y unir las tramas para generar de nuevo la información original.

Para implementar el nivel de transporte se utilizan dos protocolos:

- **UDP:** proporciona un nivel de transporte no fiable de datagramas, ya que apenas añade información al paquete que envía al nivel inferior, solo la necesaria para la comunicación extremo a extremo. Lo utilizan aplicaciones como NFS y RPC, pero sobre todo se emplea en tareas de control.

- **TCP (Transport Control Protocolo):** es el protocolo que proporciona un transporte fiable de flujo de bits entre aplicaciones. Está pensado para poder enviar grandes cantidades de información de forma fiable, liberando al programador de aplicaciones de la dificultad de gestionar la fiabilidad de la

conexión (retransmisiones, perdidas de paquete, orden en que llegan los paquetes, duplicados de paquetes,...) que gestiona el propio protocolo. Pero la complejidad de la gestión de la fiabilidad tiene un coste en eficiencia, ya que para llevar a cabo las gestiones anteriores se tiene que añadir bastante información a los paquetes a enviar. Debido a que los paquetes a enviar tienen un tamaño máximo, como mas información añada el protocolo para su gestión, menos información que proviene de la aplicación podrá contener ese paquete. Por eso, cuando es más importante la velocidad que la fiabilidad, se utiliza UDP, en cambio TCP asegura la recepción en destino de la información a transmitir.

1.1.4.3. Nivel de Internet

Coloca la información que le pasa el nivel de transporte en datagramas IP, le añade cabeceras necesaria para su nivel y lo envía al nivel inferior. Es en este nivel donde se emplea el algoritmo de encaminamiento, al recibir un datagrama del nivel inferior decide, en función de su dirección, si debe procesarlo y pasarlo al nivel superior, o bien encaminarlo hacia otra máquina. Para implementar este nivel se utilizan los siguientes protocolos:

- **IP (Internet Protocol):** es un protocolo no orientado a la conexión, con mensajes de un tamaño máximo. Cada datagrama se gestiona de forma independiente, por lo que dos datagramas pueden utilizar diferentes caminos para llegar al mismo destino, provocando que lleguen en diferente orden o bien duplicados. Es un protocolo no fiable, eso quiere decir que no corrige los anteriores problemas, ni tampoco informa de ellos. Este protocolo recibe información del nivel superior y le añade la información necesaria para su gestión (direcciones IP , checksum)

- **ICMP (Internet Control Message Protocol):** proporciona un mecanismo de comunicación de información de control y de errores entre maquinas intermedias por las que viajaran los paquetes de datos. Esto datagramas los suelen emplear las maquinas (gateways, host,...) para informarse de condiciones especiales en la red, como la existencia de una congestión, la existencia de errores y las posibles peticiones de cambios de ruta. Los mensajes de ICMP están encapsulados en datagramas IP.

- **IGMP (Internet Group Management Protocol):** este protocolo está íntimamente ligado a IP. Se emplea en maquinas que emplean IP multicast. El IP multicast es una variante de IP que permite emplear datagramas con múltiples destinatarios.

También en este nivel tenemos una serie de protocolos que se encargan de la resolución de direcciones:

- **ARP (Address Resolution Protocol):** cuando una maquina desea ponerse en contacto con otra conoce su dirección IP, entonces necesita un mecanismo dinámico que permite conocer su dirección física. Entonces envía una petición ARP por broadcast (o sea a todas las maquinas). El protocolo establece que solo contestara a la petición, si esta lleva su dirección IP. Por lo tanto solo contestara la maquina que corresponde a la dirección IP buscada, con un mensaje que incluya la dirección física. El software de comunicaciones debe mantener una cache con los pares IP-dirección física. De este modo la siguiente vez que hay que hacer una transmisión a esa dirección IP, ya conoceremos la dirección física.

- **RARP (Reverse Address Resolution Protocol):** a veces el problema es al revés, o sea, una máquina solo conoce su dirección física, y desea conocer su dirección lógica. Esto ocurre, por ejemplo, cuando se accede a Internet con una dirección diferente, en el caso de PC que acceden por módem a Internet, y se le asigna una dirección diferente de las que tiene el proveedor sin utilizar. Para solucionar esto se envía por broadcast una petición RARP con su dirección física, para que un servidor pueda darle su correspondencia IP.

- **BOOTP (Bootstrap Protocol):** el protocolo RARP resuelve el problema de la resolución inversa de direcciones, pero para que pueda ser más eficiente, enviando más información que meramente la dirección IP, se ha creado el protocolo BOOTP. Este además de la dirección IP del solicitante, proporciona información adicional, facilitando la movilidad y el mantenimiento de las maquinas.

1.1.4.4. Nivel de Acceso de Red

“Este nivel se limita a recibir datagramas del nivel superior (nivel de red) y transmitirlo al hardware de la red. Pueden usarse diversos protocolos: DLC (IEEE 802.2), Frame Relay, X.25, etc.

La interconexión de diferentes redes genera una red virtual en la que las maquinas se identifican mediante una dirección de red lógica. Sin embargo a la hora de transmitir información por un medio físico se envía y se recibe información de direcciones físicas. Un diseño eficiente implica que una dirección lógica sea independiente de una dirección física, por lo tanto es necesario un mecanismo que relacione las direcciones lógicas con las direcciones físicas. De esta forma podremos cambiar nuestra dirección lógica IP conservando el mismo hardware, del mismo modo podremos cambiar una tarjeta de red, la cual contiene una dirección física, sin tener que cambiar nuestra dirección lógica IP. ”[3]

1.1.5. ESTÁNDARES DE PROTOCOLO

El modelo OSI se utiliza para definir los protocolos que se tienen que utilizar en cada nivel. Los productos de distintos fabricantes que se ajustan a este modelo se pueden comunicar entre sí.

La ISO, el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE), ANSI (Instituto de Estandarización Nacional Americano), CCITT (Comité Consultivo Internacional de Telegrafía y Telefonía), ahora llamado ITU (Unión Internacional de Telecomunicaciones) y otros organismos de estandarización han desarrollado protocolos que se correspondan con algunos de los niveles del modelo OSI.

1.1.5.1. Los protocolos de IEEE a nivel físico son:

- **802.3 (Ethernet)**. Es una red lógica en bus que puede transmitir datos a 10 Mbps Los datos se transmiten en la red a todos los equipos. Sólo los equipos que tenían que recibir los datos informan de la transmisión. El protocolo de acceso de múltiple con detección de portadora con detección de colisiones (CSMA/CD) regula el tráfico de la red permitiendo la transmisión sólo cuando la red esté despejada y no haya otro equipo transmitiendo.

- **802.4 (Token Bus)**. Es una red en bus que utiliza un esquema de paso de testigo. Cada equipo recibe todos los datos, pero sólo los equipos en los que

[3] Que es Tcp/ip.[http://www.mailxmail.com/curso-que-son-redes/que-es-tcp-ip]

coincida la dirección responderán. Un testigo que viaja por la red determina quién es el equipo que tiene que informar.

- **802.5 (Token Ring)**. Es un anillo lógico que transmite a 4 ó a 16 Mbps Aunque se le llama en anillo, está montada como una estrella ya que cada equipo está conectado a un hub. Realmente, el anillo está dentro del hub. Un Token a través del anillo determina qué equipo puede enviar datos.

El IEEE definió estos protocolos para facilitar la comunicación en el subnivel de Control de acceso al medio (MAC).

Un controlador MAC está situado en el subnivel de Control de acceso al medio; este controlador de dispositivo es conocido como controlador de la NIC. Proporciona acceso a bajo nivel a los adaptadores de red para proporcionar soporte en la transmisión de datos y algunas funciones básicas de control del adaptador.

Un protocolo MAC determina qué equipo puede utilizar el cable de red cuando varios equipos intenten utilizarlo simultáneamente. CSMA/CD, el protocolo 802.3, permite a los equipos transmitir datos cuando no hay otro equipo transmitiendo. Si dos máquinas transmiten simultáneamente se produce una colisión. El protocolo detecta la colisión y detiene toda transmisión hasta que se libera el cable. Entonces, cada equipo puede volver a tratar de transmitir después de esperar un período de tiempo aleatorio.

1.1.6. CÓMO FUNCIONA TCP/IP

1.1.6.1. TCP (*Transmission-Control-Protocol)*:

TCP (*Protocolo de Control de Transmisión*) es uno de los principales protocolos de la capa de transporte del modelo TCP/IP. En el nivel de aplicación, posibilita la administración de datos que vienen del nivel más bajo del modelo, o van hacia él, (es decir, el protocolo IP). Cuando se proporcionan los datos al protocolo IP, los agrupa en datagramas IP, fijando el campo del protocolo en 6 (para que sepa con anticipación que el protocolo es TCP). TCP es un protocolo orientado a conexión, es decir, que permite que dos máquinas que están comunicadas controlen el estado de la transmisión.

Las principales características del protocolo TCP son las siguientes:

- Permite colocar los datagramas nuevamente en orden cuando vienen del protocolo IP.

- Que permite monitoreo del flujo de los datos y así evita la saturación de la red.

- Que los datos se formen en segmentos de longitud variada para "entregarlos" al protocolo IP.

- Permite multiplexar los datos, es decir, que la información que viene de diferentes fuentes (por ejemplo, aplicaciones) en la misma línea pueda circular simultáneamente.

- TCP permite comenzar y finalizar la comunicación amablemente.

1.1.6.1.1. El objetivo de TCP

Con el uso del protocolo TCP, las aplicaciones pueden comunicarse en forma segura (gracias al sistema de acuse de recibo del protocolo TCP) independientemente de las capas inferiores. Esto significa que los routers (que funcionan en la capa de Internet) sólo tienen que enviar datos en forma de datagramas, sin preocuparse con el monitoreo de datos porque esta función la cumple la capa de transporte (o más específicamente el protocolo TCP).

Durante una comunicación usando el protocolo TCP, las dos máquinas deben establecer una conexión. La máquina emisora (la que solicita la conexión) se llama cliente, y la máquina receptora se llama servidor. Por eso estamos en un entorno Cliente-Servidor.
Las máquinas de dicho entorno se comunican en modo en línea, es decir, que la comunicación se realiza en ambas direcciones.

Para posibilitar la comunicación y que funcionen bien todos los controles que la acompañan, los datos se agrupan; es decir, que se agrega un encabezado a los paquetes de datos que permitirán sincronizar las transmisiones y garantizar su recepción.

Otra función del TCP es la capacidad de controlar la velocidad de datos usando su capacidad para emitir mensajes de tamaño variable. Estos mensajes se llaman *segmentos*.

1.1.6.1.2. La función multiplexión

TCP posibilita la realización de una tarea importante: multiplexar/demultiplexar; es decir transmitir datos desde diversas aplicaciones en la misma línea o, en otras palabras, ordenar la información que llega en paralelo.

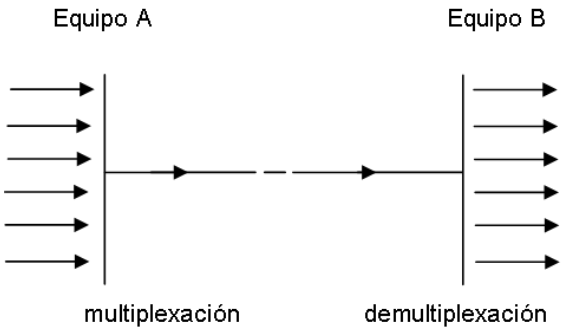


FIGURA 1.1.4 Función Multiplexión

Estas operaciones se realizan empleando el concepto de puertos (o conexiones), es decir, un número vinculado a un tipo de aplicación que, cuando se combina con una dirección de IP, permite determinar en forma exclusiva una aplicación que ejecuta en una máquina determinada.

1.1.6.1.3. El formato de los datos en TCP

“Un segmento TCP está formado de la siguiente manera:

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Puerto de origen	Puerto Destino
Numero de Secuencia	
Numero de acuse de recibo	

Margen de Datos	Reservado	URG	ACK	PSH	RST	SYN	FIN	Ventana

Suma de Control	Puntero urgente
Opciones	Relleno
Datos	

FIGURA 1.1.5 Datos de tcp

Significado de los diferentes campos:

- **Puerto de origen** (16 bits): Puerto relacionado con la aplicación en curso en la máquina origen

- **Puerto de destino** (16 bits): Puerto relacionado con la aplicación en curso en la máquina destino

- **Número de secuencia** (32 bits): Cuando el indicador SYN está fijado en 0, el número de secuencia es la primera palabra del segmento actual. Cuando SYN está fijado en 1, el número de secuencia es igual al número de secuencia inicial utilizado para sincronizar los números de secuencia (ISN).

- **Número de acuse de recibo** (32 bits): También llamado número de descargo se relaciona con el número (secuencia) del último segmento esperado y no el número del último segmento recibido.

- **Margen de datos** (4 bits): Esto permite ubicar el inicio de los datos en el paquete. Aquí, el margen es fundamental porque el campo opción es de tamaño variable.

- **Reservado** (6 bits): Un campo que actualmente no está en uso pero se proporciona para el uso futuro.

- **Indicadores** (6x1 bit): Los indicadores representan información adicional:

  - **URG**: Si este indicador está fijado en 1, el paquete se debe procesar en forma urgente.

  - **ACK**: Si este indicador está fijado en 1, el paquete es un acuse de recibo.

- **PSH** (PUSH): Si este indicador está fijado en 1, el paquete opera de acuerdo con el método PUSH.
- **RST**: Si este indicador está fijado en 1, se restablece la conexión.
- **SYN**: El indicador SYN de TCP indica un pedido para establecer una conexión.
- **FIN**: Si este indicador está fijado en 1, se interrumpe la conexión.

- **Ventana** (16 bits): Campo que permite saber la cantidad de bytes que el receptor desea recibir sin acuse de recibo.
- **Suma de control** (CRC): La suma de control se realiza tomando la suma del campo de datos del encabezado para poder verificar la integridad del encabezado.
- **Puntero urgente** (16 bits): Indica el número de secuencia después del cual la información se torna urgente.
- **Opciones** (tamaño variable): Diversas opciones
- **Relleno**: Espacio restante después de que las opciones se rellenan con ceros para tener una longitud que sea múltiplo de 32 bits.

1.1.6.1.4. Confiabilidad de las transferencias

El protocolo TCP permite garantizar la transferencia de datos confiable, a pesar de que usa el protocolo IP, que no incluye ningún monitoreo de la entrega de datagramas.

De hecho, el protocolo TCP tiene un sistema de acuse de recibo que permite al cliente y al servidor garantizar la recepción mutua de datos. Cuando se emite un segmento, se lo vincula a un **número de secuencia**. Con la recepción de un segmento de datos, la máquina receptora devolverá un segmento de datos donde el indicador ACK esté fijado en 1 (para poder indicar que es un acuse de recibo) acompañado por un número de acuse de recibo que equivale al número de secuencia anterior.

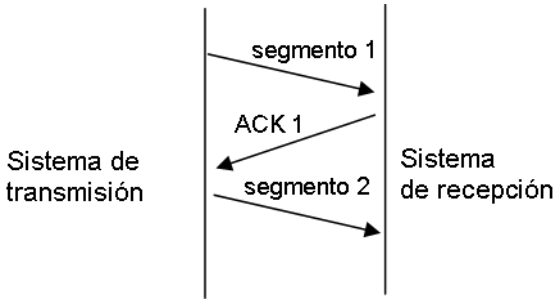


FIGURA 1.1.6 Confiabilidad de transferencias

Además, usando un temporizador que comienza con la recepción del segmento en el nivel de la máquina originadora, el segmento se reenvía cuando ha transcurrido el tiempo permitido, ya que en este caso la máquina originadora considera que el segmento está perdido.

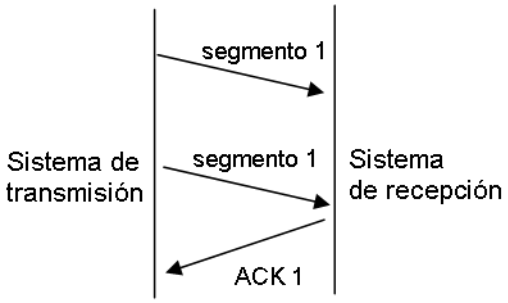


FIGURA 1.1.7 Confiabilidad de transferencias A

Sin embargo, si el segmento no está perdido y llega a destino, la máquina receptora lo sabrá, gracias al número de secuencia, que es un duplicado, y sólo retendrá el último segmento que llegó a destino.

1.1.6.1.5. Cómo establecer una conexión

Considerando que este proceso de comunicación, que se produce con la transmisión y el acuse de recibo de datos, se basa en un número de secuencia, las máquinas originadora y receptora (cliente y servidor) deben conocer el número de secuencia inicial de la otra máquina.

La conexión establecida entre las dos aplicaciones a menudo se realiza siguiendo el siguiente esquema:

- Los puertos TCP deben estar abiertos.

- La aplicación en el servidor es pasiva, es decir, que la aplicación escucha y espera una conexión.

- La aplicación del cliente realiza un pedido de conexión al servidor en el lugar donde la aplicación es abierta pasiva. La aplicación del cliente se considera "abierta activa".

Las dos máquinas deben sincronizar sus secuencias usando un mecanismo comúnmente llamado *negociación en tres pasos* que también se encuentra durante el cierre de la sesión.

Este diálogo posibilita el inicio de la comunicación porque se realiza en tres etapas, como su nombre lo indica:

- En la primera etapa, la máquina originadora (el cliente) transmite un segmento donde el indicador SYN está fijado en 1 (para indicar que es un segmento de sincronización), con número de secuencia N llamado número de secuencia inicial del cliente.

- En la segunda etapa, la máquina receptora (el servidor) recibe el segmento inicial que viene del cliente y luego le envía un acuse de recibo, que es un segmento en el que el indicador ACK está fijado en 1 y el indicador SYN está fijado en 1 (porque es nuevamente una sincronización). Este segmento incluye el número de secuencia de esta máquina (el servidor), que es el número de secuencia inicial para el cliente. El campo más importante en este segmento es el de acuse de recibo que contiene el número de secuencia inicial del cliente incrementado en 1.

- Por último, el cliente transmite un acuse de recibo, que es un segmento en el que el indicador ACK está fijado en 1 y el indicador SYN está fijado en 0 (ya no es un segmento de sincronización). Su número de secuencia está incrementado y el acuse de recibo representa el número de secuencia inicial del servidor incrementado en 1.

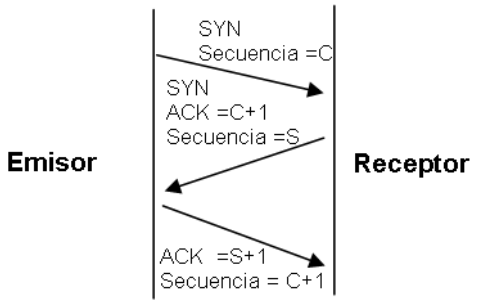


FIGURA 1.1.8 Establecer Conexión

Después de esta secuencia con tres intercambios, las dos máquinas están sincronizadas y la comunicación puede comenzar.

Existe una técnica de piratería llamada falsificación de IP, que permite corromper este enlace de aprobación con fines maliciosos.

1.1.6.1.6. Método de ventana corrediza

En muchos casos, es posible limitar la cantidad de acuses de recibo con el fin de aliviar el tráfico en la red. Esto se logra fijando un número de secuencia después del cual se requiera un acuse de recibo. Este número en realidad se guarda en el campo *ventana* del encabezado TCP/IP.

Este método se llama efectivamente el "*el método de la ventana corrediza*" porque, en cierta medida, se define una serie de secuencias que no necesitan acuses de recibo y que se desplaza a medida que se reciben los acuses de recibo.

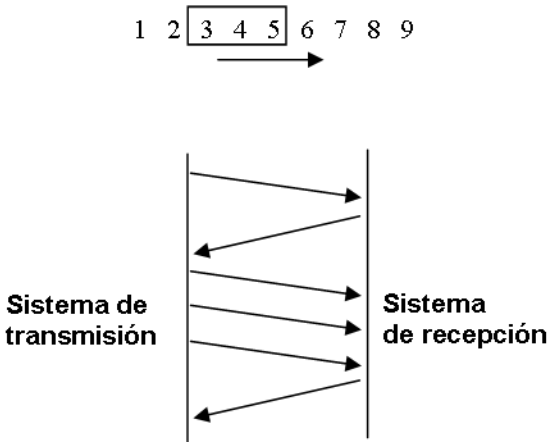


FIGURA 1.1.9 Ventana Corrediza

Además, el tamaño de esta ventana no es fijo. De hecho, el servidor puede incluir el tamaño de la ventana que considera más apropiado en sus acuses de recibo guardándolo en el campo ventana. De este modo, cuando el acuse de recibo indica un pedido para aumentar la ventana, el cliente se desplazará al borde derecho de la ventana.

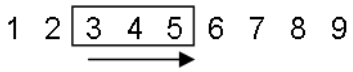


FIGURA 1.1.10 Ventana Corrediza A

Por el contrario, en el caso de una reducción, el cliente no desplazará el borde derecho de la ventana hacia la izquierda sino que esperará que avance el borde izquierdo (al llegar los acuses de recibo).

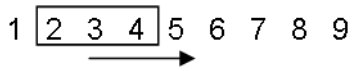


FIGURA 1.1.11 Ventana Corrediza B

1.1.6.1.7. Cómo terminar una conexión

El cliente puede pedir que se termine una conexión del mismo modo que el servidor. Para terminar una conexión se procede de la siguiente manera:

- Una de las máquinas envía un segmento con el indicador *FIN* fijado en 1, y la aplicación se auto coloca en estado de espera, es decir que deja de recibir el segmento actual e ignora los siguientes.

- Después de recibir este segmento, la otra máquina envía un acuse de recibo con el indicador *FIN* fijado en 1 y sigue enviando los segmentos en curso. Después de esto, la máquina informa a la aplicación que se ha recibido un segmento *FIN* y luego envía un segmento *FIN* a la otra máquina, que cierra la conexión. “[4]

1.1.6.2. IP (*Internet Protocol*) versión 4:

[4] Protocolo TCP [http://es.kioskea.net/contents/internet/tcp.php3]

El protocolo IP es parte de la capa de Internet del conjunto de protocolos TCP/IP. Es uno de los protocolos de Internet más importantes ya que permite el desarrollo y transporte de datagramas de IP (paquetes de datos), aunque sin garantizar su "entrega". En realidad, el protocolo IP procesa datagramas de IP de manera independiente al definir su representación, ruta y envío.

El protocolo IP determina el destinatario del mensaje mediante 3 campos:

- el campo de dirección IP: Dirección del equipo;
- el campo de máscara de subred: una máscara de subred le permite al protocolo IP establecer la parte de la dirección IP que se relaciona con la red;
- el campo de pasarela predeterminada: le permite al protocolo de Internet saber a qué equipo enviar un datagrama, si el equipo de destino no se encuentra en la red de área local.

1.1.6.2.1. Datagramas

“Los datos circulan en Internet en forma de datagramas (también conocidos como paquetes). Los datagramas son datos encapsulados, es decir, datos a los que se les agrega un encabezado que contiene información sobre su transporte (como la dirección IP de destino).

Los routers analizan (y eventualmente modifican) los datos contenidos en un datagrama para que puedan transitar.

A continuación se indica cómo se ve un datagrama:

< 32 bits >

<table>
<tr><td>Versión
(4 bits)</td><td>Longitud del encabezado
(4 bits)</td><td>Tipo de servicio
(8 bits)</td><td colspan="2">Longitud total
(16 bits)</td></tr>
<tr><td colspan="3">Identificación
(16 bits)</td><td>Indicador
(3 bits)</td><td>Margen del fragmento
(13 bits)</td></tr>
<tr><td colspan="2">Tiempo de vida
(8 bits)</td><td>Protocolo
(8 bits)</td><td colspan="2">Suma de comprobación del encabezado
(16 bits)</td></tr>
<tr><td colspan="5">Dirección IP de origen (32 bits)</td></tr>
</table>

Dirección IP de destino (32 bits)
Datos

FIGURA 1.1.12 Datagrama

A continuación se indican los significados de los diferentes campos:

Versión (4 bits): es la versión del protocolo IP que se está utilizando (se utiliza la versión 4 *IPv4*) para verificar la validez del datagrama. Está codificado en 4 bits.

- **Longitud del encabezado** o *IHL* por *Internet Header Length (Longitud del encabezado de Internet)* (4 bits): es la cantidad de palabras de 32 bits que componen el encabezado (Importante: el valor mínimo es 5). Este campo está codificado en 4 bits.

- **Tipo de servicio** (8 bits): indica la forma en la que se debe procesar el datagrama.

- **Longitud total** (16 bits): indica el tamaño total del datagrama en bytes. El tamaño de este campo es de 2 bytes, por lo tanto el tamaño total del datagrama no puede exceder los 65536 bytes. Si se lo utiliza junto con el tamaño del encabezado, este campo permite determinar dónde se encuentran los datos.

- **Identificación, indicadores y margen del fragmento** son campos que permiten la fragmentación de datagramas. Esto se explica a continuación.

- **TTL** o **Tiempo de vida** (8 bits): este campo especifica el número máximo de routers por los que puede pasar un datagrama. Por lo tanto, este campo disminuye con cada paso por un router y cuando alcanza el valor crítico de 0, el router destruye el datagrama. Esto evita que la red se sobrecargue de datagramas perdidos.

- **Protocolo** (8 bits): este campo, en notación decimal, permite saber de qué protocolo proviene el datagrama.
    - ICMP 1
    - IGMP: 2

- TCP: 6
- UDP: 17

- **Suma de comprobación del encabezado (16 bits)**: este campo contiene un valor codificado en 16 bits que permite controlar la integridad del encabezado para establecer si se ha modificado durante la transmisión. La suma de comprobación es la suma de todas las palabras de 16 bits del encabezado (se excluye el campo *suma de comprobación*). Esto se realiza de tal modo que cuando se suman los campos de encabezado (suma de comprobación inclusive), se obtenga un número con todos los bits en 1.

- **Dirección IP de origen** (32 bits): Este campo representa la dirección IP del equipo remitente y permite que el destinatario responda.

- **Dirección IP de destino** (32 bits): dirección IP del destinatario del mensaje."[5]

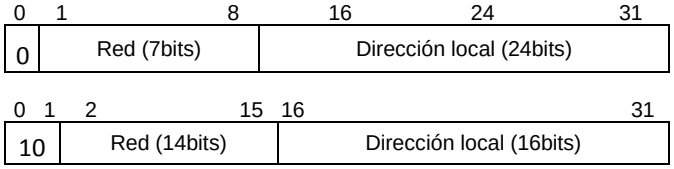
1.1.6.2.2. DIRECCIONAMIENTO IP (Internet)

El TCP/IP utiliza una dirección de 32 bits para identificar una máquina y la red a la cual está conectada.

Únicamente el NIC (Centro de Información de Red) asigna las direcciones IP (o Internet), aunque si una red no está conectada a Internet, dicha red puede determinar su propio sistema de numeración.

Hay cuatro formatos para la dirección IP, cada uno de los cuales se utiliza dependiendo del tamaño de la red.

Los cuatro formatos, Clase A hasta Clase D (aunque últimamente se ha añadido la Clase E para un futuro) aparecen en la figura:



[5] Protocolo IP [http://es.kioskea.net/contents/internet/protip.php3]

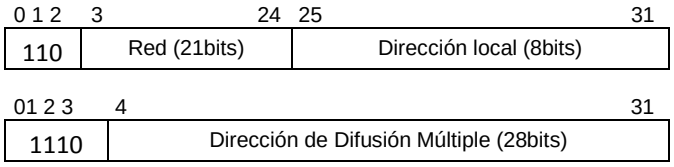


FIGURA 1.1.13 Direccionamiento IP

Conceptualmente, cada dirección está compuesta por un par (RED (netid), y Dir. Local (hostid)) en donde se identifica la red y el host dentro de la red.

La clase se identifica mediante las primeras secuencias de bits, a partir de los 3 primeros bits (de orden más alto).

Las direcciones de Clase A corresponden a redes grandes con muchas máquinas. Las direcciones en decimal son 0.1.0.0 hasta la 126.0.0.0 (lo que permite hasta 1.6 millones de hosts).

Las direcciones de Clase B sirven para redes de tamaño intermedio, y el rango de direcciones varía desde el 128.0.0.0 hasta el 191.255.0.0. Esto permite tener 16320 redes con 65024 host en cada una.

Las direcciones de Clase C tienen sólo 8 bits para la dirección local o de anfitrión (host) y 21 bits para red. Las direcciones de esta clase están comprendidas entre 192.0.1.0 y 223.255.255.0, lo que permite cerca de 2 millones de redes con 254 hosts cada una.

“Por último, **las direcciones de Clase D** se usan con fines de multidifusión, cuando se quiere una difusión general a más de un dispositivo. El rango es desde 224.0.0.0 hasta 239.255.235.255.

Cabe decir que, las **direcciones de clase E** (aunque su utilización será futura) comprenden el rango desde 240.0.0.0 hasta el 247.255.255.255.

Por tanto, las direcciones IP son cuatro conjuntos de 8 bits, con un total de 32 bits. Por comodidad estos bits se representan como si estuviesen separados por un punto, por lo que el formato de dirección IP puede ser red local. Para Clase A hasta red local para clase C.

A partir de una dirección IP, una red puede determinar si los datos se enviarán a través de una compuerta (GTW, ROUTER). Obviamente, si la dirección de la red es la misma que la dirección actual (enrutamiento a un dispositivo de red local, llamado host directo), se evitará la compuerta; pero todas las demás direcciones de red se enrutarán a una compuerta para que salgan de la red local. La compuerta que reciba los datos que se transmitirán a otra red, tendrá entonces que determinar el enrutamiento can base en la dirección IP de los datos y una tabla interna que contiene la información de enrutamiento.

Otra de las ventajas que ofrece el direccionamiento IP es el uso de direcciones de difusión (broadcast addresses), que hacen referencia a todos los host de la misma red. Según el estándar, cualquier dirección local (hostid) compuesta toda por 1s está reservada para difusión (broadcast). Por ejemplo, una dirección que contenga 32 1s se considera un mensaje difundido a todas las redes y a todos los dispositivos. Es posible difundir en todas las máquinas de una red alterando a 1s toda la dirección local o de anfitrión (hostid), de manera que la dirección 147.10.255.255 para una red de Clase B se recibiría en todos los dispositivos de dicha red; pero los datos no saldrían de dicha red.”[6]

1.1.7. EN QUE SE UTILIZA TCP/IP

Muchas grandes redes han sido implementadas con estos protocolos, incluyendo DARPA Internet "Defense Advanced Research Projects Agency Internet", en español, Red de la Agencia de Investigación de Proyectos Avanzados de Defensa. De igual forma, una gran variedad de universidades, agencias gubernamentales y empresas de ordenadores, están conectadas mediante los protocolos TCP/IP. Cualquier máquina de la red puede comunicarse con otra distinta y esta conectividad permite enlazar redes físicamente independientes en una red virtual llamada Internet. Las máquinas en Internet son denominadas "hosts" o nodos.

TCP/IP proporciona la base para muchos servicios útiles, incluyendo correo electrónico, transferencia de ficheros y login remoto.

[6] TANNEMBAUN, Redes Para Ordenadores. [http://www.radioutn.org.ar/comunicaciones/default.html]

El correo electrónico está diseñado para transmitir ficheros de texto pequeños. Las utilidades de transferencia sirven para transferir ficheros muy grandes que contengan programas o datos. También pueden proporcionar chequeos de seguridad controlando las transferencias.

El login remoto permite a los usuarios de un ordenador acceder a una máquina remota y llevar a cabo una sesión interactiva.

1.1.8. LA NUEVA VERSIÓN DE IP (IPng) versión 6

La nueva versión del protocolo IP recibe el nombre de IPv6, aunque es también conocido comúnmente como IPng (*Internet Protocol Next Generation*). El número de versión de este protocolo es el 6 (que es utilizada en forma mínima) frente a la antigua versión utilizada en forma mayoritaria. Los cambios que se introducen en esta nueva versión son muchos y de gran importancia, aunque la transición desde la versión antigua no debería ser problemática gracias a las características de compatibilidad que se han incluido en el protocolo. IPng se ha diseñado para solucionar todos los problemas que surgen con la versión anterior, y además ofrecer soporte a las nuevas redes de alto rendimiento (como ATM, Gigabit Ethernet, etc.)

Una de las características más llamativas es el nuevo sistema de direcciones, en el cual se pasa de los 32 a los 128 bit, eliminando todas las restricciones del sistema actual. Otro de los aspectos mejorados es la seguridad, que en la versión anterior constituía uno de los mayores problemas. Además, el nuevo formato de la cabecera se ha organizado de una manera más efectiva, permitiendo que las opciones se sitúen en extensiones separadas de la cabecera principal.

1.1.8.1. Formato de la cabecera.

El tamaño de la cabecera que el protocolo IPv6 añade a los datos es de 320 bit, el doble que en la versión antigua. Sin embargo, esta nueva cabecera se ha simplificado con respecto a la anterior. Algunos campos se han retirado de la misma, mientras que otros se han convertido en opcionales por medio de las extensiones. De esta manera

los *routers* no tienen que procesar parte de la información de la cabecera, lo que permite aumentar de rendimiento en la transmisión. El formato completo de la cabecera sin las extensiones es el siguiente:

- **Versión:** Número de versión del protocolo IP, que en este caso contendrá el valor 6.
  Tamaño: 4 bit.

- **Prioridad:** Contiene el valor de la prioridad o importancia del paquete que se está enviando con respecto a otros paquetes provenientes de la misma fuente.
  Tamaño: 4 bit
- **Etiqueta de flujo:** Campo que se utiliza para indicar que el paquete requiere un tratamiento especial por parte de los *routers* que lo soporten.
  Tamaño: 24 bit.

- **Longitud:** Es la longitud en bytes de los datos que se encuentran a continuación de la cabecera.
  Tamaño: 16 bit.

- **Siguiente cabecera:** Se utiliza para indicar el protocolo al que corresponde la cabecera que se sitúa a continuación de la actual. El valor de este campo es el mismo que el de protocolo en la versión 4 de IP.
  Tamaño: 8 bit.

- **Límite de existencia:** Tiene el mismo propósito que el campo de la versión 4, y es un valor que disminuye en una unidad cada vez que el paquete pasa por un nodo.
  Tamaño: 8 bit.

- **Dirección de origen:** El número de dirección del *host* que envía el paquete.
  Su longitud es cuatro veces mayor que en la versión 4.
  Tamaño: 128 bit.

- **Dirección de destino:** Número de dirección de destino, aunque puede no coincidir con la dirección del *host* final en algunos casos. Su longitud es cuatro veces mayor que en la versión 4 del protocolo IP.

Tamaño: 128 bit.

<table>
<tr><td>Versión
(4 bits)</td><td>Prioridad
(4 bits)</td><td colspan="3">Etiqueta de flujo
(24 bits)</td></tr>
<tr><td colspan="3">Longitud carga (16 bits)</td><td>Siguiente
Cabecera (4 bits)</td><td>Límite de saltos
(8 bits)</td></tr>
<tr><td colspan="5">Dirección origen
(128 bits)</td></tr>
<tr><td colspan="5">Dirección destino
(128 bits)</td></tr>
</table>

FIGURA 1.1.14 Formato cabecera

Las extensiones que permite añadir esta versión del protocolo se sitúan inmediatamente después de la cabecera normal, y antes de la cabecera que incluye el protocolo de nivel de transporte. Los datos situados en cabeceras opcionales se procesan sólo cuando el mensaje llega a su destino final, lo que supone una mejora en el rendimiento.

Otra ventaja adicional es que el tamaño de la cabecera no está limitado a un valor fijo de bytes como ocurría en la versión 4.

Por razones de eficiencia, las extensiones de la cabecera siempre tienen un tamaño múltiplo de 8 bytes.

Actualmente se encuentran definidas extensiones para *routing* extendido, fragmentación y ensamblaje, seguridad, confidencialidad de datos, etc.

1.1.8.2. Direcciones en la versión 6.

El sistema de direcciones es uno de los cambios más importantes que afectan a la versión 6 del protocolo IP, donde se han pasado de los 32 a los 128 bit (cuatro veces mayor). Estas nuevas direcciones identifican a un interfaz o conjunto de interfaces y no a un nodo, aunque como cada interfaz pertenece a un nodo, es posible referirse a éstos a través de su interfaz.

El número de direcciones diferentes que pueden utilizarse con 128 bits es enorme. Teóricamente serían 2^{128} direcciones posibles, siempre que no apliquemos algún formato u organización a estas direcciones. Este número es extremadamente alto, pudiendo llegar a soportar más de 665.000 **trillones** de direcciones distintas por cada

metro cuadrado de la superficie del planeta Tierra. Según diversas fuentes consultadas, estos números una vez organizados de forma práctica y jerárquica quedarían reducidos en el peor de los casos a 1.564 direcciones por cada metro cuadrado, y siendo optimistas se podrían alcanzar entre los tres y cuatro trillones.

Existen tres tipos básicos de direcciones IPng según se utilicen para identificar a un interfaz en concreto o a un grupo de interfaces. Los bits de mayor peso de los que componen la dirección IPng son los que permiten distinguir el tipo de dirección, empleándose un número variable de bits para cada caso. Estos tres tipos de direcciones son:

- **Direcciones *unicast:*** Son las direcciones dirigidas a un único interfaz de la red. Las direcciones *unicast* que se encuentran definidas actualmente están divididas en varios grupos. Dentro de este tipo de direcciones se encuentra también un formato especial que facilita la compatibilidad con las direcciones de la versión 4 del protocolo IP.

- **Direcciones *anycast:*** Identifican a un conjunto de interfaces de la red. El paquete se enviará a un interfaz cualquiera de las que forman parte del conjunto. Estas direcciones son en realidad direcciones *unicast* que se encuentran asignadas a varios interfaces, los cuales necesitan ser configurados de que el de las direcciones *unicast*.

- **Direcciones *multicast:*** Este tipo de direcciones identifica a un conjunto de interfaces de la red, de manera que el paquete es enviado a cada una de ellos individualmente.

Las direcciones de *broadcast* no están implementadas en esta versión del protocolo, debido a que esta misma función puede realizarse ahora mediante el uso de las direcciones *multicast*.

1.2. ADMINISTRACION DE REDES Y MONITOREO

La administración de Redes es un conjunto de técnicas tendientes a mantener una red operativa, eficiente, segura, constantemente monitoreada y con una planeación adecuada y propiamente documentada.

Sus objetivos son:

- Mejorar la continuidad en la operación de la red con mecanismos adecuados de control y monitoreo, de resolución de problemas y de suministro de recursos.

- Hacer uso eficiente de la red y utilizar mejor los recursos, como por ejemplo, el ancho de banda.

- Reducir costos por medio del control de gastos y de mejores mecanismos de cobro.
- Hacer la red más segura, protegiéndola contra el acceso no autorizado, haciendo imposible que personas ajenas puedan entender la información que circula en ella.

- Controlar cambios y actualizaciones en la red de modo que ocasionen menos interrupciones posibles, en el servicio a los usuarios

La administración de la red se vuelve más importante y difícil si se considera que las redes actuales comprenden lo siguiente:

- Mezclas de diversas señales, como voz, datos, imagen y graficas.

- Interconexión de varios tipos de redes, como WAN, LAN y MAN.

WAN: Es una red punto a punto, es decir, red de paquete conmutado. Las redes WAN pueden usar sistemas de comunicación vía satélite o de radio. Fue la aparición de los portátiles y los PDA la que trajo el concepto de redes inalámbricas.
LAN: Es la interconexión de varios ordenadores y periféricos. Su extensión está limitada físicamente a un edificio o a un entorno de 200 metros o con repetidores podríamos llegar a la distancia de un campo de 1 kilómetro.

MAN: Las Redes MAN BUCLE, se basan en tecnologías Bonding, de forma que los enlaces están formados por múltiples pares de cobre con el fin de ofrecer el ancho de banda necesario.

- El uso de multiplex medios de comunicación, como par trenzado, cable coaxial, fibra óptica, satélite infrarrojo y microondas.

Par trenzado: Es una forma de conexión en la que dos aisladores son entrelazados para darle mayor estética al terminado del cable y aumentar la potencia.

Cable Coaxial: Cable usado por las redes de cómputo al igual que en la televisión por cable. El nombre se debe a su estructura: un blindaje metálico que rodea a un alambre central. El blindaje protege la señal del alambre interior contra interferencias eléctricas.

Fibra Óptica: Los circuitos de fibra óptica son filamentos de vidrio (compuestos de cristales naturales) o plástico (cristales artificiales). Llevan mensajes en forma de haces de luz que realmente pasan a través de ellos de un extremo a otro, donde quiera que el filamento vaya (incluyendo curvas y esquinas) sin interrupción.

Satélite: Son medios de transmisión localizados en órbita alrededor de la tierra. En este tipo de redes los enrutadores tienen una antena por medio de la cual pueden enviar y recibir.

Laser: El laser es un rayo de luz proveniente de un "cañón" que lo genera a partir de un proceso opto-físico. Tiene además una apertura de haz muy pequeña que hace que se lo pueda utilizar como un "pincel" a la distancia. Moviéndolo convenientemente mediante espejos comandados por computadoras, se consiguen crear efectos tales como túneles, techos, olas y proyección de historias gráficas **(*)**.

Infrarrojo: (radiación térmica), tipo de radiación electromagnética de mayor longitud de onda que la luz visible pero menos que las microondas.

Los rayos infrarrojos son utilizados para visión nocturna, comando a distancia, comunicación a corta distancia entre periféricos y las computadoras.

Microondas: Ondas electromagnéticas en el intervalo de 1 a 30 GHz. Las redes basadas en microondas poseen una tecnología evolutiva cada vez más utilizada debido al elevado ancho de banda y los costos relativamente bajos.

- Diversos protocolos de comunicación, incluyendo TCP/IP, SPX/IPX, SNA, OSI.

  SNA: (System Network Architecture): Es un conjunto de protocolos para interconectar computadoras y sus recursos.

- El empleo de muchos sistemas operativos, como DOS, NetWare, Windows NT, UNIX, OS/2.

  DOS: (Disk Operating System - Sistema Operativo de Disco) Familia de sistemas operativos utilizados en PCs.
  NetWare: (Novell NetWare): Sistema operativo de red desarrollado por la empresa Novell. Es considerada una de las plataformas más fiables que ofrecen acceso seguro y continuo a la red y los recursos.

  Windows NT: Línea de sistemas operativos tipo Windows desarrollado por Microsoft dedicados originalmente a estaciones de trabajo y servidores de red, luego orientados al hogar y profesionales desde su versión de Windows XP en adelante.

  UNIX: Sistema operativo multiplataforma, multitarea y multiusuario desarrollado originalmente Bell de AT&T.

  OS/2: (Operating System 2) sistema operativo no libre desarrollado por IBM que intentó suceder a DOS es multitarea con un entorno grafico de 32 bits.

- Diversas arquitecturas de red, incluyendo Ethernet 10 base T, Fast Ethernet, Token Ring, FDDI, 100vg-AnyLAN y Fiber channel.

  Ethernet 10 base T: También conocido como Twisted Pair Ethernet, es una tecnología que se apoya en un tipo de trenzado similar al cable telefónico, conector que es conocido como RJ45 y que permite una **longitud máxima de 100 metros** para un trabajo eficiente.

  Fast Ethernet: Este ofrece un aumento de alta velocidad mayor que el de la especificación 10BaseT de Ethernet.

Token Ring: La red Token Ring consta de un conjunto de nodos conectados en forma de anillo.

FDDI: (Interfaz de Datos Distribuida por Fibra) es un conjunto de estándares ISO y ANSI para la transmisión de datos en redes de computadoras de área extendida o local (LAN) mediante cable de fibra óptica.

100vg-AnyLan: Tecnología de medios Fast Ethernet y Token Ring de 100 Mbps que utiliza cuatro pares de cableado UTP de Categoría 3, 4 ó 5. Esta tecnología de transporte de alta velocidad, desarrollada por Hewlett-Packard, puede operar en redes Ethernet 10BaseT existentes.

Fiber Channel: (Canal de Fibra). Tecnología de red a velocidad de gigabit, principalmente utilizada para redes de almacenamiento especialmente empresarial.

- Varios métodos de comprensión, códigos de línea, etc.…

El sistema de administración de red opera bajo los siguientes pasos básicos:

1. Colección de información acerca del estado de la red y componentes del sistema. La información recolectada de los recursos debe incluir: eventos, atributos y acciones operativas.

2. Transformación de la información para presentarla en formatos apropiados para el entendimiento del administrador.
3. Transportación de la información del equipo monitoreado al centro de control.

4. Almacenamiento de los datos coleccionados en el centro de control.

5. Análisis de parámetros para obtener conclusiones que permitan deducir rápidamente lo que pasa en la red.

6. Actuación para generar acciones rápidas y automáticas en respuesta a una falla mayor.

La característica fundamental de un sistema de administración de red moderno es la de ser un sistema abierto, capaz de manejar varios protocolos y lidiar con varias arquitecturas de red. Esto quiere decir soporte para los protocolos de red más importantes.

Mediante este proceso los administradores de red pueden detectar y corregir problemas, observar y analizar el estado y comportamiento de red.
La administración de redes definida por ISO, describe una arquitectura de Administración cuya función es permitir supervisar (monitoreo) y controlar una red de datos. Está dividida en cinco categorías denominadas **Áreas Funcionales Específicas de Administración** Estas categorías son las siguientes:

Administración de fallos: Detecta, diagnóstica y corrige los fallos de la red y de las condiciones de error. Incluye:

1. Notificación de fallos
2. Sondeo periódico en busca de mensajes de error
3. Establecimiento de alarmas
4. Corrección de fallos (posiblemente en forma automática.

Administración del Desempeño (Prestaciones): Se define como la evaluación del comportamiento de los elementos de la red. La gerencia del desempeño permite:

1. Obtener tasas de utilización y error de los dispositivos de la red
2. Proporcionar un nivel constante de desempeño, asegurando que los dispositivos tengan suficiente capacidad.

Administración de contabilidad: Determina los costos asociados a la utilización de los recursos y administra y regula dichos costos manteniendo un nivel de desempeño aceptable.

Administración de seguridad: Controla los accesos no autorizados y daños tanto a los recursos de la red como a la información manejada (datos privados). La protección de la red incluye aspectos como la gestión de claves, cortafuegos e históricos de seguridad.

Administración de Configuración: Se refiere al monitoreo o seguimiento de los cambios en el tiempo con respecto a la configuración inicial de los dispositivos de la red y comprende una serie de facilidades mediante las cuales se realizan las siguientes funciones:

1. Iniciación y desactivación.
2. Definición o cambio de parámetros de configuración.
3. Recogida de información de estado.

Herramientas Open Source para Administrar la Red

Con las herramientas libres se puede administrar en la red sin perder ninguna funcionalidad que haciéndolo con herramientas pagadas como bm Tivoli, Netview, OpenView, NetFlow, CiscoWorks. Estas herramientas libres para monitorear son:

- **Nagios:** Sistema de monitorización de equipos y servicios, diseñado para informarte de los problemas de tu red. Diseñado para ejecutarse en Sistemas GNU/Linux. El demonio de monitorización realiza chequeos intermitentes en los equipos y en los servicios que especificados usando "plugins" externos los cuales devuelven información a Nagios.
- **MRTG:** Herramienta para monitorizar la carga del tráfico o el estado de la red, existen multitud de servicios como squid, apache, snort que tienen plantillas predefinidas para monitorizar parámetros específicos de cada servicio con MRTG. MRTG genera páginas HTML que contienen imágenes PNG que nos proporcionan una información visual VIVA del tráfico de nuestros dispositivos.

- **Snort:** Sistema de detección de intrusos capaz de generar análisis del tráfico en tiempo real y generar logs de paquetes en redes IP. Puede realizar análisis de protocolos, búsqueda de patrones establecidos y se puede usar para detectar gran variedad de ataques e intentos, tales como buffer overflows, escaneos de puertos silenciosos, ataques CGI, escaneos SMB, intentos de huellas de OS (fingerprinting) y muchos más.
- **Ethereal:** Es un analizador de protocolos de red libre para Unix y Windows. Permite examinar datos de una red en vivo, o capturarlos en disco. Puedes buscar interactivamente entre los datos capturados, ver sumarios y detalles de

información de cada paquete. Ethereal tiene muchas características poderosas que incluyen un completo lenguaje de filtros y la habilidad de ver el flujo reconstruido de una sesión TCP. Esta herramienta ha sido clave en el desarrollo de samba.

- **Password Safe:** Gestiona todas las passwords de tus dispositivos. Password Safe nos permitirá tener todas las contraseñas de todos los programas, equipos y webs que utilicemos normalmente, sin tener que recordar.

Administración de una Red

El administrador de red es la persona responsable de supervisar y controlar el hardware y software de una red. El administrador trabaja en la detección y corrección de problemas que hacen ineficiente o imposible la comunicación y en la eliminación de las condiciones que pudieran llegar a provocar el problema nuevamente. Ya que tanto las fallas de hardware como de software pueden generar problemas, el administrador de red debe supervisar ambos.

Para este proceso de Administración de red se han desarrollado el protocolo SNMP el cuales facilitara mucho el trabajo del administrador, pues constituyen una herramienta de información sumamente útil, razón por la cual en este capítulo se realizara el estudio del mismo.

1.2.1. Protocolo Simple de Administración de Red (SNMP)

SNMP es un protocolo del nivel de aplicación basado en paquetes UDP, define una relación cliente/servidor entre el gestor de red (que actúa de cliente) y los elementos gestionados (que son los servidores y reciben el nombre de "agentes").

UDP (User Datagram Protocol): Cuenta el número de datagramas UDP, enviados, recibidos y entregados.

Utiliza UDP como un protocolo de transporte de mensajes, trabaja con el puerto 161 el cual se utiliza para todos los mensajes, excepto para los traps que son escuchados en el puerto 162 de UDP.

Por lo que consta de 3 elementos: los "Agents", el "Manager" y los MIB (Management Information Base). Los Agents son programas que son instalados en Routers, Hubs, Estaciones de Trabajo, etc., que permiten una interfaz entre el protocolo SNMP y la configuración local del equipo. El "Manager" es el programa central que consulta (o configura) las variables de cada nodo, interactuando con el "Agent" respectivo. El MIB es el "árbol" de especificaciones de la red, donde la raíz del árbol contiene las variables más globales de lo que sucede en la red y las "hojas" del árbol corresponden a la información detallada de cada nodo en la red (los Agents).

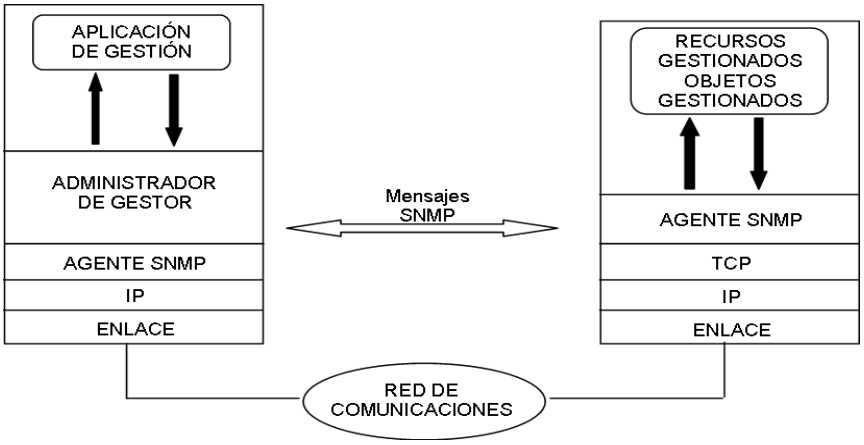


FIGURA 1.2.1 Funcionamiento de SNMP

“Para el protocolo SNMP la red constituye un conjunto de elementos básicos Administradores y Gestores. Esta información es relativa a los elementos gestionados por lo que se conoce como polling (el gestor solicita al agente los valores de ciertos parámetros, y el agente responde con la información contenida en su MIB) de manera secuencial, apoyándose en los parámetros contenidos en sus MIBs.

Los cinco tipos de mensajes SNMP intercambiados entre los Agentes y los Administradores, se indican en la siguiente figura:

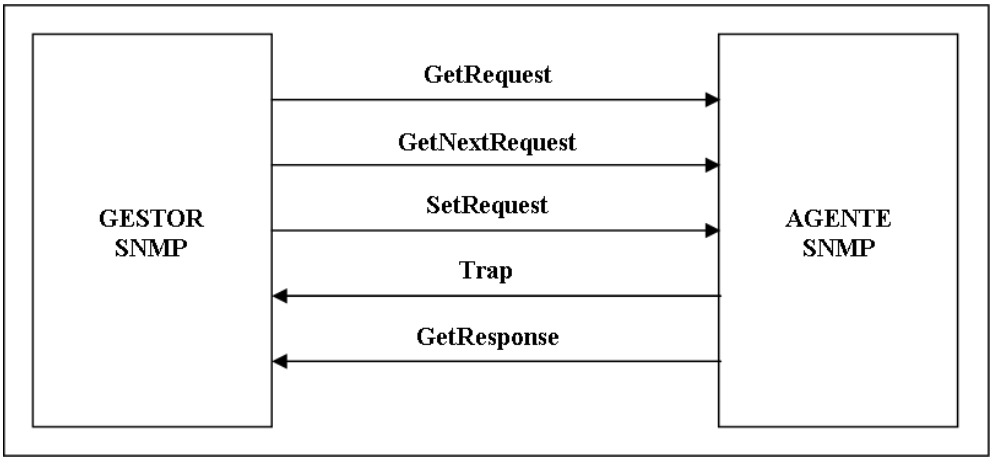


FIGURA 1.2.2 Tipos de Mensajes

En donde:

- **Get Request**: Una petición del Administrador al Agente para que envíe los valores contenidos en el MIB (base de datos).
- **Get Next Request**: Una petición del Administrador al Agente para que envíe los valores contenidos en el MIB referente al objeto siguiente al especificado anteriormente.
- **Get Response**: La respuesta del Agente a la petición de información lanzada por el Administrador.
- **Set Request**: Una petición del Administrador al Agente para que cambie el valor contenido en el MIB referente a un determinado objeto.
- **Trap**: Un mensaje espontáneo enviado por el Agente al Administrador, al detectar una condición predeterminada, como es la conexión/desconexión de una estación o una alarma."[7]

"SNMP reúne todas las operaciones en el paradigma obtener-almacenar (fech store paradigm). Conceptualmente, SNMP contiene solo dos comandos que permiten a un administrador buscar y obtener un valor desde un elemento de datos o almacenar un valor en un elemento de datos. Todas las otras operaciones se definen como consecuencia de las operaciones básicas.

[7] J.Manuel Huidrovo, Snmp. Un Protocolo Simple de Gestión [http://www.coit.es/publbit/bit102/quees.htm]
Formato de las Tramas SNMP [http://www.timat.unican.es/siteadmin/submaterials/89.pdf]

La mayor ventaja de usar paradigma obtener-almacenar es la estabilidad, simplicidad. El SNMP es, especialmente, estable ya que sus definiciones se mantienen fijas aún, cuando nuevos elementos de datos se añadan a la MIB y se definan nuevas operaciones como efectos del almacenamiento de esos elementos.

Por otro lado, el formato de los paquetes SNMP consta de dos campos que especifican el número de versión de SNMP y un nombre de comunidad; el resto del paquete depende del tipo del mismo, y se denomina PDU (Protocol Data Unit) de SNMP."[7]

Versión	Comunidad	PDU de SNMP

Mensaje de SNMP

Tipo PDU	Request ID	0	0	Asignaciones de Variables

PDUs de GetRequest, GetNextRequest y SetRequest

Tipo PDU	Request ID	Cód. error	índice error	Asignaciones de Variables

PDUs de GetResponse

Nombre 1	Valor 1	Nombre 2	Valor 2	...	Nombre N	Valor N

Asignación de variables

En donde:

CAMPO	SIGNFICADO
Request ID:	Identificador único por cada petición.
Código de Error:	Indica que ha ocurrido una excepción al procesar una

	Petición; posibles valores: noError (0), tooBig (1), noSuchName(2), badValue(3), readOnly(4), genErr(5).
Índice de error:	Indica que la variable de la lista causo la excepción, cuando el código de error no es 0.
Asignación de Variables:	Lista de nombres de variables y sus correspondientes valores; en GetRequest su valor es null.
Empresa:	Nombre del objeto que genera el trap (valor de sysObjectID).
Dirección de agente:	Dirección IP del agente que genera el trap.
Trap genérica	Tipo de trap genérico.
Trap especifico:	Código de trap especifico.
Time stamp:	Tiempo transcurrido entre la última reinicialización de la entidad y la generación del trap (valor de sysUpTime).

Tabla# 1.2 Significado del PDU de SNMP

1.2.2. MODELO DE COMUNICACIÓN

1.2.2.1. Modelo de Arquitectura

Como se mencionó brevemente la arquitectura de SNMP está compuesta por la estación de administración, agente de administración, MIB y por el protocolo de administración de red. A continuación se detalla cada uno de los elementos, los mismos que se encuentran representados en la Fig. 1.2.2

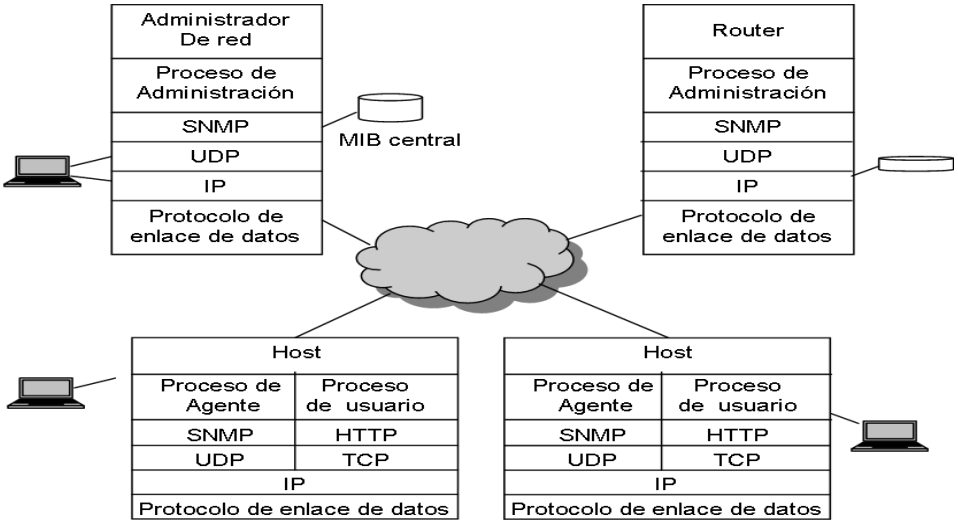


FIGURA 1.2.2 Arquitectura de SNMP

a. Estación de administración:

- La estación de administración es la interfaz del administrador de red al sistema de red. Posee los programas para manipular los datos y controlar la red.
- La estación de administración también mantiene una base de datos de información de administración (MIB) extraída de los dispositivos bajo su administración.

b. Agente de administración:

El agente de administración es el componente incluido en los dispositivos que se deben administrar. Puentes, routers, hubs y switches pueden contener agentes SNMP que les permitan ser controlados por la estación de administración. El agente de administración responde a la estación de administración de dos maneras. En primer lugar, mediante sondeo, la estación de administración requiere datos desde el agente y el agente responde con los datos solicitados. Trapping es un método de recopilación de datos diseñado para reducir el tráfico en la red y el procesamiento en los dispositivos que se controlan. En lugar de que la estación de administración haga un sondeo a los agentes a intervalos específicos, se establecen umbrales (límites superiores o inferiores) en el dispositivo administrado. Si se supera este umbral en el dispositivo, el dispositivo administrado envía un mensaje de alerta a la estación de administración. Esto elimina la necesidad de realizar sondeos continuos de todos los dispositivos administrados en la red. El Trapping es muy ventajoso en las redes que incluyen una gran cantidad de dispositivos que necesitan administrarse. Reduce la cantidad de tráfico SNMP en la red para proporcionar mayor ancho de banda para la transferencia de datos.

c. Base de información de administración:

La base de información de administración tiene una estructura de base de datos y reside en cada dispositivo administrado. La base de datos contiene una serie de objetos, que son datos sobre recursos reunidos en el dispositivo administrado. Algunas de las categorías en el MIB incluyen datos de interfaz de puerto, datos de TCP y datos de ICMP.

d. Protocolo de administración de red:

SNMP es un protocolo de capa de aplicación diseñado para comunicar datos entre la consola de administración y el agente de administración. Tiene tres capacidades clave. La capacidad para OBTENER, que implica que la consola de administración recupera datos del agente, COLOCAR, que implica que la consola de administración establece los valores de los objetos en el agente, y TRAP, que implica que el agente notifica a la consola de administración acerca de los sucesos de importancia.

1.2.3. FUNCIONAMIENTO CLIENTE-SERVIDOR

Como hemos mencionado anteriormente, SNMP es un protocolo desarrollado para la gestión de redes con protocolos TCP/IP.

El Modelo de Comunicación es similar al de la gestión OSI, existe un proceso Gestor (servidor) y un proceso Agente (cliente) que intercambian información de gestión mediante el protocolo SNMP. La Fig. 1.2.3: Funciones que desempeñan

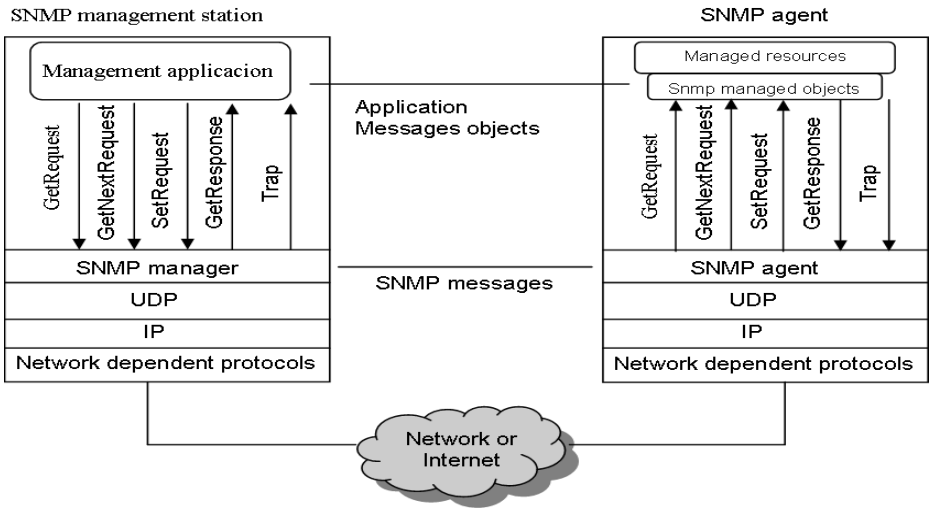


FIGURA 1.2.3 Funcionamiento Cliente- Servidor SNMP

Aquí se puede observar que la estación de gestión es un dispositivo que sirve como interface entre la persona encargada de la administración de la red y el sistema de gestión de red. Esta estación debe tener como mínimo:

- Un conjunto de aplicaciones de gestión para análisis de datos, recuperación de fallas, detección de alarmas estadísticas, etc.

- Una interface a través de la cual la persona encargada de la administración de la red pueda monitorear y controlar la red.
- La capacidad de trasladar los requerimientos del administrador a los dispositivos remotos que conforman la red.
- Una base de datos de Información de gestión de la red extraída a partir de las bases de datos de todas las entidades gestionadas en la red.

Mientras que el agente compuesto por hubs, routers, bridges o hosts puede estar equipado con un agente de software el cual puede ser manejado desde una estación de gestión. El agente responde a solicitudes de información y de acción que provienen de la estación de gestión, y puede proveer asincrónicamente información importante a la estación de gestión que no ha sido solicitada (alarmas).

Existe además un tipo de agente especial encargado de mediar entre el sistema a administrar y el sistema administrativo, este agente es conocido como un agente Proxy, el mismo que es necesario utilizar cuando existen dispositivos como los modem que no soporta toda la suite del protocolo TCP/IP. En la figura 1.2.4:

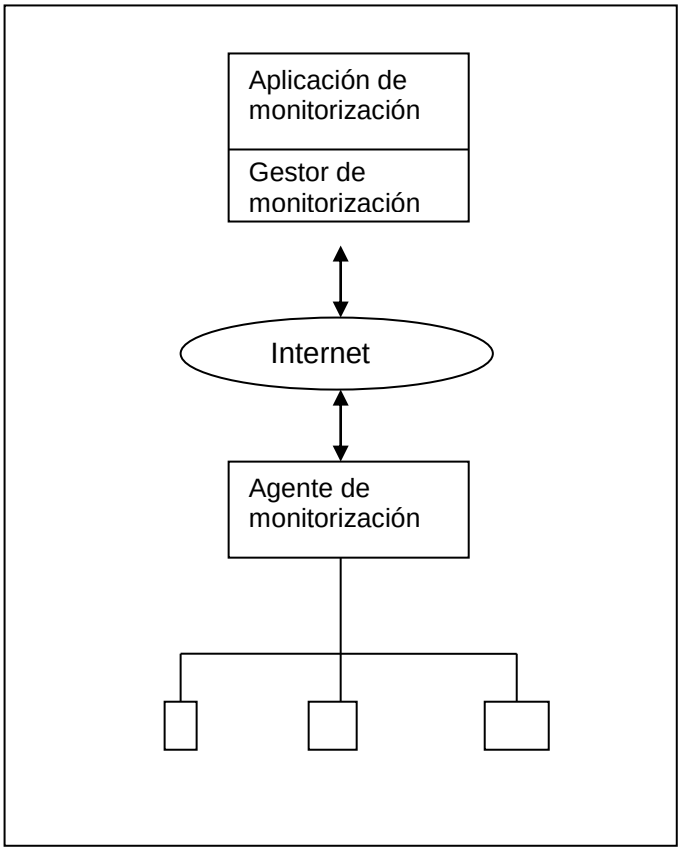


FIGURA 1.2.4 Agente Proxy

1.2.4 Base de Información de Administración (MIB)

“SNMP define un estándar separado para los datos gestionados por el protocolo. Este estándar define los datos mantenidos por un dispositivo de red, así como las operaciones que están permitidas. Los datos están estructurados en forma de árbol; en el que sólo hay un camino desde la raíz hasta cada variable. Esta estructura en árbol se llama Management Information Base (MIB) y se puede encontrar información sobre ella en varios RFC's.

La versión actual de TCP/IP MIB es la 2 (MIB-II) y se encuentra definida en el RFC-1213. En ella se divide la información que un dispositivo debe mantener en ocho categorías. Cualquier variable ha de estar en una de estas categorías (ver tabla 1.3).

Tabla# 1.3 Categorías TCP/IP

Categoría	**Información**
system	Información del host del sistema de encaminamiento
interfaces	Información de los interfaces de red
addr-translation	Información de traducción de direcciones
ip	Información sobre el protocolo IP
icmp	Información sobre el protocolo ICMP
tcp	Información sobre el protocolo TCP
udp	Información sobre el protocolo UDP
egp	Información sobre el protocolo (Exterior Gateway)

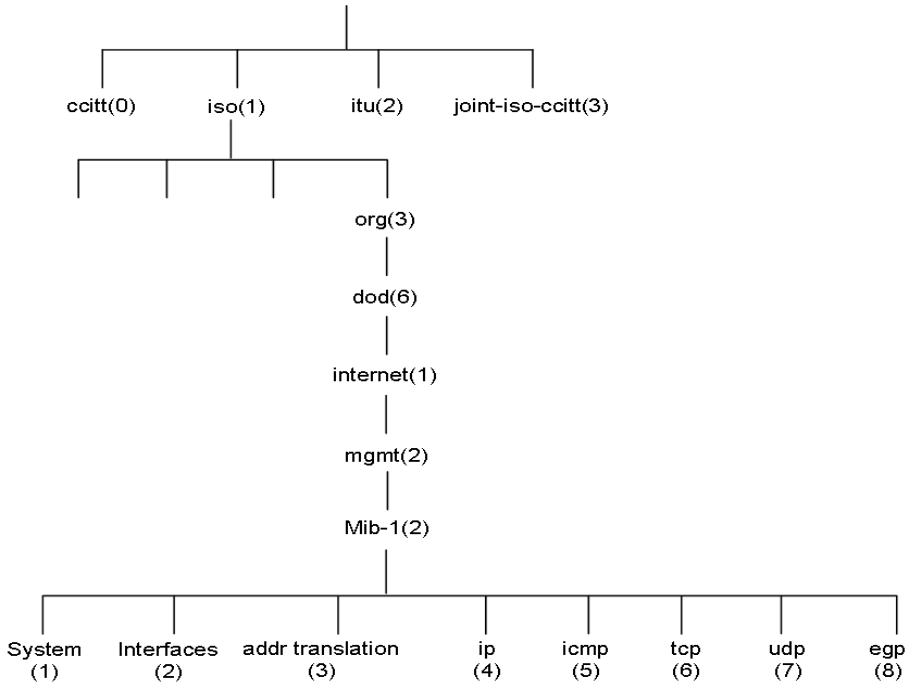
La definición de un elemento concreto MIB implica la especificación del tipo de dato que puede contener. Normalmente, los elementos de un MIB son enteros, pero también pueden almacenar cadenas de caracteres o estructuras más complejas como tablas. A los elementos de un MIB se les llama "objetos". Los objetos son los nodos hoja del árbol MIB, si bien, un objeto puede tener más de una instancia, como por ejemplo un objeto tabla. Para referirse al valor contenido en un objeto, se ha de añadir el número de la instancia. Cuando sólo exista una instancia del objeto, está es la instancia cero.

Existe otro estándar que define e identifica las variables MIB, llamado "Structure of Management Information" (SMI). SMI especifica las variables MIB, éstas se declaran empleando un lenguaje formal ISO llamado ASN.1, que hace que tanto la forma como los contenidos de estas variables sean no ambiguos.

El espacio de nombres ISO (árbol) está situado dentro de un espacio de nombres junto con otros árboles de otros estándares de otras organizaciones. Dentro del espacio de nombres ISO hay una rama específica para la información MIB. Dentro de esta rama MIB, los objetos están a su vez jerarquizados en suárboles para los distintos protocolos y aplicaciones, de forma que esta información puede representarse unívocamente.

La Figura 1.2.5 muestra el espacio de nombres del MIB del TCP/IP, éste está situado justo bajo el espacio del IAB "mgmt". La jerarquía también específica el número para cada nivel."[8]

FIGURA 1.2.5 Árbol de la MIB en TCP/IP



Aquí algunos ejemplos de los objetos de cada grupo.

- Grupo de sistema (1.3.6.1.2.1.1)
  - sysDescr - Descripción completa del sistema(versión, HW, OS)
  - sysObjectID - Identificación que da el distribuidor al objeto
  - sysUpTime - Tiempo desde la última reinicialización
  - sysContact - Nombre de la persona que hace de contacto
  - sysServices - Servicios que ofrece el dispositivo
- Grupo de interfaces (1.3.6.1.2.1.2)
  - ifIndex - Número de interfaz
  - ifDescr - Descripción de la interfaz
  - ifType - Tipo de la interfaz
  - ifMtu - Tamaño máximo del datagrama IP
  - ifAdminisStatus - Status de la interfaz
  - ifLastChange - Tiempo que lleva la interfaz en el estado actual

[8] http://www.david-guerrero.com/papers/snmp/lj.es.html

  - ifINErrors - Número de paquetes recibidos que contenían errores
  - ifOutDiscards - Número de paquetes enviados y desechados
- Grupo de traducción de direcciones (1.3.6.1.2.1.3)
  - atTable - Tabla de traducción de direcciones
  - atEntry - Cada entrada que contiene una correspondencia de dirección de red a dirección física
  - atPhysAddress - La dirección física dependiente del medio
  - atNetAddress - La dirección de red correspondiente a la dirección física
- Grupo IP (1.3.6.1.2.1.4)
  - ipForwarding - Indicación de si la entidad es una pasarela IP
  - ipInHdrErrors - Número de datagramas de entrada desechados debido a errores en sus cabeceras IP
  - ipInAddrErrors - Número de datagramas de entrada desechados debido a errores en sus direcciones IP
  - ipInUnknownProtos - Número de datagramas de entrada desechados debido a protocolos desconocidos o no soportados
  - ipReasmOKs - Número de datagramas IP re ensamblados con éxito
  - ipRouteMask - Máscara de subred para el encaminamiento
- Grupo ICMP (1.3.6.1.2.1.5)
  - icmpInMsgs - Número de mensajes ICMP recibidos
  - icmpInDestUnreachs - Número de mensajes ICMP "destino inalcanzable"(destination unreachable) recibidos
  - icmpInTimeExcds - Número de mensajes ICMP "time exceeded"(tiempo excedido) recibidos
  - icmpInSrcQuenchs - Número de mensajes ICMP "Source quench(desbordamiento del emisor) recibidos
  - icmpOutErrors - Número de mensajes ICMP no enviados debido a problemas en ICMP
- Grupo TCP (1.3.6.1.2.1.6)
  - tcpRtoAlgorithm - Algoritmo que determina el time out para retransmitir octetos para los que no se ha recibido reconocimiento
  - tcpMaxConn - Límite en el número de conexiones TCP que puede soportar la entidad
  - tcpActiveOpens - Número de veces que las conexiones TCP han efectuado una transición directa del estado SYN-SENT al estado CLOSED

- tcpInSegs - Número de segmentos recibidos, incluyendo aquellos con error
- tcpConnRemAddress - La dirección IP remota para esta conexión TCP
- tcpInErrs - Número de segmentos desechados debido a errores de formato
- tcpOutRsts - Número de resets generados

- Grupo UDP (1.3.6.1.2.1.7)
  - udpInDatagrams - Número de datagramas UDP entregados a usuarios UDP
  - udpNoPorts - Número de datagramas UDP recibidos para los que no existía aplicación en el puerto de destino
  - udpInErrors - Número de datagramas UDP recibidos que no se pudieron entregar por razones otras que la ausencia de la aplicación en el puerto de destino
  - udpOutDatagrams - Número de datagramas UDP enviados por la entidad
- Grupo EGP (1.3.6.1.2.1.8)
  - egpInMsgs - Número de mensajes EGP recibidos sin error
  - egpInErrors - Número de mensajes EGP con error
  - egpOutMsgs - Número de mensajes EGP generados localmente
  - egpNeighAddr - La dirección IP del vecino de esta entrada EGP
  - egpNeighState - El estado EGP del sistema local con respecto a la entrada EGP vecino

1.2.5. DESARROLLO DEL SNMP: VERSIONES

A lo largo del uso de SNMP han aparecido varias desventajas como: problemas para transferir grandes cantidades de información, poca ó ninguna seguridad, así como los débiles mecanismos de autenticación y privacidad.

Como realmente las capacidades de SNMP para el manejo básico de una red son buenas, se decidió en 1993 introducir una nueva versión SNMPv2 la cual fue revisada en 1996. SNMPv2 estaba orientado a corregir las capacidades de transmisión de grandes cantidades de información, sin embargo esta versión seguía sin ofrecer solución alguna en cuanto a seguridad y privacidad se refiere. Para corregir este tipo de deficiencias, de tanta importancia en Enero de 1998 ya se había producido una serie de estándares que fueron propuestos en las publicaciones de los RFC's 2271-

2275[1], y cuyo resultado es SNMPv3. En estos documentos se definen las especificaciones de seguridad y control de acceso de las redes manejadas o gestionadas con SNMP, y que por supuesto incluyen las funciónabilidades de las versiones SNMPv1 y SNMPv2.

1.2.5.1 Administración y seguridad de SNMPv1

La seguridad en SNMP se basa en el concepto de comunidad (comunity): Una comunidad es una relación entre un agente SNMP y un conjunto de estaciones de gestión SNMP, que define unas características de autenticación y control de acceso.

- El agente establece una comunidad para combinación de autenticación y control de acceso, y a cada comunidad se la da un nombre único dentro del agente (comunity name).
- Las estaciones de administración pertenecientes a una comunidad deben emplear ese nombre en todas las operaciones get y set.
- El agente puede establecer cualquier número de comunidades.
- Una estación de administración puede pertenecer a varias comunidades.
- Una estación debe almacenar los nombres de comunidad asociados a cada Agente.
- Mediante el uso de comunidades un agente puede limitar el acceso a su MIB en dos formas:
    - Vista de la MIB SNMP: Subconjunto de los objetos de la MIB.
    - Modo de acceso: READ-ONLY o READ-WRITE.

- La combinación de una vista de la MIB y un modo de acceso se denomina perfil de comunidad SNMP (SNMP, comunity profile).
- A cada comunidad se le asigna un perfil denominándose a esta estación de política de acceso SNMP (SNMP, Access policy).
- Cada paquete snmp contiene el nombre de la comunidad sin codificar.
- El agente solo atiende la petición si el nombre de la comunidad es correcto para el tipo de acceso solicitado.
- Se trata de un esquema de seguridad muy limitado.

1.2.5.2 SNMPv2

La infraestructura de la versión 2 de SNMP consta de las siguientes disciplinas:

- SMI ("Structure of Management Information"): Definición del subconjunto de ASNN.1 para la creación de módulos MIB.

- Convenios textuales: Definición del conjunto inicial de convenios textuales disponible para todos los módulos MIB.

- Operaciones del protocolo: Definición de las operaciones del protocolo con respecto a las PDUs enviadas y recibidas.

- Mapeados de transporte: Definición del mapeado de SNMPv2 sobre un conjunto inicial de dominios de transporte ya que se puede utilizar en diferentes pilas de protocolo. El mapeado en UDP es el preferido. El RFC también define OSI, AppleTalk, IPX, etc.

- Instrumentación del protocolo: Definición del MIB y del MIB Manager-Manager.

- Infraestructura administrativa: Definición de SNMPv2 Party, SP ("Security Protocols") y Party MIB. Descripción en los RFC's 1445, 1446 y 1447.

- Compatibilidades: Definición de la *compatibilidad* o *capacidad* de notación de los agentes. Descripción en el RFC 1444.

a. Entidad SNMPv2

"Una entidad SNMPv2 es un proceso real que realiza operaciones de gestión de red mediante la generación y/o respuesta a/de mensajes SNMPv2. Todas las posibles operaciones de una entidad se pueden restringir a un subconjunto de las operaciones que puede efectuar el entorno de gestión ("SNMPv2 Party" o EG). Una entidad SNMPv2 podría pertenecer a múltiples entidades gestoras, y mantiene las siguientes bases de datos locales:

- Una base de datos para todos los EG que conoce la entidad, que podrían ser:

    - Operación local
    - Operación local realizada por interacciones con EG o dispositivos remotos
    - Operación realizada por otras entidades SNMPv2

- Otra base de datos que representa todos los recursos de los objetos gestionados que conoce la entidad
- Como mínimo, una base de datos que representa una política de control de acceso que define los privilegios de acceso de acuerdo con los EG conocidos

Una entidad SNMPv2 puede actuar como agente o como Administrador de SNMPv2.

b. Entorno de gestión ("SNMPv2 Party" o EG)

Un entorno de gestión es un entorno de ejecución virtual cuyas operaciones se restringen, por razones de seguridad o de otra índole, a un subconjunto definido administrativamente de todas las operaciones que puede realizar una entidad SNMPv2 particular. Arquitectónicamente, cada EG comprende:

- Una identidad unívoca del entorno
- Una localización lógica de red en la que se ejecuta el EG, caracterizada por un dominio del protocolo de transporte y por información de direccionamiento del nivel de transporte
- Un sólo protocolo de autentificación y parámetros asociados con los que se autentifican el origen y la integridad de los mensajes del protocolo generados por el entorno
- Un sólo protocolo de privacidad y parámetros asociados con los que los mensajes de protocolo que recibe el entorno se protegen de cualquier intrusión.

El protocolo SNMPv2 es del tipo simple demanda / respuesta. La unidad básica de intercambio es el mensaje, en cual consiste de un envoltorio exterior de mensaje y un PDU (protocol Data Unit, Protocolo de unidad de Datos) en su interior. Ocho tipos PDUs pueden ser transportados en un mensaje SNMP. El formato general del mensaje para SNMPv2 se muestra a continuación."[9]

PDU type	request-id	0	0	Variables-vinculadas

GetRequest-PDU, GetNextRequest-PDU, SetRequest-PDU, SNMPv2-Tr

InformRequest-PDU

[9] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/snmp.htm

PDU type	request-id	0	0	Variables-vinculadas

Response-PDU

PDU type	request-id	non-repeaters	Max-repetitions	Variables-vinculadas

GetBulkRequest-PDU

nombre 1	Valor 1	Nombre 2	Valor2	…	Nombre n	Valor n

Variables- vinculadas

CAMPO	**SIGNIFICADO**
Get-request	Solicita el valor de una o más variables
Get-next-request	Solicita la siguiente varible
Get-bulk-request	Busca una gran tabla
Set-request	Actualiza una o más variables
Inform-request	Administrador a administrador describiendo el MIB local.
SnmpV2-trap	Reporte de trap a agente a administrador
Response	Respuesta del agente a una solicitud Get-Request

Tabla# 1.4 Los PDU de SNMPv2

1.2.5.3 SNMPv3

SNMPv3, la última versión del protocolo SNMP incluye la funcionalidad de las versiones anteriores y tiene como principales objetivos:

- proporcionar seguridad a través de la verificación de la integridad del mensaje (asegurar que el paquete no haya sido violado durante la transmisión), encriptación (como forma de prevención) y la autentificación (permite determinar si el mensaje proviene de una fuente válida);

- utilizar al máximo posible el hardware existente;
- proveer compatibilidad con el software existente;
- facilitar la implementación de actualizaciones del protocolo;
- posibilitar el soporte necesario para el monitoreo de grandes redes; y
- llevar a cabo estos objetivos de una forma sencilla y relativamente barata.

Las entidades SNMP

"Las entidades SNMP están formadas por un motor y una o varias aplicaciones. Estas entidades tienen una composición modular, por lo que dependiendo de los módulos que se le asignen se tratará de un gestor, un agente o una combinación de los dos.

Los módulos pueden reemplazarse o actualizarse fácilmente, lo que intenta cumplir con el objetivo de facilitar las actualizaciones del sistema y permitir la compatibilidad con sistemas anteriores. La Fig. 1.2.6

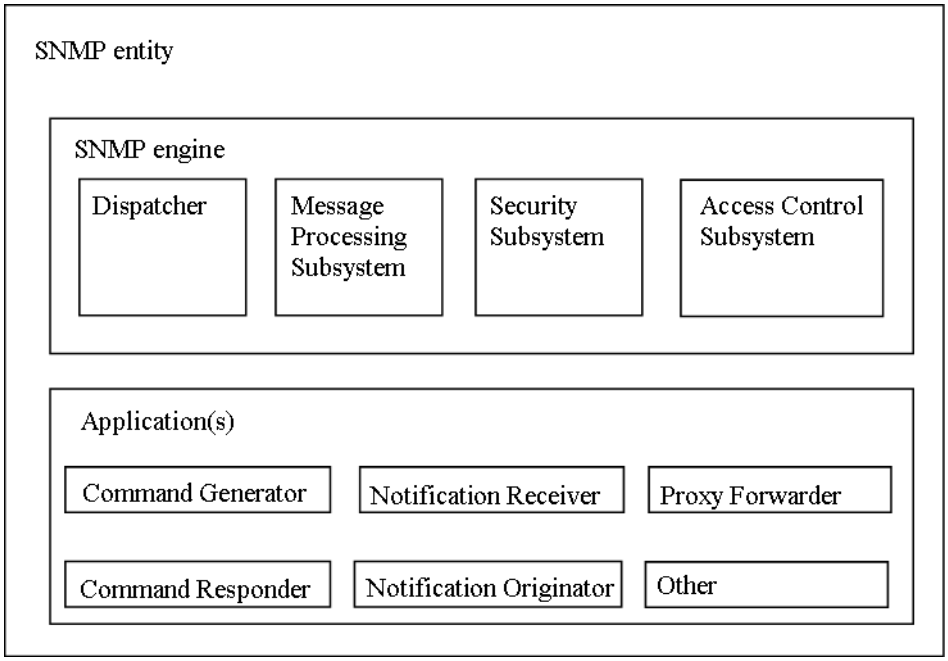


FIGURA 1.2.6 Entidades SNMP

Un motor SNMP se identifica por la variable snmpEngineID y puede estar formado por los siguientes módulos:

- Dispatcher (distribuidor)
- Message Processing Subsistem (subsistema de proceso de mensajes)
- Security Subsistem (Subsistema de seguridad)

- Access Control Subsistem (Subsistema de Control de Accesos)

Este motor se acompaña de aplicaciones que pueden ser:

- Command Generator (Generador de comandos)
- Command Responder (Respondedor de Comandos)
- Notification Receiver (Receptor de Notificaciones):
- Notification Originator (Creador de notificaciones)
- Proxy Forwarder

a. El Dispatcher

Es un "manejador de tránsito". Permite soporte a mensajes de múltiples versiones del protocolo SNMP y su tarea es:

- Intercambiar mensajes con la red (enviar y recibir mensajes);
- Determinar la versión del protocolo SNMP de los mensajes entrantes e interactuar con el subsistema de proceso de mensaje correspondiente para extraer los mensajes entrantes y armar los mensajes salientes;
- Proveer una interfaz abstracta a las aplicaciones SNMP para entregar PDUs a las otras aplicaciones y a entidades remotas.
- Colecciona estadísticas a cerca de las versiones de mensajes SNMP recibidos y enviados.

Solo puede haber un Dispatcher en una entidad SNMP.

Cuando es necesario preparar un mensaje para enviarlo a través de la red o cuando se necesita enviar datos a otras aplicaciones de la misma entidad, el Despachador llama al Subsistema de Proceso de Mensajes

b. Subsistema de Proceso de Mensajes

El Subsistema de Proceso de Mensajes prepara los mensajes para que sean enviados agregándoles el header correspondiente a la versión necesaria y extrae los datos de los mensajes recibidos.

Este Subsistema puede estar compuesto por al menos un Modelo de Proceso de mensajes. Puede haber un Modelo de Proceso de Mensajes para cada versión de protocolo necesario en la red.

Por ejemplo podríamos tener un Subsistema de Procesos de mensajes que contenga distintos Modelos de proceso de mensajes:

- Un modelo de proceso de mensajes que encapsule y desencapsule mensajes para el protocolo SNMPv1
- Un modelo de proceso de mensajes que encapsule y desencapsule mensajes para el protocolo
- SNMPv2
- Un modelo de proceso de mensajes que encapsule y desencapsule mensajes para el protocolo SNMPv3
- Un modelo de proceso de mensajes que encapsule y desencapsule mensajes para otro protocolo ya sea existente o que pueda desarrollarse en el futuro.

Es el responsable de proporcionar compatibilidad con las versiones anteriores y futuras del protocolo. Para los mensajes entrantes la versión de protocolo es proporcionada por el Dispatcher (en algunos casos lo tomará del mensaje entrante y en otros casos lo podrá encontrar mediante un algoritmo) mientras que para los mensajes salientes el valor es provisto por las aplicaciones.

El subsistema de proceso de mensajes interactúa con el subsistema de seguridad para lograr la encriptación o des encriptación de los datos que la necesiten.

c. El Subsistema de seguridad

Potencialmente puede contener varios subsistemas. Es responsable de proveer los servicios de seguridad que pueden ser autenticación y privacidad en el mensaje. Este subsistema puede contener uno o varios Modelos de Seguridad.

Se ha desarrollado un Modelo de Seguridad Basado en Usuarios (User-Based Security Model), aunque este podría reemplazarse o utilizarse en conjunto con otro dependiendo de las necesidades.

Tiene como objetivos:

- Verificar que los mensajes SNMP que se reciben no hayan sido modificados durante la transmisión;
- Verificar la identidad de los usuarios que interactúan con el sistema;
- Detectar si los mensajes que llegan a la entidad son recientes;

- Cuidar la privacidad de la información enviada y recibida.

d. Subsistema de Control de Acceso

Provee un conjunto de servicios de autentificación de acceso que las aplicaciones pueden utilizarse ante operaciones de recuperación, generación de notificaciones.

El control de acceso posibilita restringir el acceso al MIB y limitar las operaciones que los gestores pueden realizar sobre los agentes.

Se define para utilizar con SNMP un Modelo de Control de Acceso basado en vistas, comúnmente llamado VACM (View-Based Access Control Model).

El modelo de control de acceso basado en vistas está constituido por cinco elementos:

- Grupos;
- Nivel de seguridad;
- Contexto;
- Vistas de MIB; y
- Políticas de acceso.

e. Aplicación Generador de comandos

Inicia las PDUs SNMP Get, GetNext, GetBulk o SetRequest que envía el sistema local y procesa las respuestas a los pedidos que antes se habían enviado.

Para iniciar las respuestas deberá llamar al Dispatcher, dándole los datos que luego formarán parte del header del mensaje, entre los que se encontrarán el destino del mensaje, la versión del protocolo a utilizar, modelo de seguridad y nivel de seguridad que serán requeridos, la PDU y una bandera indicando si espera o no respuesta entre otros.

f. Aplicación Respondedor de Comandos

Recibe las solicitudes destinadas al sistema local y luego deberá desarrollar la operación de protocolos necesaria para generar una respuesta adecuada y reenviarla a la entidad solicitante. Deberá utilizar control de acceso para verificar si el solicitante está autorizado a obtener esa información u ordenar la modificación de datos.

Una vez recibida la solicitud y determinado que el mensaje debe responderse, esta aplicación deberá determinar el tipo de mensaje entrante, comunicarse con la base de datos, preparar la respuesta y luego entregar esa respuesta al Dispatcher para que éste la envíe.

Si por el contrario se determina que esa solicitud no debe responderse se envía al solicitante un mensaje comunicando una falla en el acceso.

g. Aplicación Creador de notificaciones

Es el encargado de monitorear al sistema ante condiciones o eventos particulares y, de producirse una anormalidad, genera un mensaje Trap o Inform relativo a esas condiciones monitoreadas.

El Creador de notificaciones actúa de la siguiente manera: Primero, empleando mecanismos de filtro apropiados se determina cuál es la información que debe enviarse. Si el filtro determina que una notificación no debe enviarse se continúa el proceso, sino se recuperan variables de la Base de datos de Información local que permitan determinar la entidad a la que se le debe enviar el mensaje, el modelo de seguridad a utilizar y el nivel de seguridad requerido. Luego se hace una verificación para determinar si debe enviarse o no la notificación. Una vez concluidos estos pasos se construye una PDU que si no necesita respuesta se envía al Despachador, en caso contrario antes de que la PDU sea enviada al Despachador se indica la necesidad de una respuesta, se cachean los datos del gestor al que se le envió la información ante la posible necesidad de retransmitir los datos.

h. Aplicación Receptor de Notificaciones

Espera en modo pasivo la llegada de mensajes de notificación. Los mensajes de notificaciones son Inform (de gestor a gestor) y Trap (de agente a gestor). Si el mensaje que se recibe es de tipo Inform deberá responderse.

Lo primero que hace el Receptor de Notificaciones es registrar la llegada de la notificación y determinar de qué tipo de notificación se trata. Si se necesita una respuesta la prepara y se la envía al Despachador.

i. Aplicación Proxy Forwarder

Es una aplicación de implementación opcional, se implementa si:

- hay partes de la red que no soportan el protocolo SNMP;
- cuando es necesario tener información en cache para minimizar la carga de trabajo de los dispositivos;
- Para autentificar y autorizar peticiones

Se encarga de adelantar mensajes. Usa primitivas del Despachador para adelantar cuatro tipos de mensajes:

- Los mensajes creados por la aplicación Generador de Comandos, o sea los mensajes que el gestor le envía al cliente: determina a qué motor debería ir el mensaje y entrega la respuesta que antes se había recibido de ese motor.
- Los mensajes creados por la aplicación Creador de Notificaciones, o sea que contienen notificaciones ya sea Trap o Inform: el Proxy Forwarder debe determinar qué motores deberán recibir la notificación.
- Los mensajes creados por la aplicación respondedora de comandos, o sea las respuestas que el Agente le envía al Gestor: en este caso el Proxy determina las solicitudes y notificaciones que antes estuvieron en juego para adelantar la respuesta.
- Mensajes que contienen indicaciones de reporte: el proxy determina qué clases internas de PDU y que notificaciones previas están en juego.

Para que el proxy pueda llevar a cabo su tarea debe basarse en la información de contexto que le permitirá determinar: qué motores accedieron a la información y cómo adelantar los mensajes y qué motor deberá recibir notificaciones de la información."[10]

[10] http://neutron.ing.ucv.ve/revista-e/No6/BriceñoMaria/SNMPv3.html

CAPITULO II

ANÁLISIS DEL SISTEMA "Herramienta de Monitoreo para la Red de la UNL"

2. ANALISIS DE LA APLICACIÓN

2.1 DETERMINACIÓN DE REQUERIMIENTOS Y ALCANCE DE LA APLICACIÓN

2.1.1 Administrar

Usuarios

- El sistema debe permitir Crear usuarios tipo Administrador y tipo Clientes.
- Modificar y eliminar usuarios del sistema.
- No debe permitir campos vacios.

Parámetros

- Permitir Crear, Modificar y Eliminar Parámetros.

Catálogos

- El Sistema debe permitir Añadir Ítems al Catalogo
- Crear, Modificar y Eliminar Items

2.1.2 Herramientas

- Debe permitir realizar un escaneo de todos los equipos conectados a la red seleccionada
- Debe identificar si son equipos, routers o servidores
- Permitir seleccionar la red a escanear.
- Seleccionar el rango de ips a escanear
- Analizar el tráfico de cada equipo por cada segundo de los equipos que tienen snmp activado.

2.1.3 Configurar

Redes

- Permitir la administración de redes como Crear, Modificar y Eliminar las mismas
- Verificar que no existan campus vacios

Propiedades OID

- Permitir la Administración de las propiedades OID a través de Crear, Modificar y Eliminar alguna existente
- Debe identificar la dependencia de cada propiedad

2.2. Descripción del Sistema.

El sistema denominado Unlmonitoreo va permitir escanear identificar controlar el tráfico de los equipos conectados a uno o varias redes seleccionadas por el administrador.
Además mostrara la información relevante de cada equipo dependiendo del tipo de usuario que se conecte al mismo

Tabla # 2.1 Requerimientos Funcionales del Sistema

CÓDIGO	REQUERIMIENTO
R01	Crear Administrador y Usuarios.
R02	Modificar Administrador y Usuarios.
R03	Eliminar Administrador y Usuarios.
R04	Validar que los campos no estén vacios.
R05	Crear Parámetros
R06	Modificar Parámetros
R07	Eliminar Parámetros
R08	Crear catálogos
R09	Añadir ítems al catalogo
R11	Crear Ítems
R12	Escaneo de todos los equipos
R13	Presentar si son equipos Routers o servidores
R14	Seleccionar la red a escanear
R15	Seleccionar rango de ip a escanear
R16	Tráfico de paquetes.
R17	Crear propiedades OID
R18	Identificar dependencia entre cada propiedad

Tabla # 2.2 Atributos del sistema

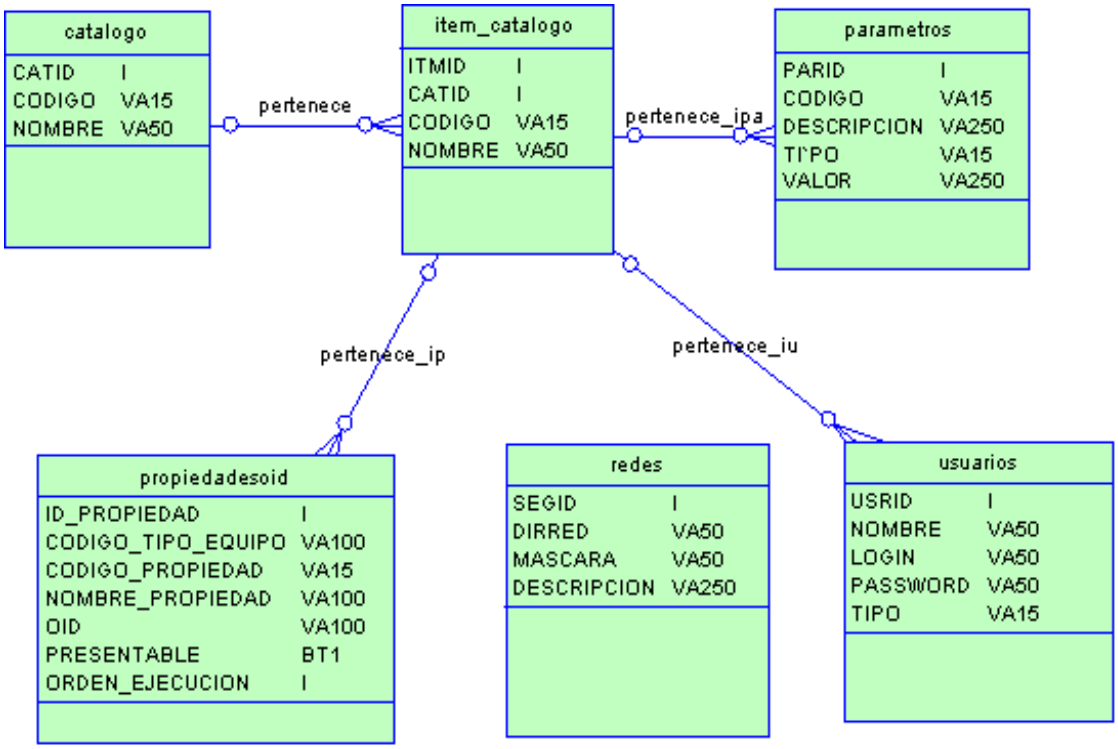
CÓDIGO	ATRIBUTO
AT01	El sistema sera multiusuario
AT02	Permitirá trabajar con el teclado y Mouse
AT03	La interfaz de usuario será amigable
AT04	El tiempo de respuesta será el más adecuado
AT05	El Herramienta será bajo la plataforma GNU/LINUX
AT06	Dispondrá de ayuda para el manejo del sistema
AT07	El sistema deberá ser tolerante a fallos.
AT08	Permitirá consultar en Equipos al los Administradores y Usuarios.

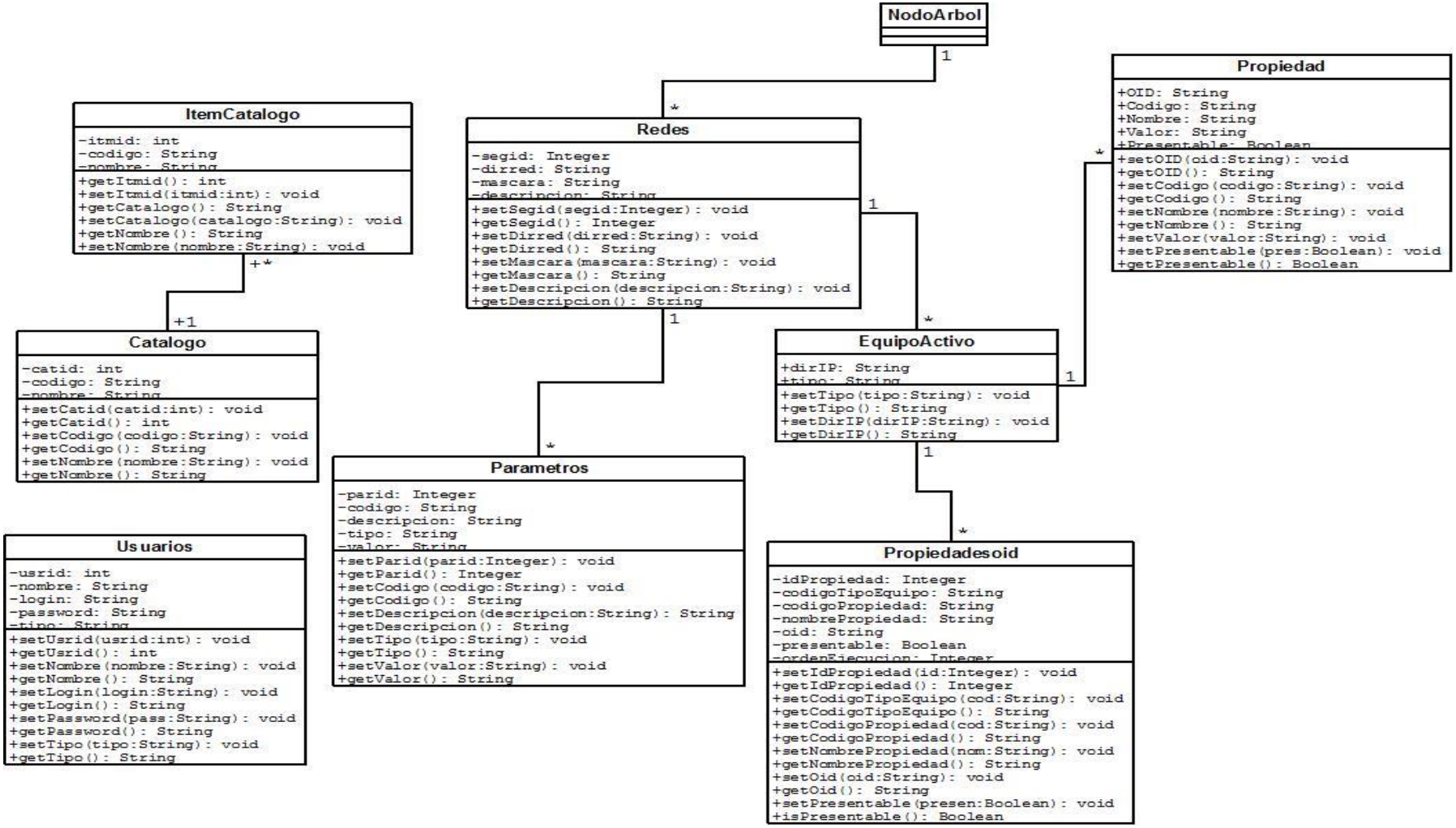
CAPITULO III

DISEÑO Y MODELADO DE LA APLICACIÓN

3.1. Diagrama de Clases.

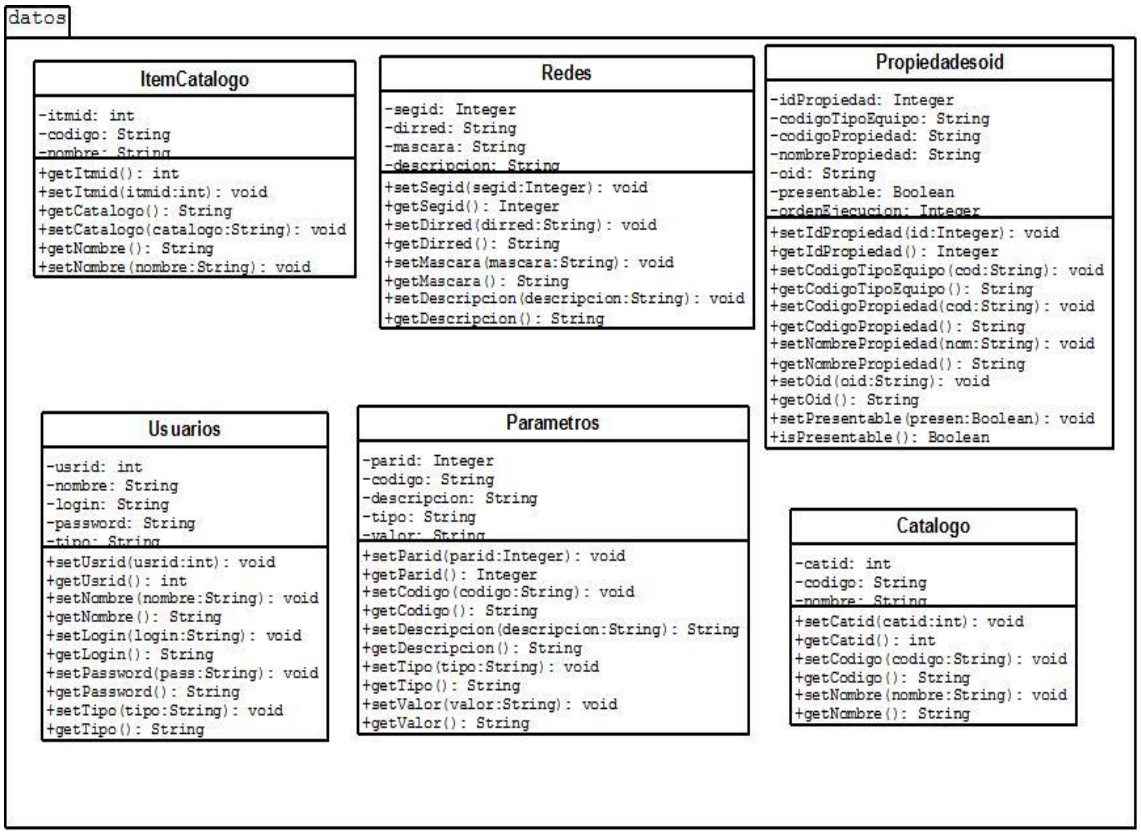
Muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Estos diagramas son los más comunes en el modelado de sistemas orientados a objetos y cubren la vista del diseño estática o la vista de procesos estática (si incluyen clases activas).

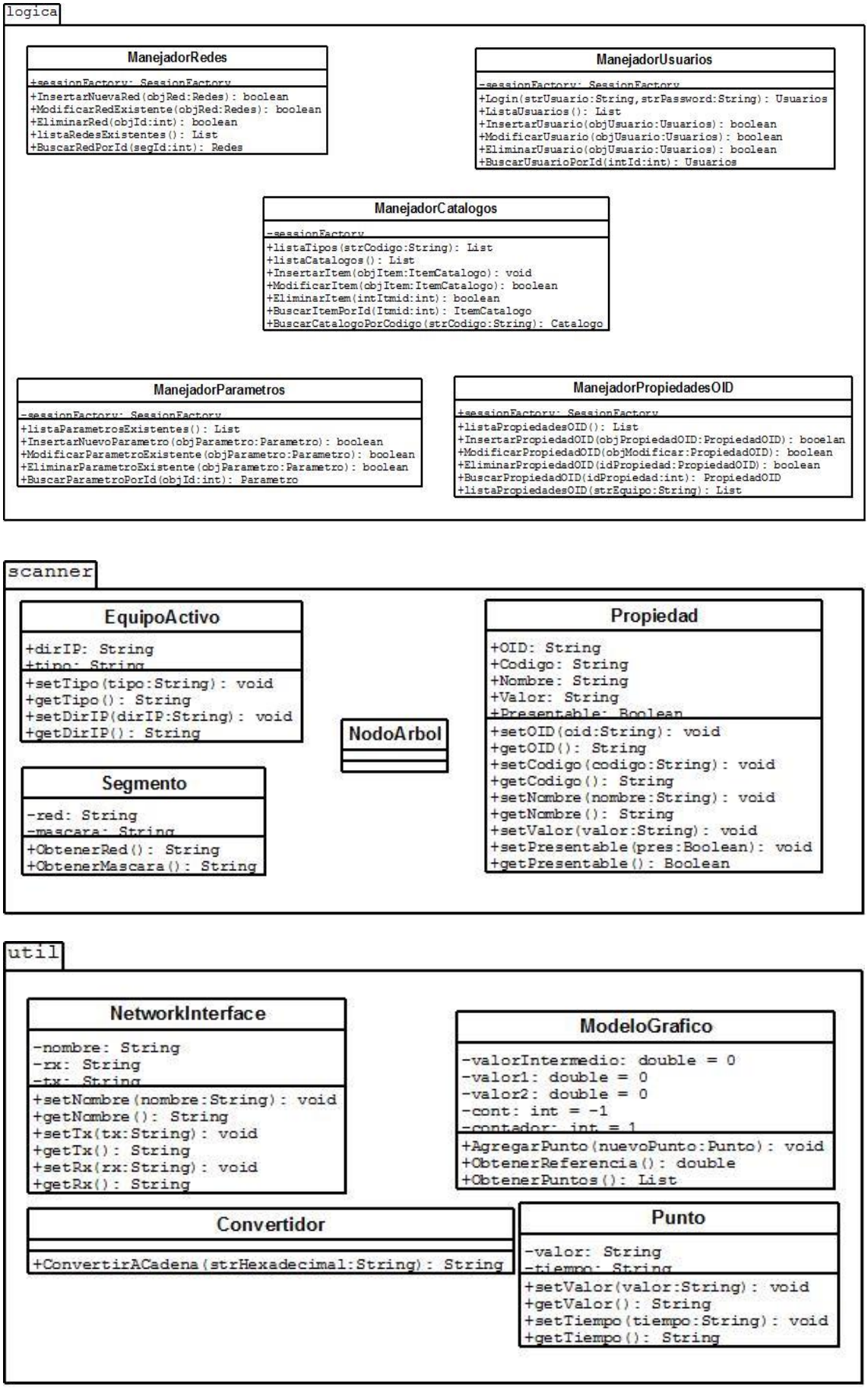



NodoArbol
1
*
Propiedad
+OID: String
+Codigo: String
+Nombre: String
+Valor: String
+Presentable: Boolean
+setOID(oid:String): void
+getOID(): String
+setCodigo(codigo:String): void
+getCodigo(): String
+setNombre(nombre:String): void
+getNombre(): String
+setValor(valor:String): void
+setPresentable(pres:Boolean): void
+getPresentable(): Boolean
ItemCatalogo
-itmid: int
-codigo: String
-nombre: String
+getItmid(): int
+setItmid(itmid:int): void
+getCatalogo(): String
+setCatalogo(catalogo:String): void
+getNombre(): String
+setNombre(nombre:String): void
Redes
-segid: Integer
-dirred: String
-mascara: String
-descripcion: String
+setSegid(segid:Integer): void
+getSegid(): Integer
+setDirred(dirred:String): void
+getDirred(): String
+setMascara(mascara:String): void
+getMascara(): String
+setDescripcion(descripcion:String): void
+getDescripcion(): String
+*
+1
Catalogo
-catid: int
-codigo: String
-nombre: String
+setCatid(catid:int): void
+getCatid(): int
+setCodigo(codigo:String): void
+getCodigo(): String
+setNombre(nombre:String): void
+getNombre(): String
EquipoActivo
+dirIP: String
+tipo: String
+setTipo(tipo:String): void
+getTipo(): String
+setDirIP(dirIP:String): void
+getDirIP(): String
Parametros
-parid: Integer
-codigo: String
-descripcion: String
-tipo: String
-valor: String
+setParid(parid:Integer): void
+getParid(): Integer
+setCodigo(codigo:String): void
+getCodigo(): String
+setDescripcion(descripcion:String): String
+getDescripcion(): String
+setTipo(tipo:String): void
+getTipo(): String
+setValor(valor:String): void
+getValor(): String
Usuarios
-usrid: int
-nombre: String
-login: String
-password: String
-tipo: String
+setUsrid(usrid:int): void
+getUsrid(): int
+setNombre(nombre:String): void
+getNombre(): String
+setLogin(login:String): void
+getLogin(): String
+setPassword(pass:String): void
+getPassword(): String
+setTipo(tipo:String): void
+getTipo(): String
Propiedadesoid
-idPropiedad: Integer
-codigoTipoEquipo: String
-codigoPropiedad: String
-nombrePropiedad: String
-oid: String
-presentable: Boolean
-ordenEjecucion: Integer
+setIdPropiedad(id:Integer): void
+getIdPropiedad(): Integer
+setCodigoTipoEquipo(cod:String): void
+getCodigoTipoEquipo(): String
+setCodigoPropiedad(cod:String): void
+getCodigoPropiedad(): String
+setNombrePropiedad(nom:String): void
+getNombrePropiedad(): String
+setOid(oid:String): void
+getOid(): String
+setPresentable(presen:Boolean): void
+isPresentable(): Boolean

3.2 Diagrama de Paquetes

Los paquetes son unidades de organización jerárquica de uso general de los modelos pueden ser utilizados para el almacenamiento, el control de acceso, la gestión de la configuración y la construcción de bibliotecas que contengan fragmentos reutilizables. Un paquete puede contener otros paquetes, sin límite de anidamiento pero cada elemento pertenece a (está definido en) sólo un paquete. Los paquetes contienen elementos del modelo al más alto nivel, tales como clases y sus relaciones, máquinas de estado, diagramas de casos de uso, interacciones y colaboraciones; atributos, operaciones, estados, líneas de vida y mensajes están contenidos en otros elementos y no aparecen como contenido directo de los paquetes.




logica
ManejadorRedes
+sessionFactory: SessionFactory
+InsertarNuevaRed(objRed:Redes): boolean
+ModificarRedExistente(objRed:Redes): boolean
+EliminarRed(objId:int): boolean
+listaRedesExistentes(): List
+BuscarRedPorId(segId:int): Redes
ManejadorUsuarios
-sessionFactory: SessionFactory
+Login(strUsuario:String,strPassword:String): Usuarios
+ListaUsuarios(): List
+InsertarUsuario(objUsuario:Usuarios): boolean
+ModificarUsuario(objUsuario:Usuarios): boolean
+EliminarUsuario(objUsuario:Usuarios): boolean
+BuscarUsuarioPorId(intId:int): Usuarios
ManejadorCatalogos
-sessionFactory
+listaTipos(strCodigo:String): List
+listaCatalogos(): List
+InsertarItem(objItem:ItemCatalogo): void
+ModificarItem(objItem:ItemCatalogo): boolean
+EliminarItem(intItmid:int): boolean
+BuscarItemPorId(Itmid:int): ItemCatalogo
+BuscarCatalogoPorCodigo(strCodigo:String): Catalogo
ManejadorParametros
-sessionFactory: SessionFactory
+listaParametrosExistentes(): List
+InsertarNuevoParametro(objParametro:Parametro): boolean
+ModificarParametroExistente(objParametro:Parametro): boolean
+EliminarParametroExistente(objParametro:Parametro): boolean
+BuscarParametroPorId(objId:int): Parametro
ManejadorPropiedadesOID
+sessionFactory: SessionFactory
+listaPropiedadesOID(): List
+InsertarPropiedadOID(objPropiedadOID:PropiedadOID): booelan
+ModificarPropiedadOID(objModificar:PropiedadOID): boolean
+EliminarPropiedadOID(idPropiedad:PropiedadOID): boolean
+BuscarPropiedadOID(idPropiedad:int): PropiedadOID
+listaPropiedadesOID(strEquipo:String): List
scanner
EquipoActivo
+dirIP: String
+tipo: String
+setTipo(tipo:String): void
+getTipo(): String
+setDirIP(dirIP:String): void
+getDirIP(): String
NodoArbol
Propiedad
+OID: String
+Codigo: String
+Nombre: String
+Valor: String
+Presentable: Boolean
+setOID(oid:String): void
+getOID(): String
+setCodigo(codigo:String): void
+getCodigo(): String
+setNombre(nombre:String): void
+getNombre(): String
+setValor(valor:String): void
+setPresentable(pres:Boolean): void
+getPresentable(): Boolean
Segmento
-red: String
-mascara: String
+ObtenerRed(): String
+ObtenerMascara(): String
util
NetworkInterface
-nombre: String
-rx: String
-tx: String
+setNombre(nombre:String): void
+getNombre(): String
+setTx(tx:String): void
+getTx(): String
+setRx(rx:String): void
+getRx(): String
ModeloGrafico
-valorIntermedio: double = 0
-valor1: double = 0
-valor2: double = 0
-cont: int = -1
-contador: int = 1
+AgregarPunto(nuevoPunto:Punto): void
+ObtenerReferencia(): double
+ObtenerPuntos(): List
Convertidor
+ConvertirACadena(strHexadecimal:String): String
Punto
-valor: String
-tiempo: String
+setValor(valor:String): void
+getValor(): String
+setTiempo(tiempo:String): void
+getTiempo(): String

3.3 Casos de uso del sistema

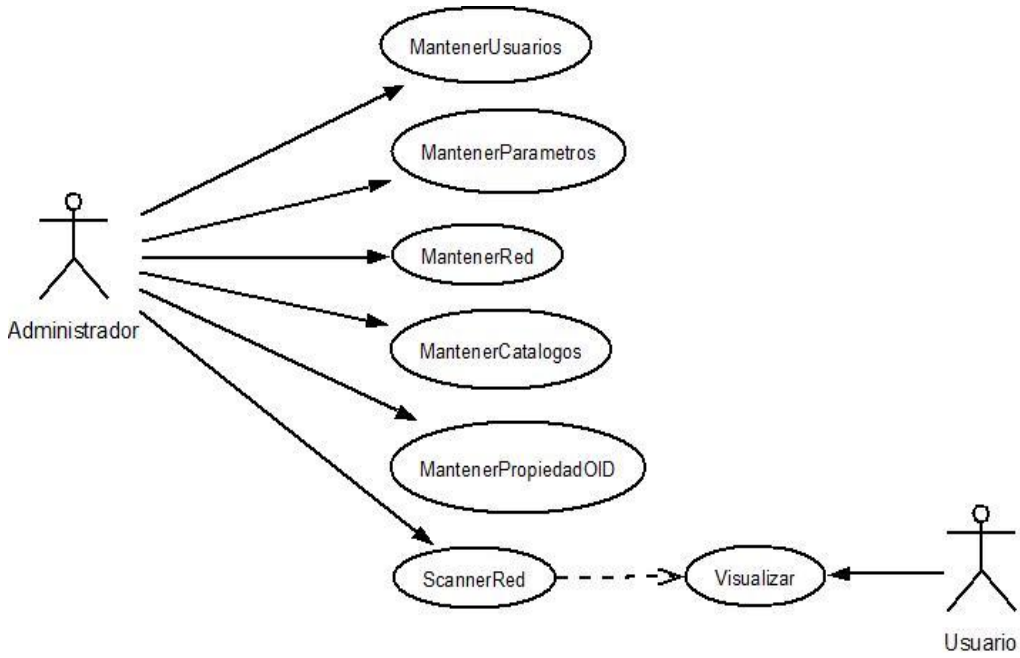


Figura 3.1 Casos de Uso del Sistema

Descripción de los Caso de Uso

- **Caso de Uso Mantener Usuarios**

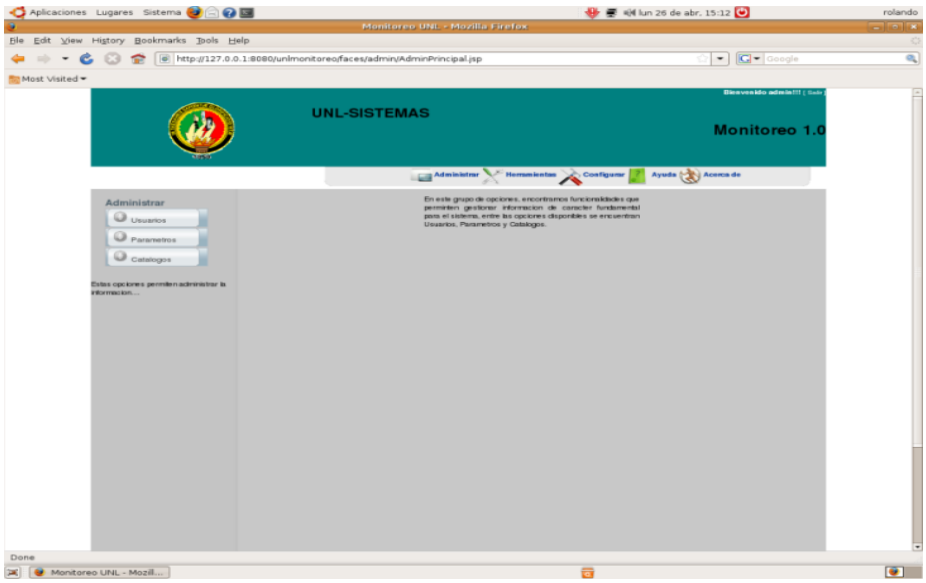


Figura 3.2 Menú Administrar

Figura 3.3 Usuario Lista

Figura 3.4 UsuarioInsertar/Modificar

Nombre	Mantener Usuarios		
Autor	Administrador		
Breve Descripción	El Administrador, estará encargado de crear usuarios para el uso de la herramienta, sean estos de tipo “administrador” o de tipo “usuario”.		
Pre-condiciones	Se debe haber ingresado al sistema. Se debe haber ingresado a la pantalla principal “AdminPrincipal”.		
Post-condiciones	Guarda el Usuario en la base de datos.		
Flujo de Eventos		**Entrada del Actor**	**Salida del Sistema**
	0	Elige “Usuarios” del menú “Administrar” (fig 3.2).	
	1		Presenta la pantalla “UsuarioLista”.
	2	Presiona el botón “Nuevo” en la pantalla Usuario Lista (fig 3.3).	
	3		Presenta la pantalla “UsuarioInsertarModificar” (fig 3.4).
	4	Ingresa la información y presiona el botón “Guardar”.	
	5		Valida la información
	6		Guarda el usuario
	7		Actualiza y presenta la pantalla “UsuarioLista”
Curso Alterno A	A1. El administrador selecciona un usuario de la tabla y presiona el botón “Modificar”, continua con el paso 3. A2. El administrador selecciona un usuario de la tabla y presiona el botón “Eliminar”, el sistema elimina el usuario de la base de datos y continúa con el paso 7.		

Curso Alterno B	B1. El administrador presiona el botón “Cancelar”, el sistema presenta la pantalla “UsuarioLista”.
Curso Alterno C	C1. La información ingresada es incorrecta, presenta un mensaje de error al Administrador.

Diagrama de Colaboración del Use Case: Mantener Usuarios

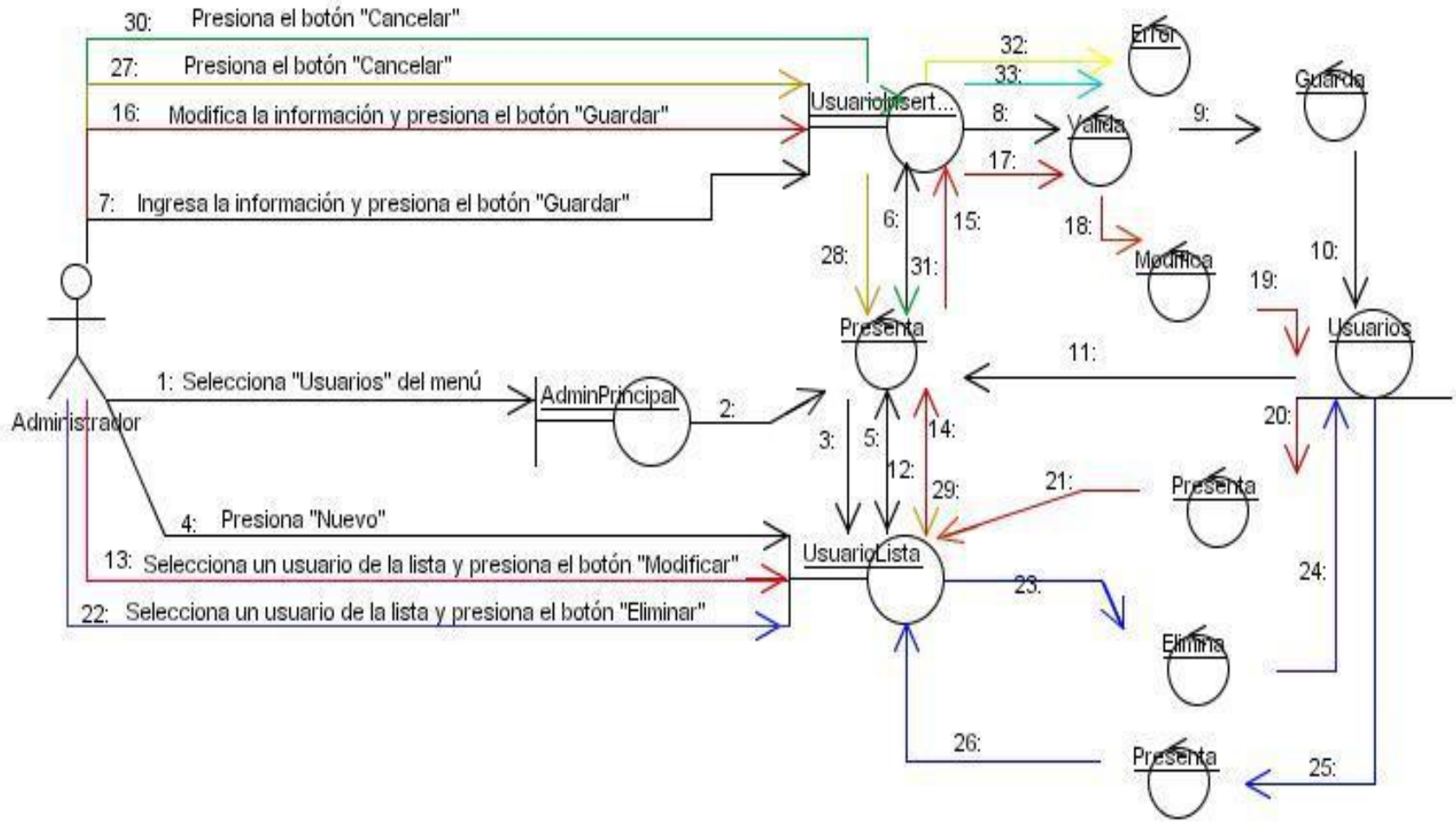
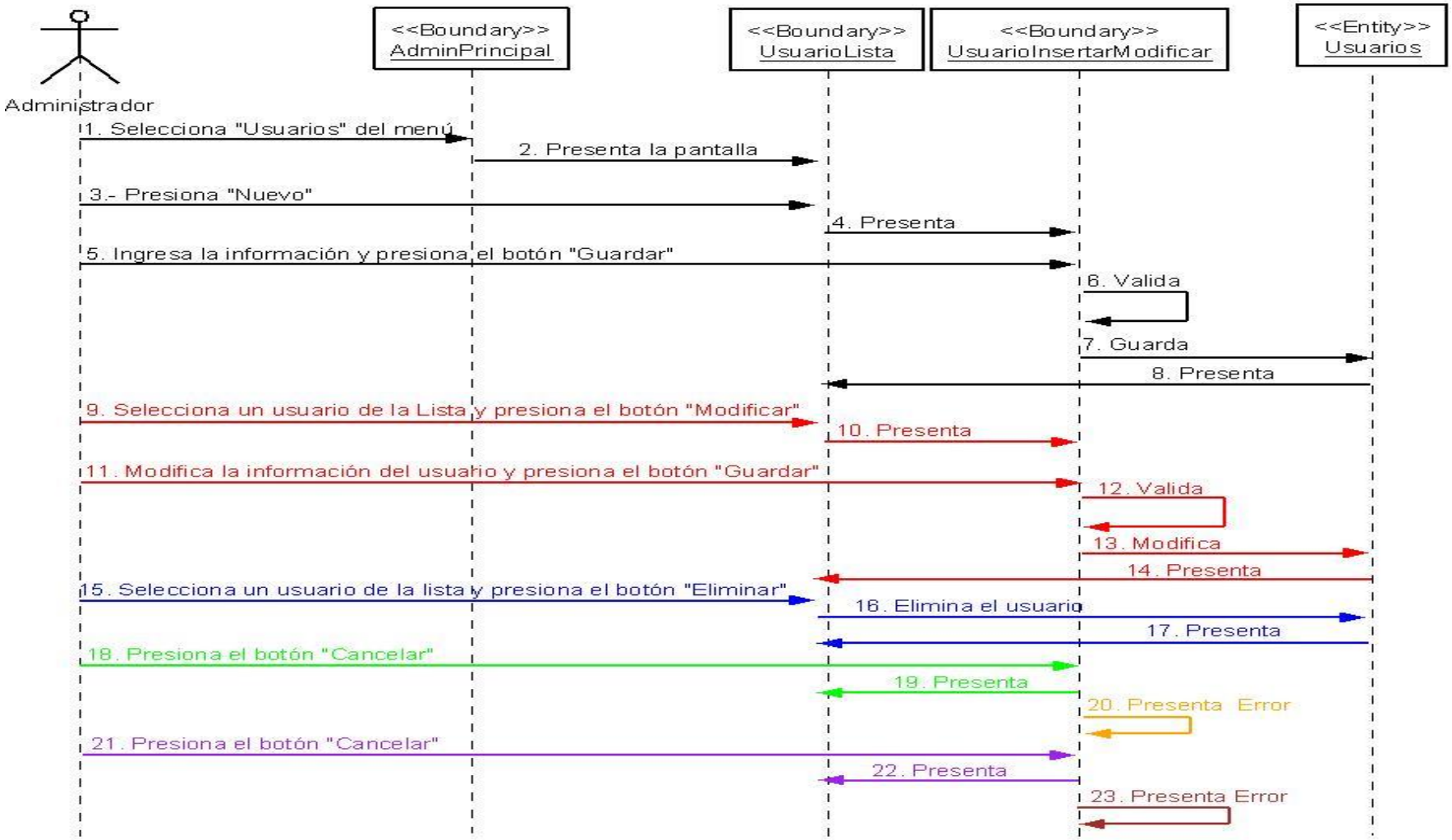


Diagrama de Secuencia del Use Case: Mantener Usuarios



❖ **Caso de Uso Mantener Parámetros**

Figura 3.5 PageParametros del Sistema

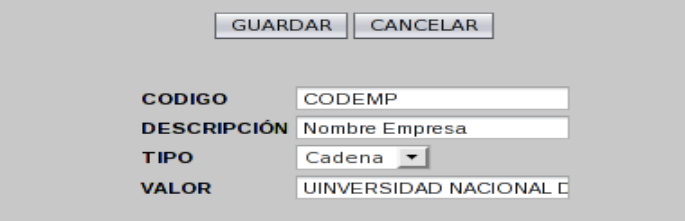


Figura 3.6 Insertar/Modificar Parametros

Nombre	MantenerParametros		
Autor	Administrador		
Breve Descripción	El Administrador, estará encargado de crear los parámetros para el uso de la herramienta.		
Pre-condiciones	Se debe haber ingresado al sistema. Se debe haber ingresado a la pantalla principal “AdminPrincipal”.		
Post-condiciones	Guarda el Usuario en la base de datos.		
Flujo de Eventos		**Entrada de Actor**	**Salida del Sistema**
	0	Elige “Parámetros” del menú “Administración”(fig 3.2).	
	1		Presenta la pantalla “PageParametros” (fig 3.5).
	2	Presiona el botón ”Nuevo”.	
	3		Presenta la pantalla “InsertarModificarParametros”(fig 3.6).
	4	Ingresa la información y presiona el botón “Guardar”.	
	5		Valida la información
	6		Guarda el parámetro.
	7		Actualiza y presenta la pantalla “PageParametros”
Curso Alterno A	A1. El administrador selecciona un parámetro de la tabla y presiona el botón “Modificar”, continua con el paso 3. A2. El administrador selecciona un parámetro de la tabla y presiona el botón “Eliminar”, el sistema elimina dicho parámetro		

	de la base de datos y continúa con el paso 7.
Curso Alterno B	B1. El administrador presiona el botón “Cancelar”, el sistema presenta la pantalla “PageParametros”.
Curso Alterno C	C1. La información ingresada es incorrecta, presenta un mensaje de error al Administrador.

***Diagrama de Colaboración del Use Case: Mantener P*arámetros**

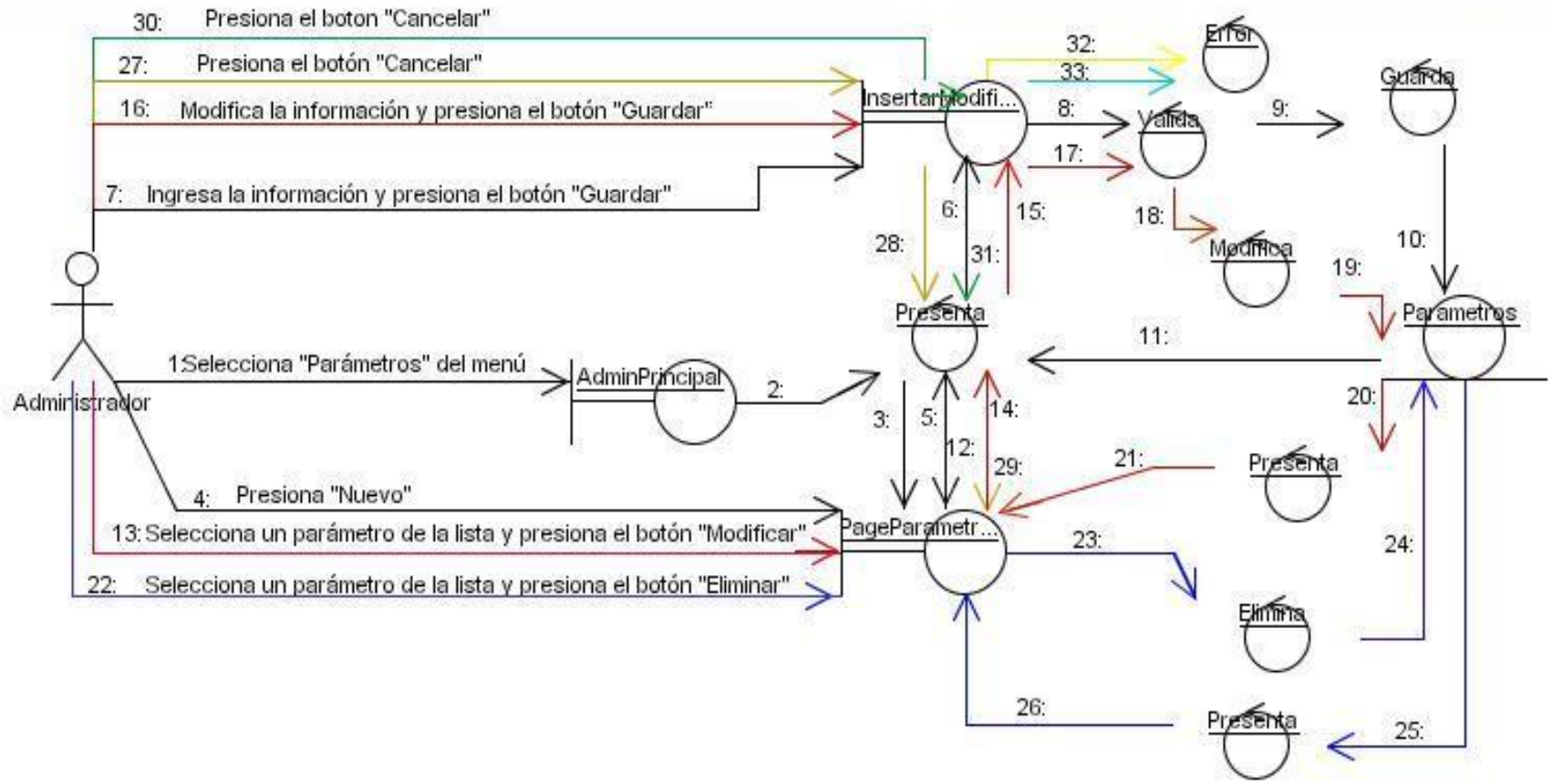
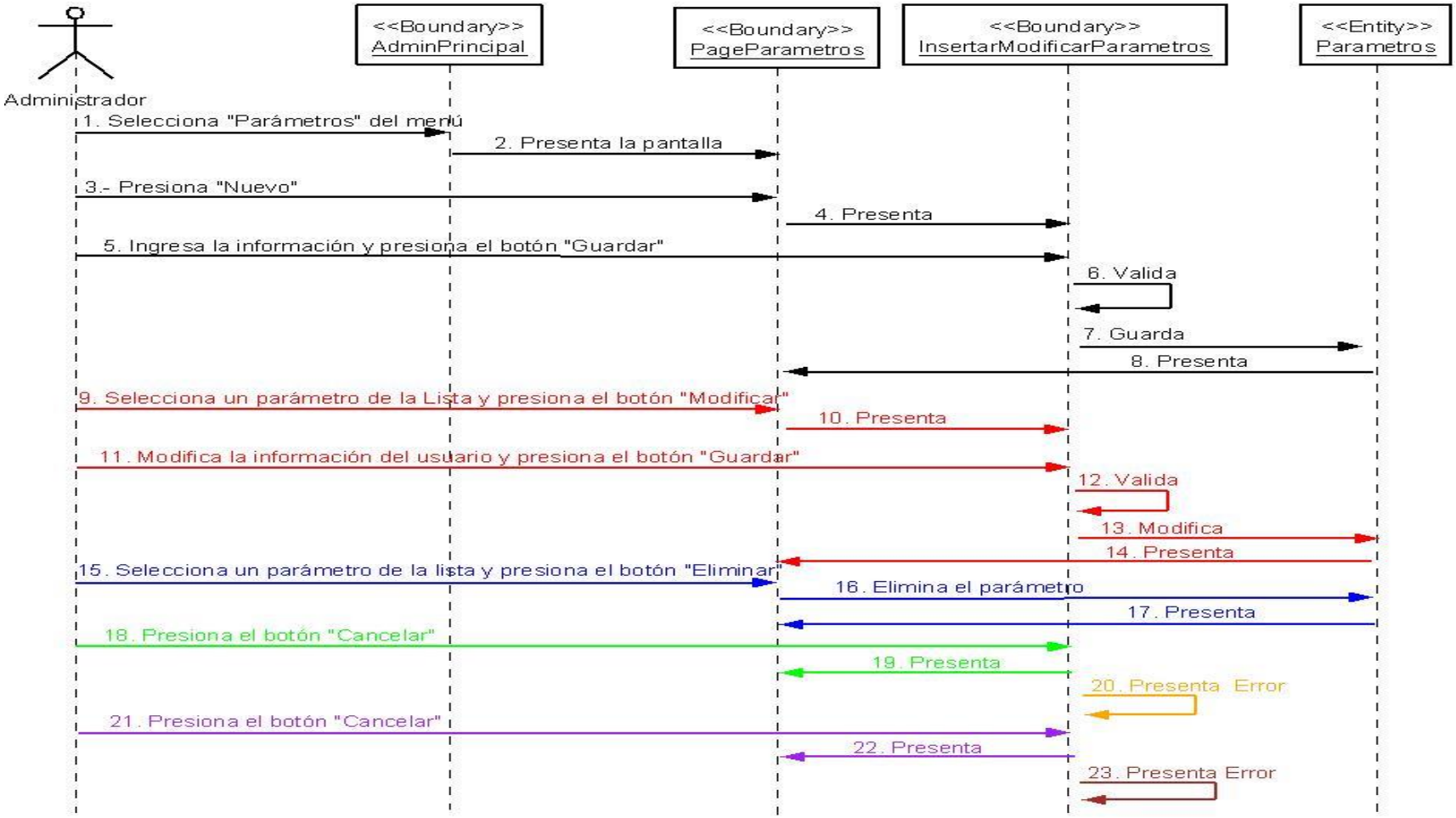


Diagrama de Secuencia del Use Case: Mantener Parámetros



❖ **Caso de Uso de Mantener Catálogos**

Figura 3.7 AdministradorCatalogos del Sistema

Figura 3.8 PageItems del Sistema

Figura 3.9 InsertarItem Catalogos

Nombre	MantenerCatalogo		
Autor	Administrador		
Breve Descripcion	El Administrador, estará encargado de crear los ítems para el catalogo necesarios para el uso de la herramienta de monitoreo de la red.		
Pre-condiciones	Se debe haber ingresado al sistema. Se debe haber ingresado a la pantalla principal “AdminPrincipal”.		
Post-condiciones	Guarda el Usuario en la base de datos.		
Flujo de Eventos		**Entrada del Actor**	**Salida del Sistema**

	0	Elige “Catálogos” del menú “Administración” (fig 3.2).	
	1		Presenta la pantalla “AdministradorCatalogos” (fig 3.7).
	2	Selecciona un catálogo de la tabla y presiona “EDITAR ITEMS”	
	3		Presenta la pantalla “PageItems”(fig 3.8).
	4	Presiona el botón “Nuevo”	
	5		Presenta la pantalla “InsertarItem”(fig 3.9).
	6	Ingresa la información y presiona el botón “Guardar”	
	7		Valida la información.
	8		Guarda el item.
	9		Presenta la pantalla “PageItems”
Curso Alterno A	A1. El administrador selecciona un ítem de la tabla y presiona el botón “Modificar”, continua con el paso 5. A2. El administrador selecciona un ítem de la tabla y presiona el botón “Eliminar”, el sistema elimina el ítem de la base de datos y continúa con el paso 9.		
Curso Alterno B	B1. El administrador presiona el botón “Cancelar”, el sistema presenta la pantalla “PageItems”.		
Curso Alterno C	C1. La información ingresada es incorrecta, presenta un mensaje de error al Administrador.		

Diagrama de Colaboración del Use Case: Mantener Catálogos

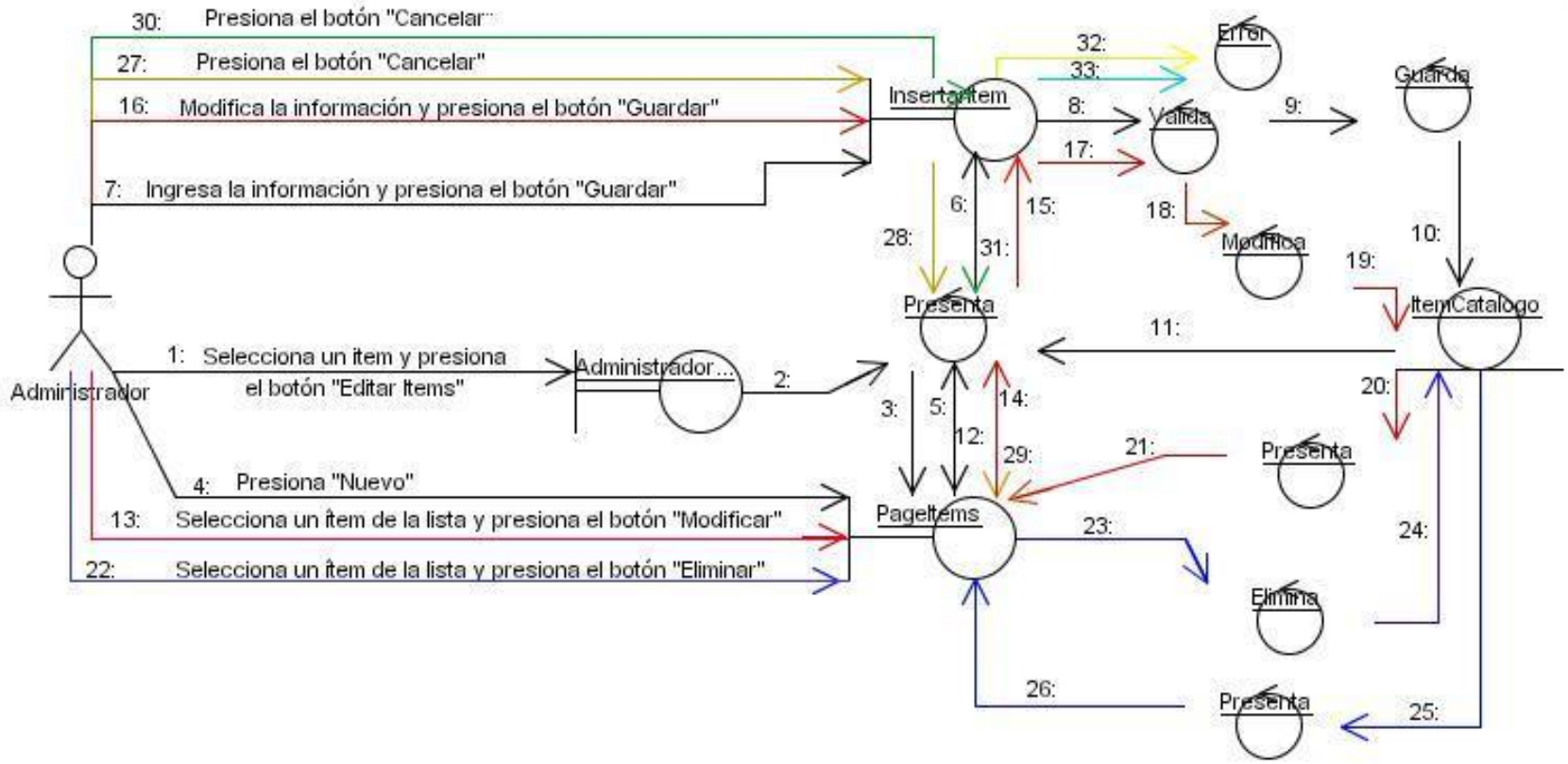
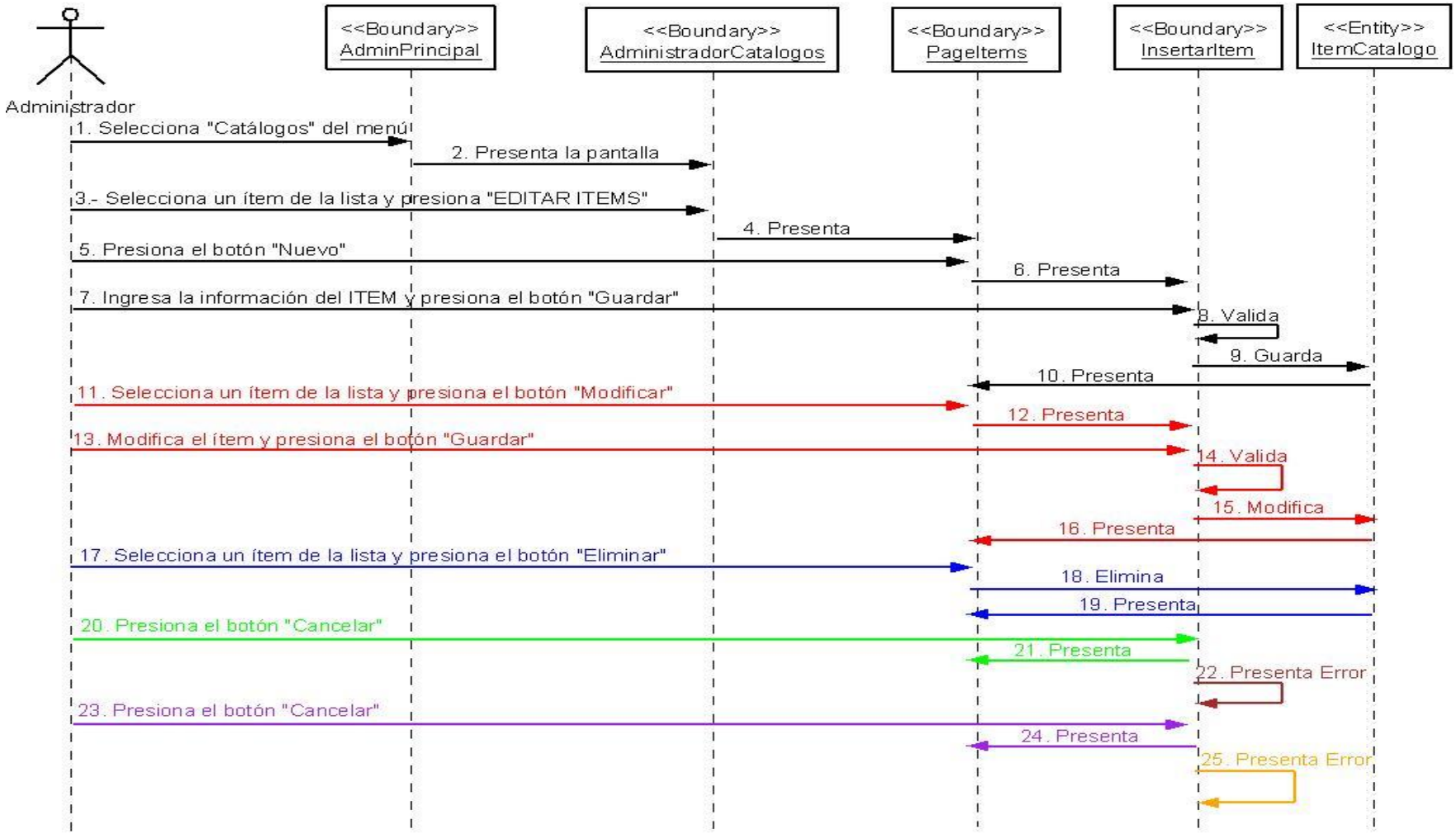


Diagrama de Secuencia del Use Case: Mantener Catálogos



- **Caso de Uso de Mantener Herramientas**

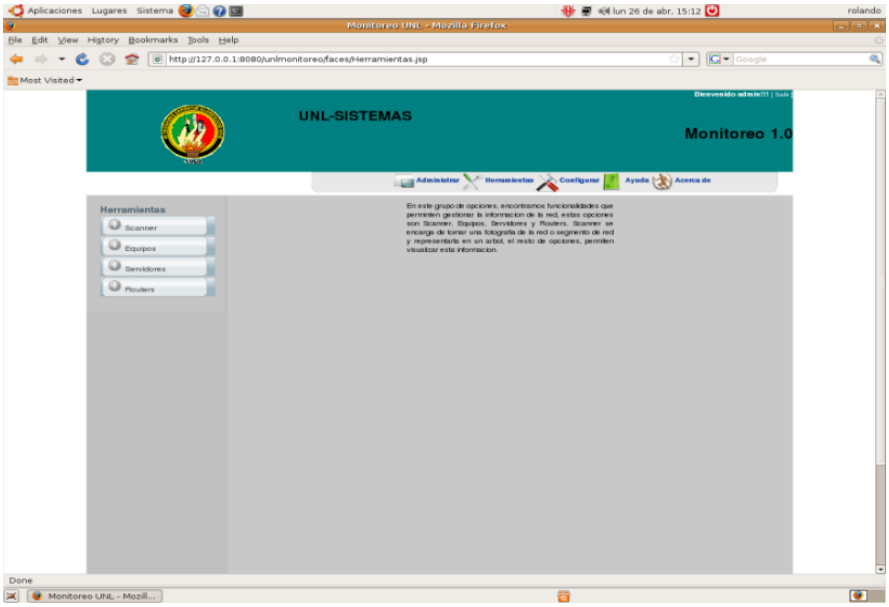


Figura 3.10 Menu Herramientas

Figura 3.11 PageScanner

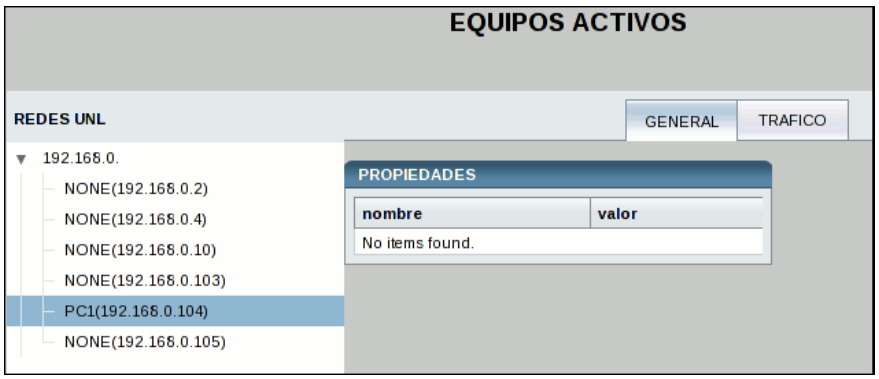


Figura 3.12 Equipos Activos

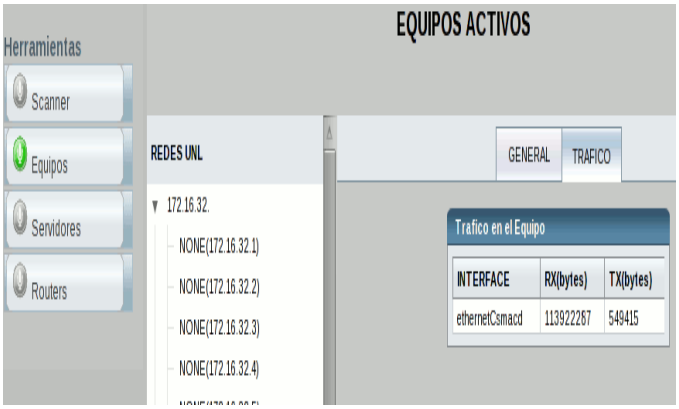


Figura 3.13 Trafico Equipos Activos

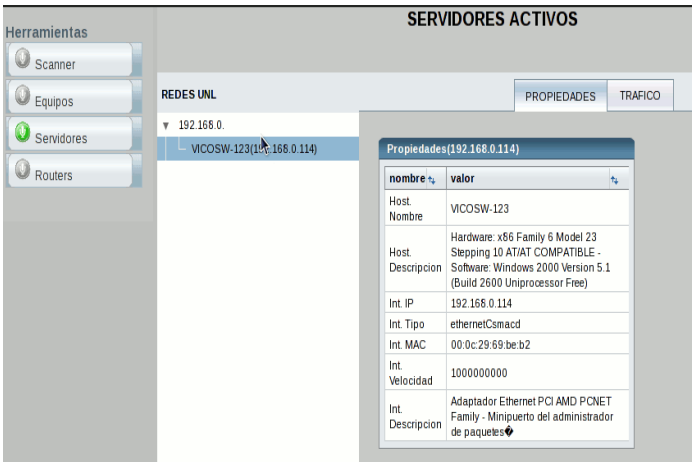


Figura 3.14 Servidores Activos

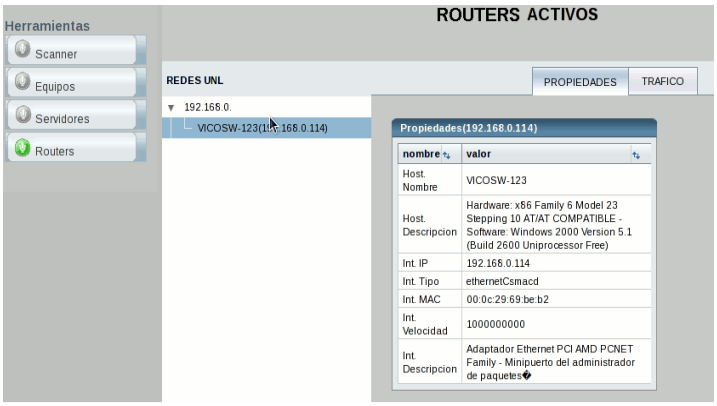


Figura 3.15 Routers Activos

Nombre	Mantener Herramientas
Autor	Administrador
Breve descripcion	El Administrador, estará encargado de escanear las redes para el uso de la herramienta.
Pre-condiciones	Se debe haber ingresado al sistema. Se debe haber ingresado a la pantalla principal “AdminPrincipal”.
Post-condiciones	Guarda el Usuario en la base de datos.
Flujo de Eventos	

	Entrada del Actor	**Salida del Sistema**
0	Elige “Scanner” del menú “Herramientas” (fig 3.10).	
1		Presenta la pantalla “PageScanner” (fig 3.11).
2	Selecciona una red el rango de las ips y presiona el botón “ScannearRedes”.	
3		Actualiza y presenta la pantalla

Curso Alterno A	A1. El usuario presiona el botón “Equipos” del menú “Herramientas”. El sistema presenta la pantalla con la información. El usuario presiona la pestaña “Trafico” el sistema presenta los paquetes enviados, recibidos y continua con el paso 2(fig 12). A2. El Usuario presiona el botón “Servidores” el sistema presenta la pantalla con la información. El usuario presiona la pestaña “Trafico” el sistema presenta los paquetes enviados, recibidos y continua con el paso 3 (fig 13,14). A3. El Usuario presiona el botón “Routers” el sistema presenta la pantalla con la información. El usuario presiona la pestaña “Trafico” el sistema presenta los paquetes enviados, recibidos y continua con el paso 3(fig 13,15).

Diagrama de Colaboración del Use Case: Mantener Herramientas

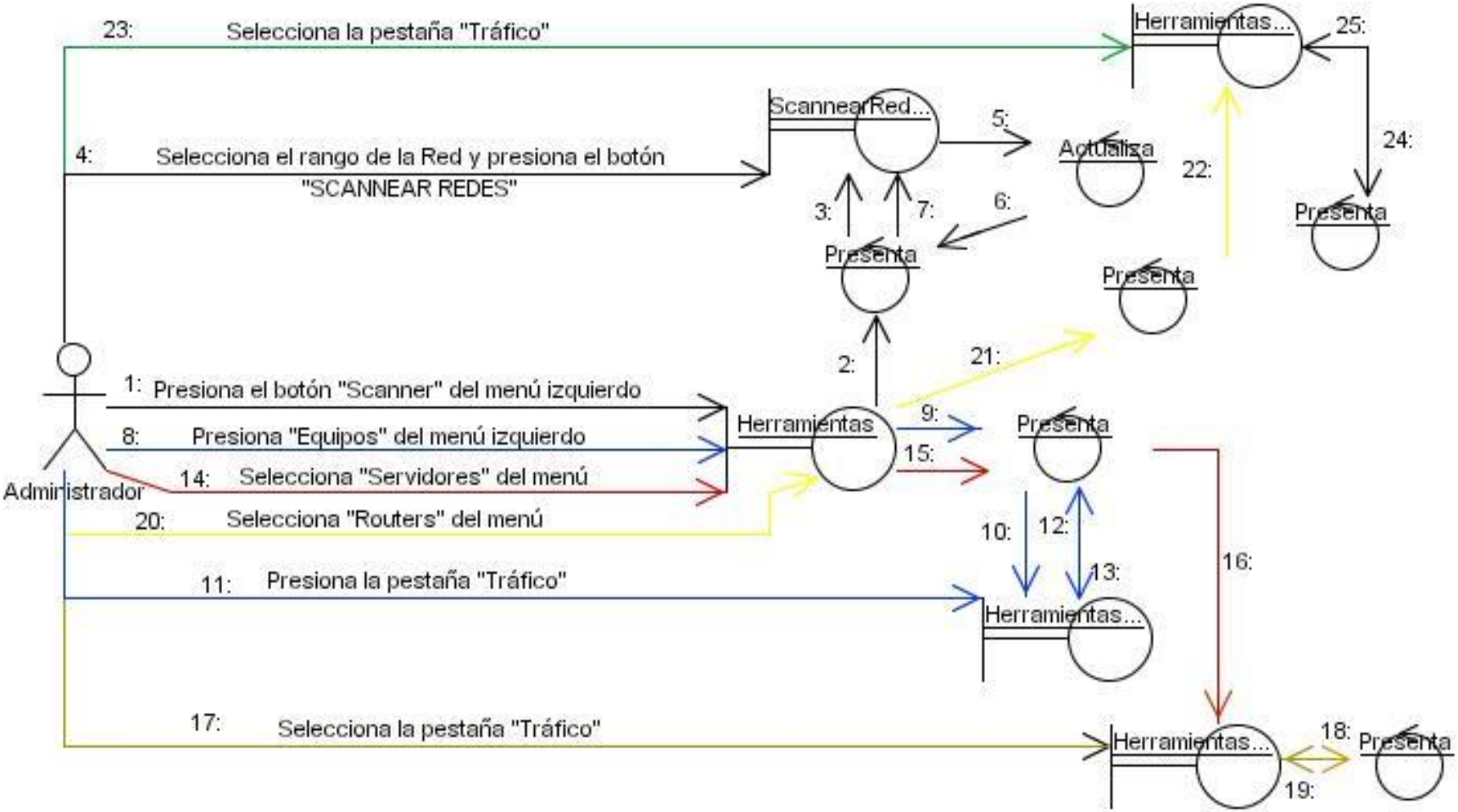
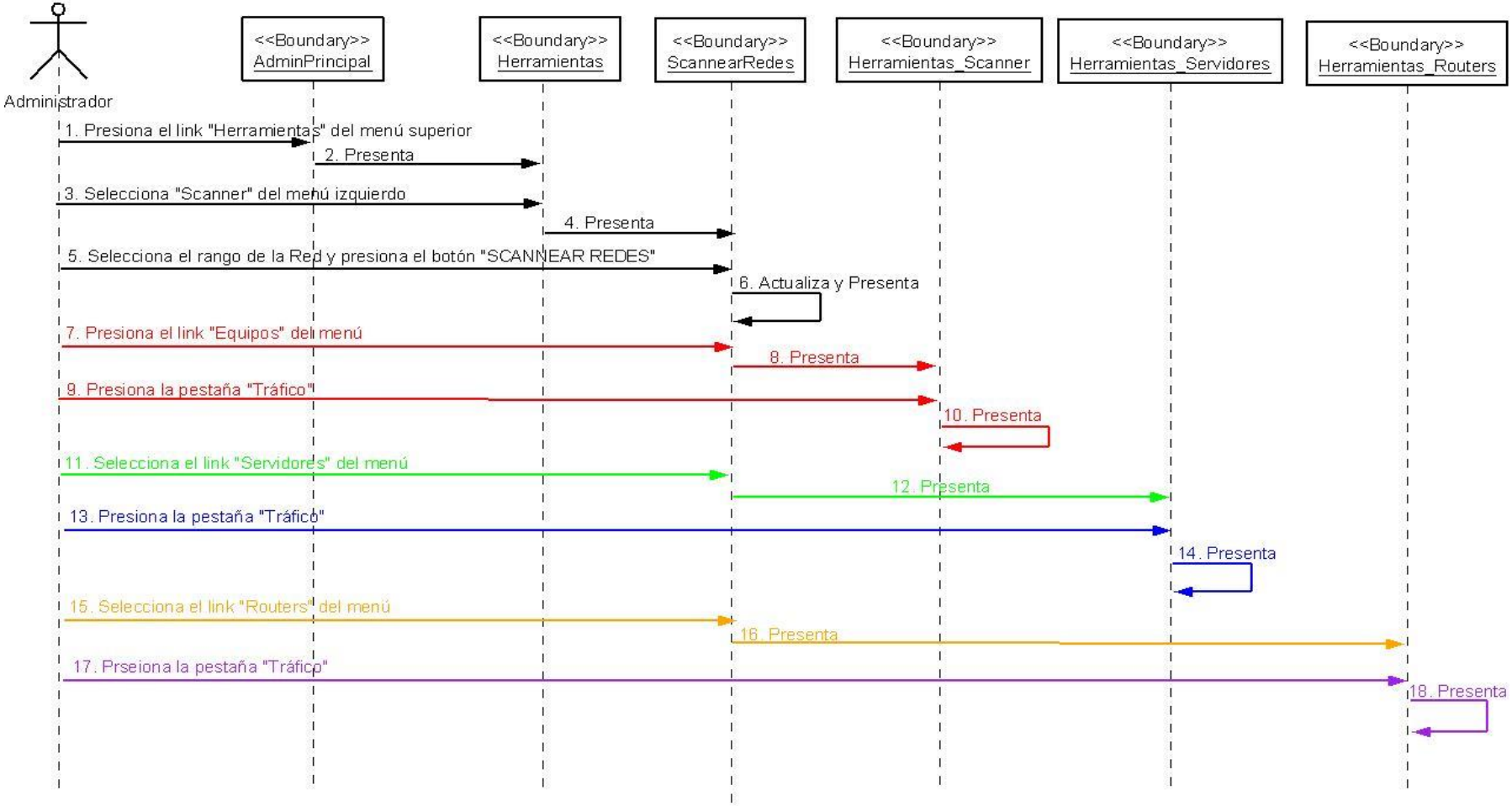


Diagrama de Secuencia del Use Case: Mantener Herramientas



- **Caso de Uso Mantener Usuario**

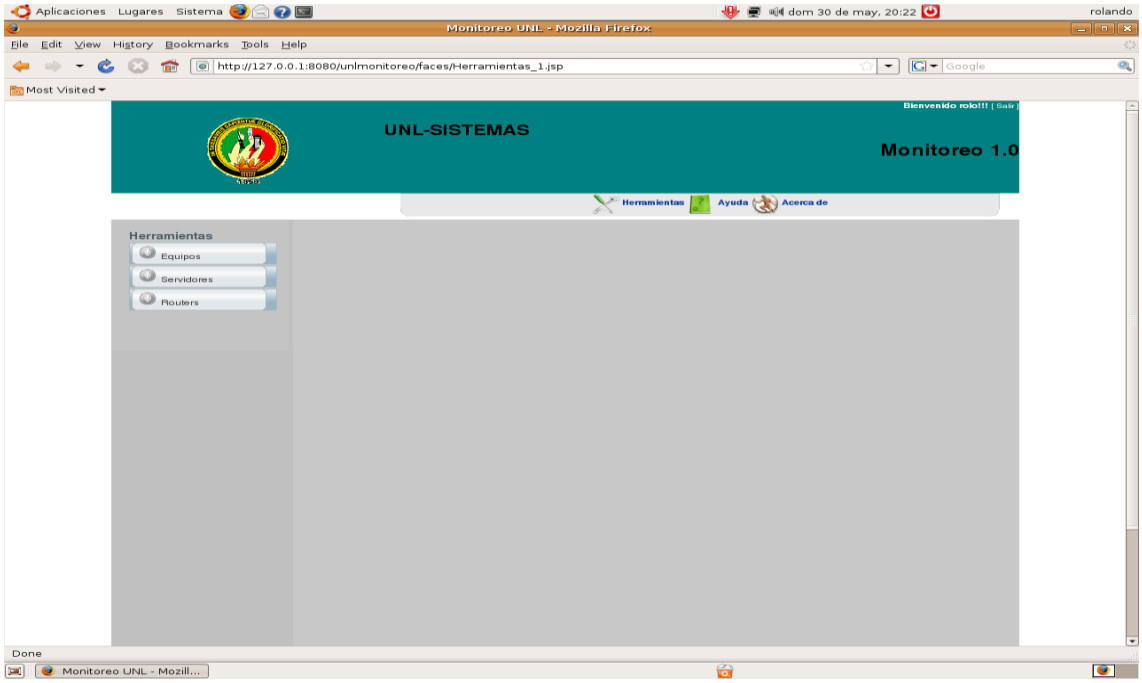


Figura 3.16 Menu Herramientas

Nombre	Mantener Usuario		
Autor	Usuario		
Breve Descripcion	El Usuario, visualizara lo escaneado en herramienta.		
Pre-condiciones	Se debe haber ingresado al sistema. Se debe haber ingresado a la pantalla principal “AdminUsuario”.		
Post-condiciones	Guarda el Usuario en la base de datos.		
Flujo of Eventos		**Entrada del Actor**	**Salida del Sistema**
	0	Elige “Equipos” del menú “Herramientas_Scanner” (fig 3.16).	
	1		Presenta la pantalla “PageUsuario” (fig 3.12).
	2	Selecciona una “ip”.	
	3		Actualiza y presenta la pantalla
Curso Alterno A	A1. El usuario presiona la pestaña “Trafico” el sistema presenta los paquetes enviados y recibidos continua con el paso 2. A.2 El Usuario presiona el botón “Servidores” el sistema presenta la pantalla con la información. El usuario presiona la pestaña “Trafico” el sistema presenta los paquetes enviados, recibidos y continua con el paso 3 (fig 13,14). A3. El Usuario presiona el botón “Routers” el sistema presenta la pantalla con la información. El usuario presiona la pestaña “Trafico” el sistema presenta los paquetes enviados, recibidos y continua con el paso 3 (fig 13,15).		

Diagrama de Colaboración del Use Case: Mantener Usuario

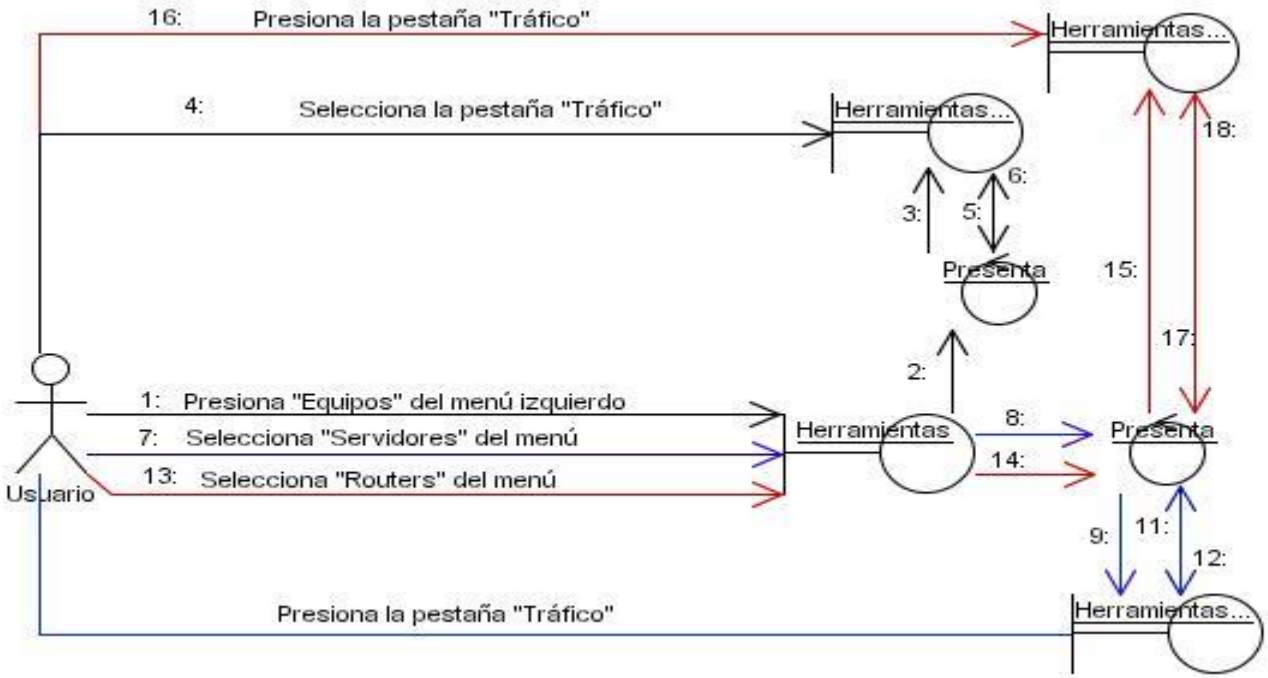
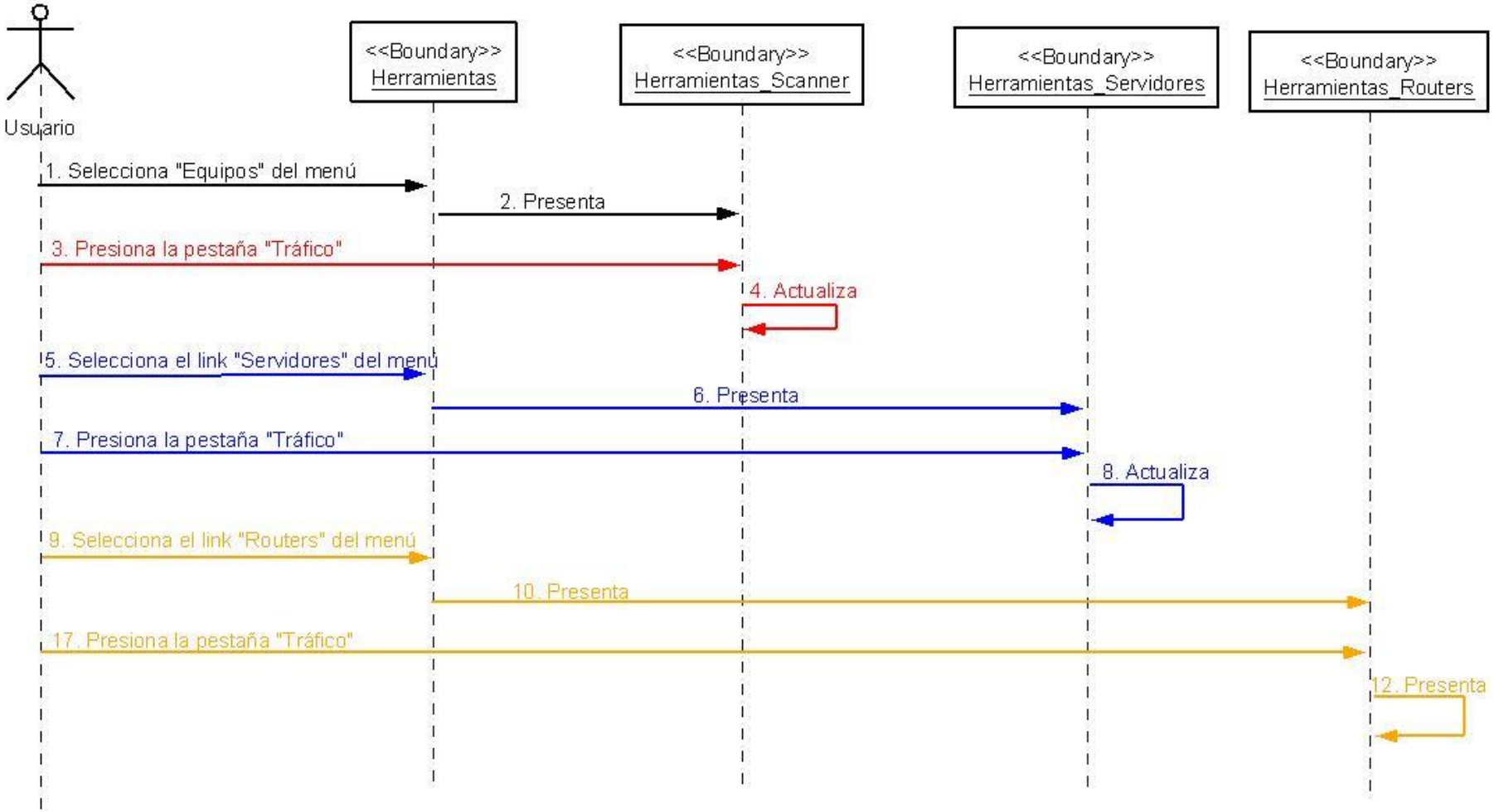


Diagrama de Secuencia del Use Case: Mantener Usuario



- **Caso de Uso Mantener Redes**

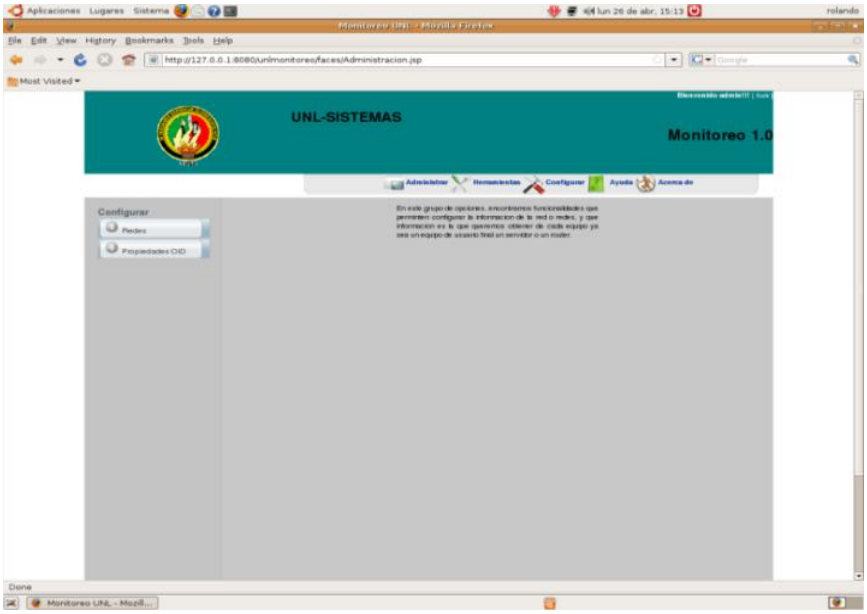


Figura 3.17 Menu Configurar

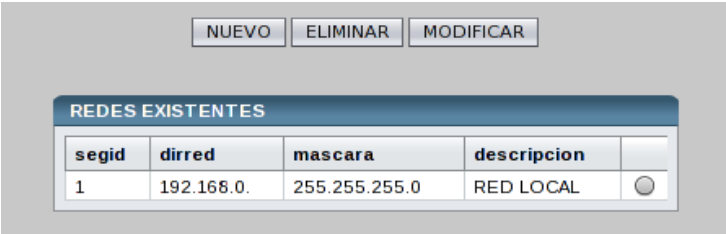


Figura 3.18 PageRedes del Sistema

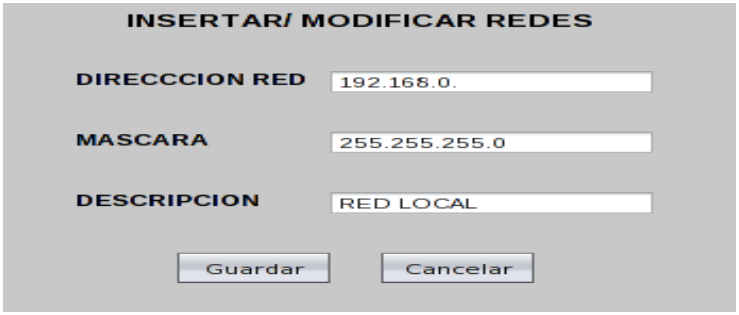


Figura 3.19 Insertar/Modificar Redes

Nombre	MantenerRedes
Autor	Administrador
Breve Descripcion	El Administrador, estará encargado de mantener la redes de la Institución, necesarias y fundamentales para el su monitoreo.
Pre-condiciones	Se debe haber ingresado al sistema. Se debe haber ingresado al link “Configurar” del menú superior.
Post-condiciones	Guarda el Usuario en la base de datos.
Flujo de Eventos	

	Entrada del Actor	**Salida del sistema**
0	Elige “Redes” del menú Configurar (fig 3.16).	
1		Presenta la pantalla “PageRedes” (fig 3.17).
2	Presiona el botón ”Nuevo”.	
3		Presenta la pantalla “InsertarRedes”(fig 3.18).
4	Ingresa la información y presiona el botón “Guardar”.	
5		Valida la información
6		Guarda la red.
7		Actualiza y presenta la pantalla “PageRedes”

Curso Alterno A	A.1 El administrador selecciona una red de la tabla y presiona el botón “Modificar”, continua con el paso 3. A.2 El administrador selecciona una red de la tabla y presiona el botón “Eliminar”, el sistema elimina dicha propiedad de la base de datos y continúa con el paso 7.
Curso Alterno B	B.1 El administrador presiona el botón “Cancelar”, el sistema presenta la pantalla “PageRedes”.
Curso Alterno C	C.1 La información ingresada es incorrecta, presenta un mensaje de error al Administrador.

Diagrama de Colaboración del Use Case: Mantener Redes

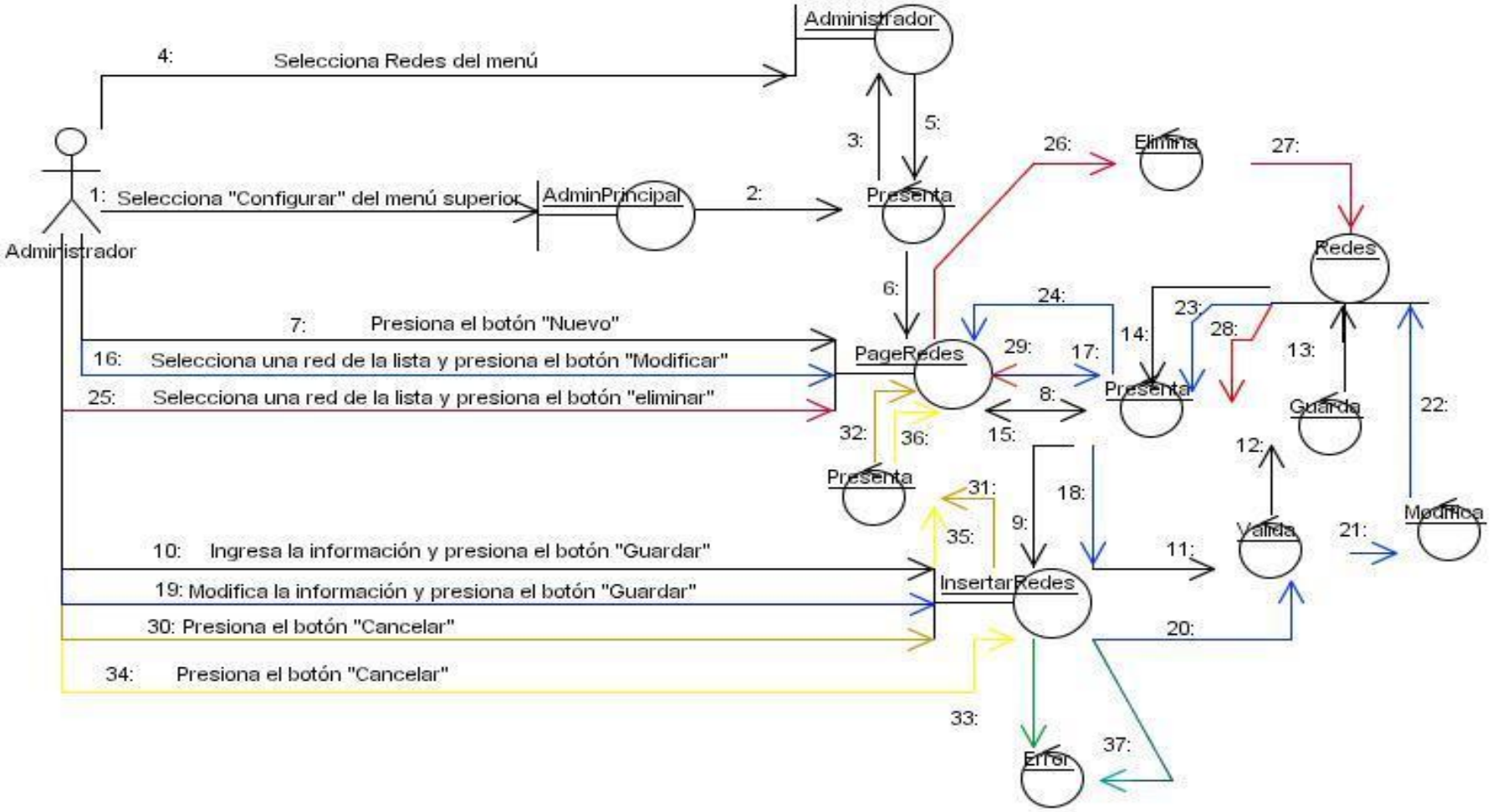
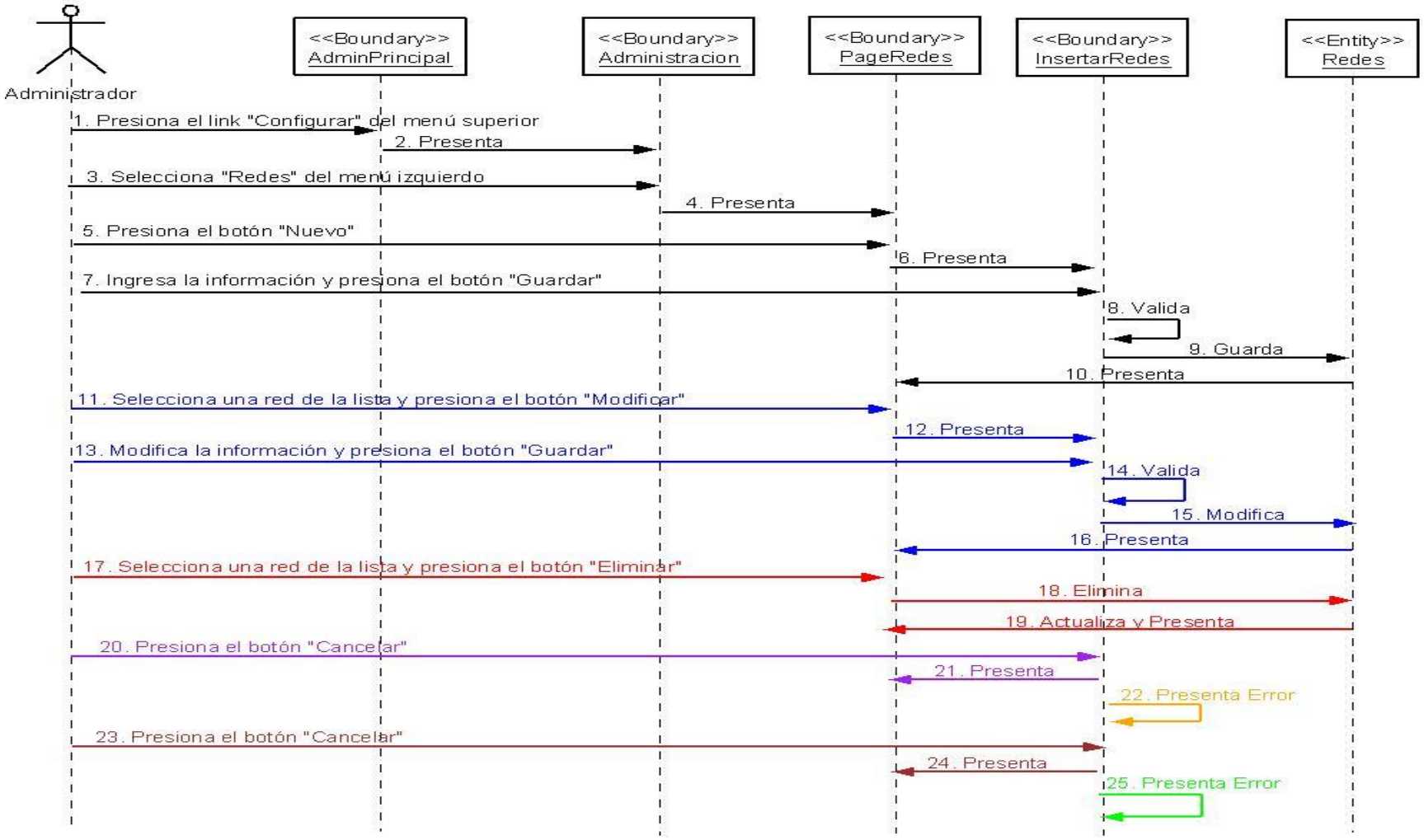


Diagrama de Secuencia del Use Case: Redes



- **Caso de Uso Mantener Propiedad OID**

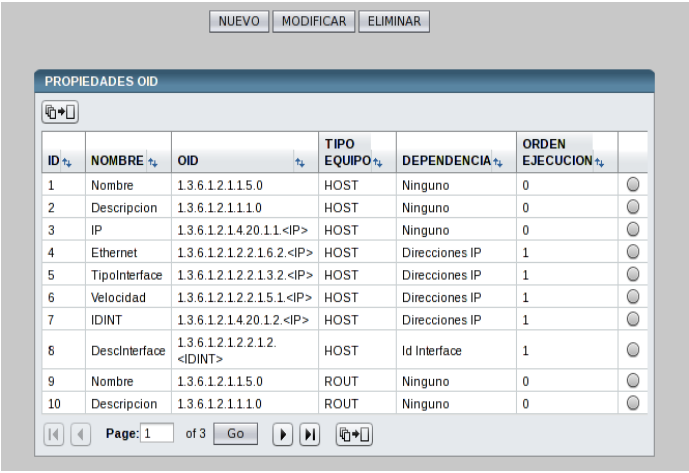


Figura 3.20 Propiedades OID del Sistema

Figura 3.21 InsertarPropiedadOID

Nombre	MantenerPropiedadOID		
Autor	Administrador		
Breve Descripcion	El Administrador, estará encargado de mantener las propiedades OID, necesarias y fundamentales para el buen desempeño del monitoreo.		
Pre-condiciones	Se debe haber ingresado al sistema. Se debe haber ingresado al link “Configurar” del menú superior.		
Post-condiciones	Guarda el Usuario en la base de datos.		
Flujo de Eventos		**Entrada Actor**	**Salida del Sistema**
	0	Elige “PropiedadesOID” del menú Configurar (fig 3.16).	
	1		Presenta la pantalla “PropiedadesOID” (fig 3.19).
	2	Presiona el botón ”Nuevo”.	
	3		Presenta la pantalla “InsertarPropiedadOID” (fig 3.20).
	4	Ingresa la información y presiona el botón “Guardar”.	
	5		Valida la información

	6		Guarda la propiedad OID.
	7		Actualiza y presenta la pantalla “PropiedadesOID”
Curso Alterno A	A.1 El administrador selecciona una propiedad de la tabla y presiona el botón “Modificar”, continua con el paso 3. 2). El administrador selecciona una propiedad de la tabla y presiona el botón “Eliminar”, el sistema elimina dicha propiedad de la base de datos y continua con el paso 7.		
Curso Alterno B	El administrador presiona el botón “Cancelar”, el sistema presenta la pantalla “PropiedadesOID”.		
Curso Alterno C	La información ingresada es incorrecta, presenta un mensaje de error al Administrador.		

Diagrama de Colaboración del Use Case: Mantener PropiedadOID

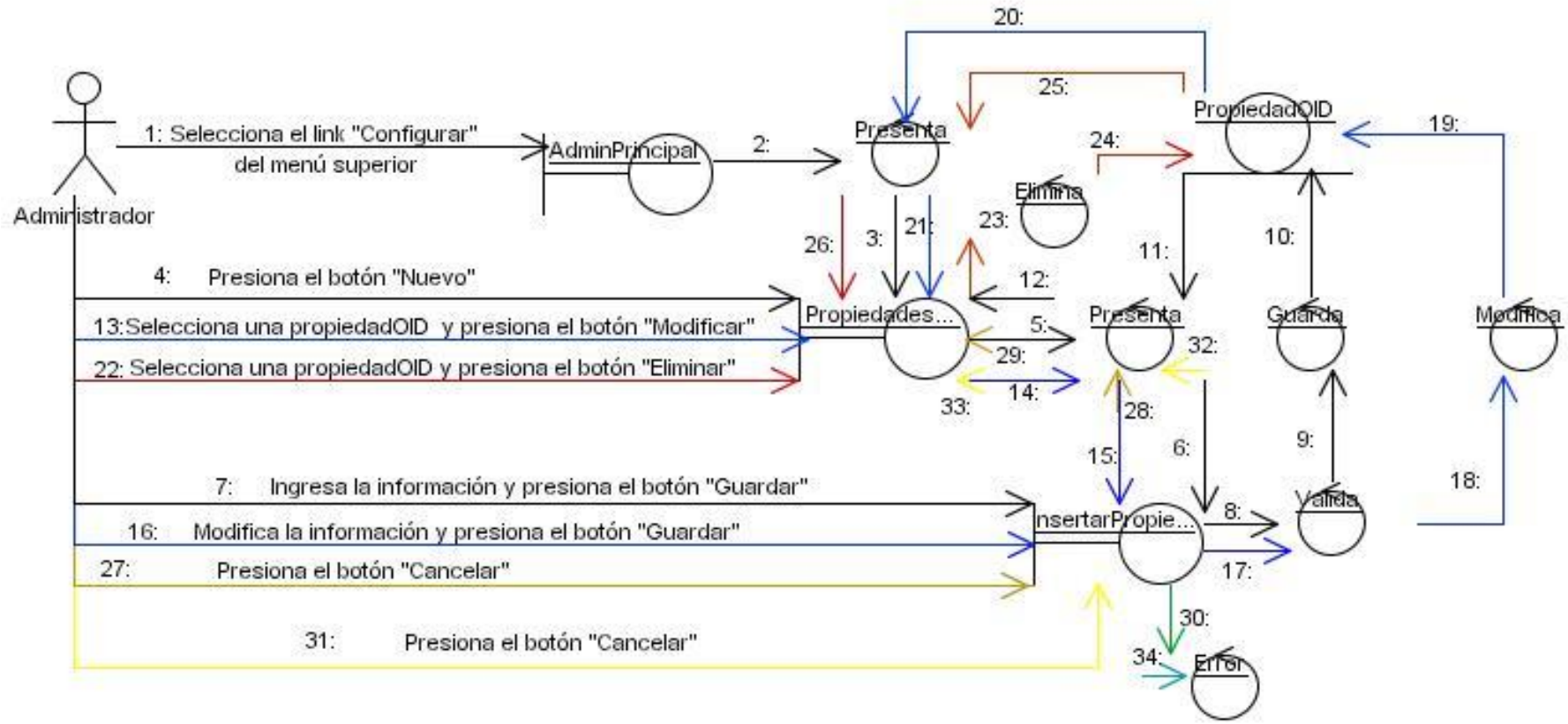
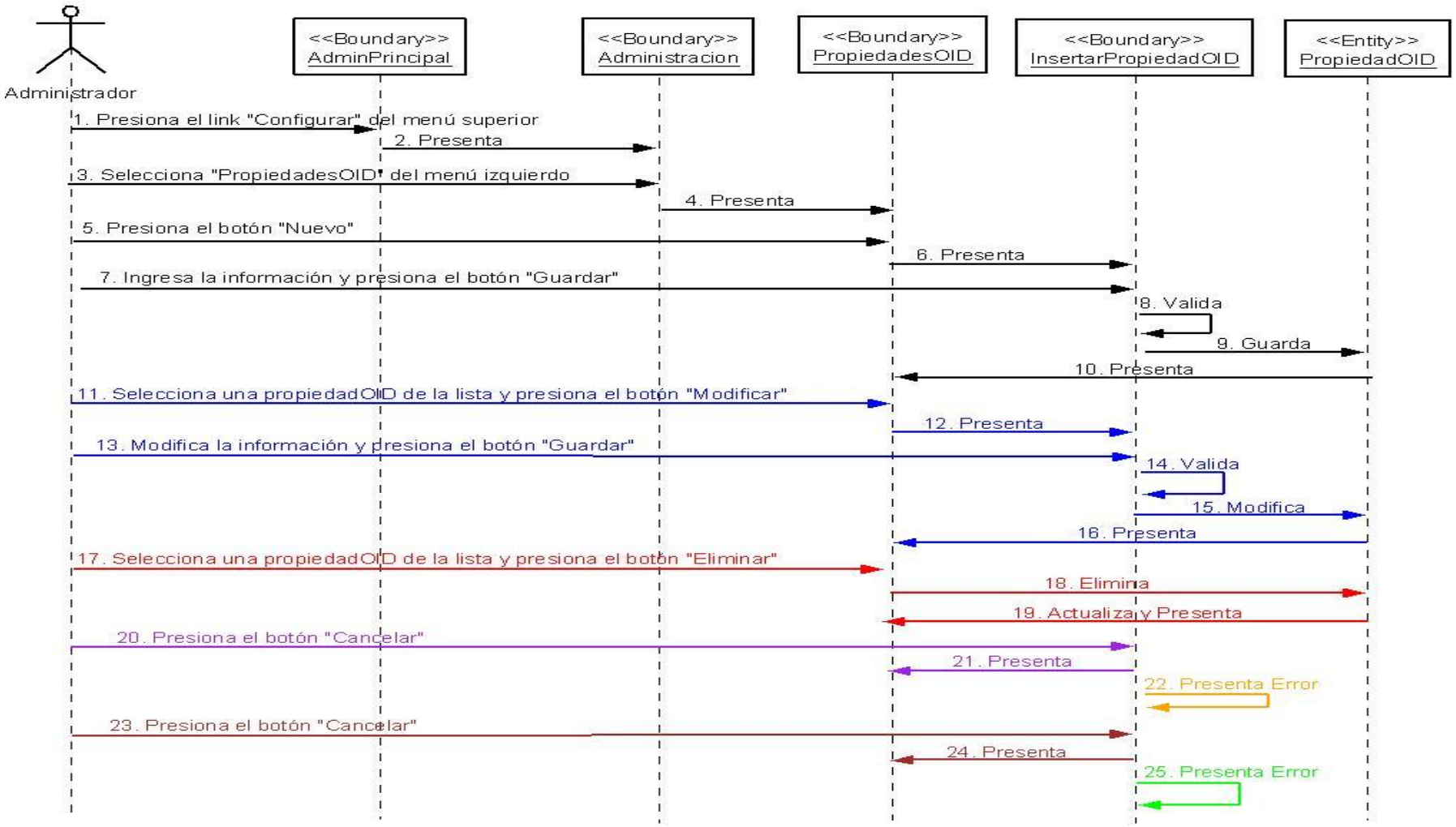


Diagrama de Secuencia del Use Case: Mantener PropiedadesOID



3.4. Diagramas de paquetes del sistema

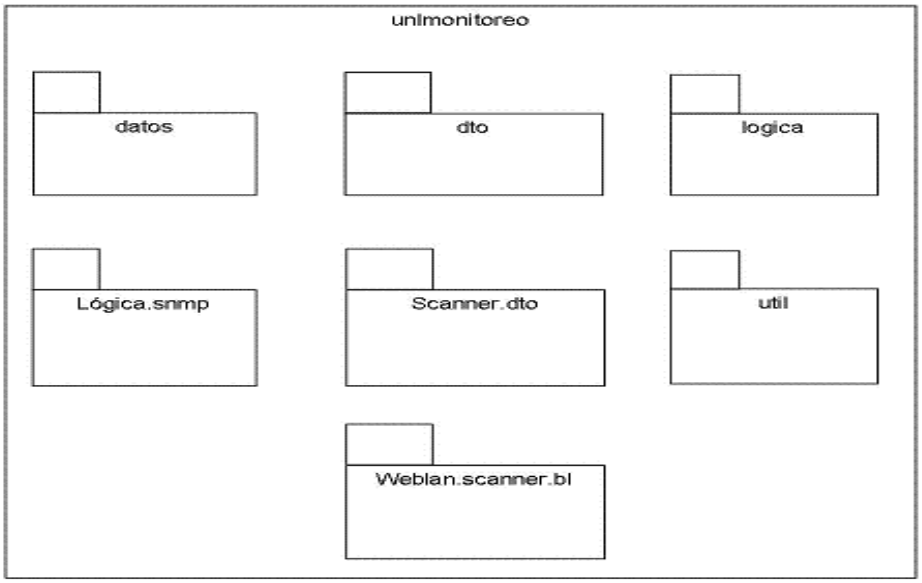
3.4.1. Arquitectura de la Aplicación

La aplicación desarrollada se basa su construcción de software por capas. La arquitectura de Unlmonitoreo considera las siguientes características:

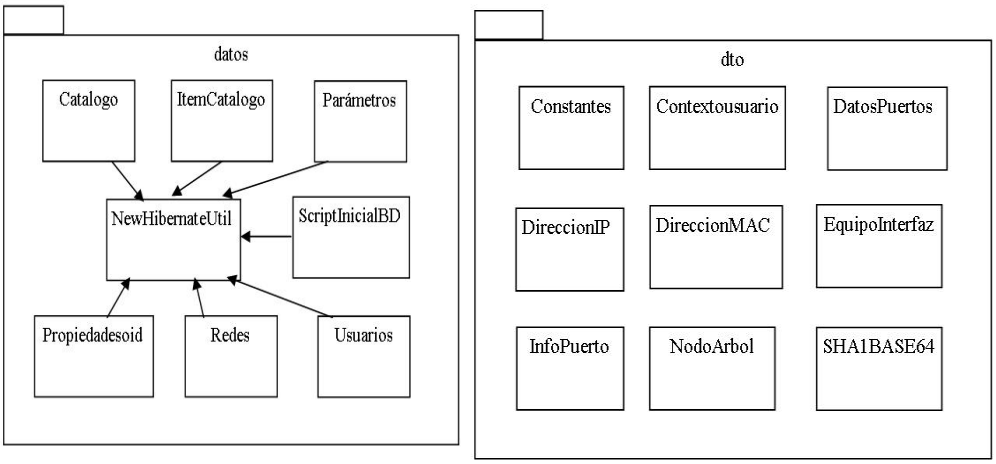
- **Capa de Dominio:** Es el nivel bajo de la aplicación. Son las clases pasivas que forman el modelo conceptual representado por el paquete << beans >>, permite el encapsulamiento de las clases.
- **Interfaz gráfica:** Son los elementos activos y dinámicos. Los cuales interactúan con el usuario. (Web-jsp).
- **Capa de Persistencia:** Permite realizar las transacciones con la Base de datos como crear, modificar y eliminar datos << dao >>.
- **Clase Principal:** Al ser una aplicación Web se utiliza la tecnología JSP la cual se encarga de realizar las llamadas a la GUI por medio de la página principal.

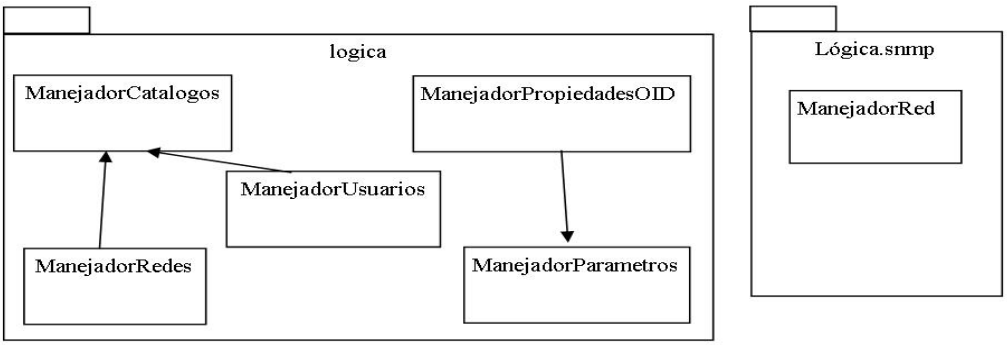
Para mostrar la arquitectura de la aplicación de forma visual utilizaremos la dependencia de paquetes tanto de sus capas, vista lógica y su despliegue visual. Pero para lo cual partiremos del dominio de la aplicación e iremos detallando cada estructura de cada paquete.

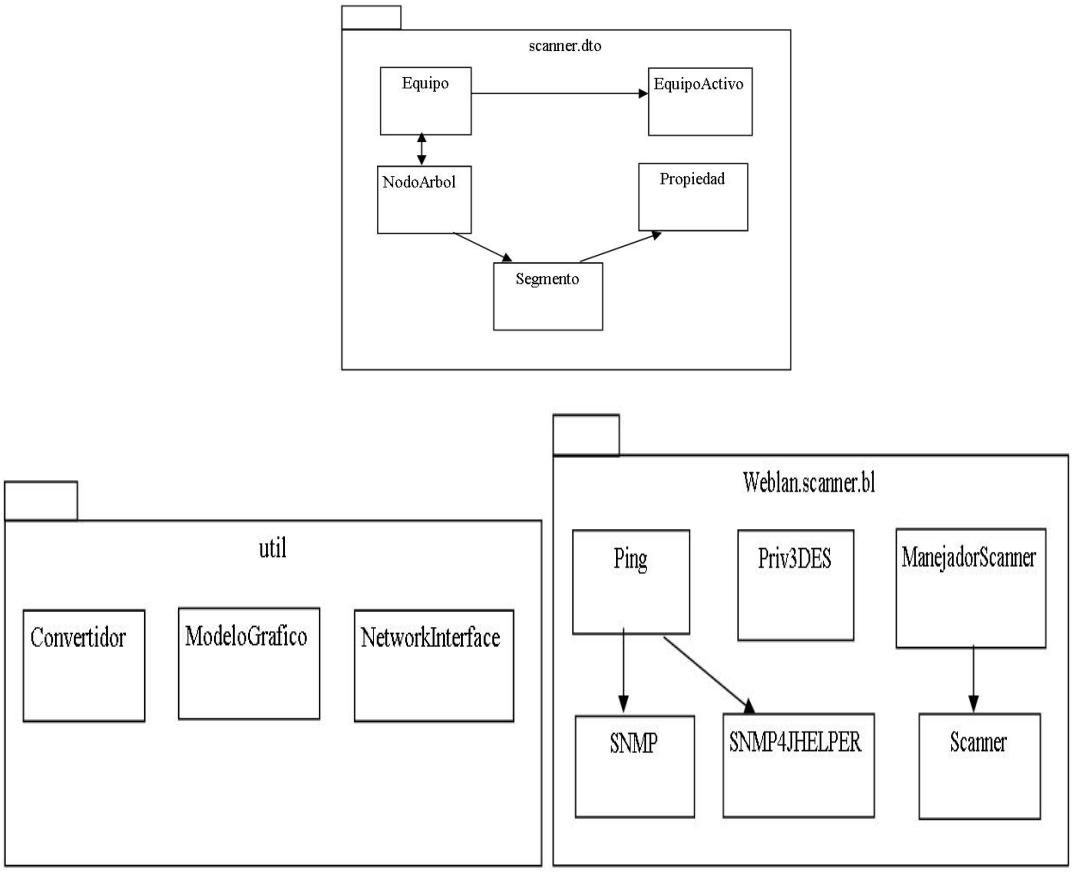
En el siguiente grafico se detalla el dominio de la aplicación, mostrando los paquetes que conforma el sistema Unlmonitoreo.



Para ir indicando la estructura de cada paquete, utilizaremos las clases que conforman el paquete y la relación entre sí:

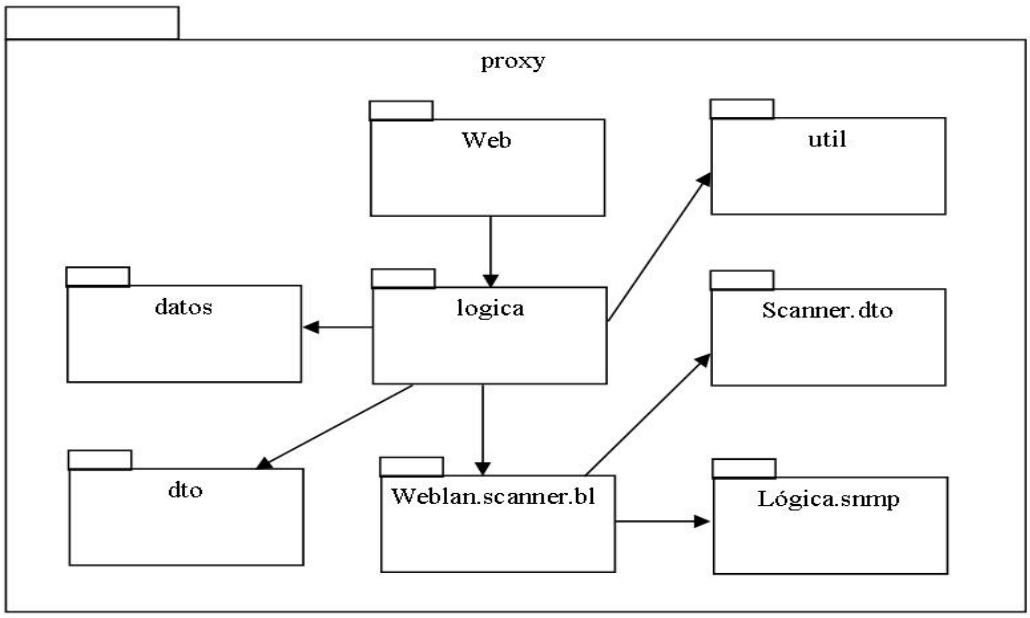




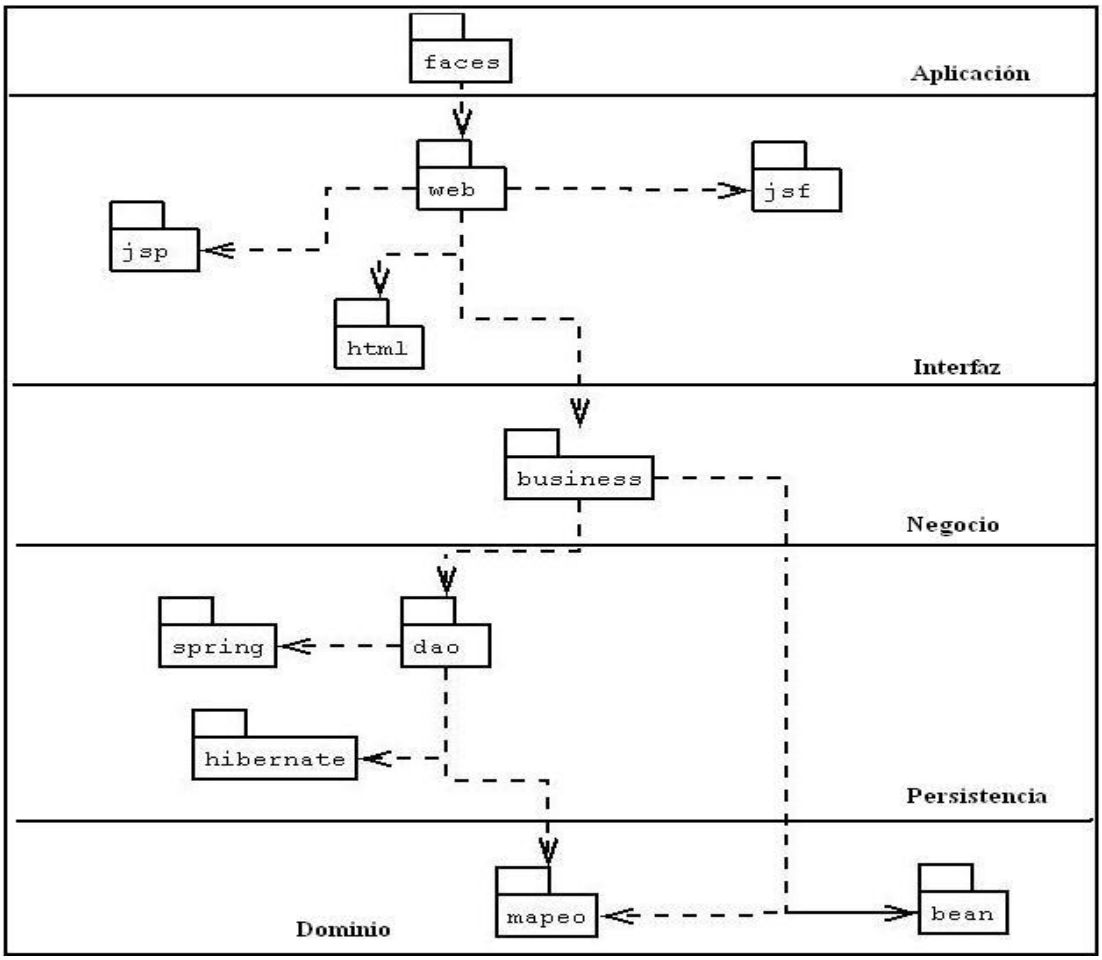


Las clases que conforman el dominio, necesitan ser persistidas por lo que cada clase del paquete << beans >>, tiene su correspondiente en el paquete << mapeo >>.

Mostrado de forma visual la estructura de cada paquete se procede a indicar la interacción entre ellos, detallando de una forma la dependencia de los mismo entre si. Como se detalla en la siguiente grafica:



La representación de la arquitectura de la aplicación indicando las diferentes capas de la misma se muestra en la siguiente figura:



En el proceso, describimos además la arquitectura del sistema dando una **vista lógica,** que muestra los componentes de alto nivel del sistema y las conexiones entre ellos. Los componentes principales de la arquitectura de nuestro sistema son:

- *Browser cliente*: la parte de la aplicación de los clientes necesita el browser para solicitar al servidor Web páginas *HTML* y *JSP*. La página devuelta por el servidor contiene un texto y controles que son enderezados por el browser y mostrados al cliente.

- *Servidor Web*: es el punto de acceso principal para los browsers de los clientes que acceden al sistema sólo a través del servidor Web. Dependiendo de la petición, el servidor Web puede iniciar algún tipo de procesamiento en el lado de servidor. Esto es lo que sucede con las páginas *JSP*.
- *Conexión HTTP*: es el protocolo utilizado entre los browser de los clientes y el servidor Web. Este elemento de la arquitectura representa un tipo de comunicación no orientado a la conexión entre clientes y servidor. Una alternativa de la conexión *HTTP* es usar "*HTTP* seguro", que es *HTTP* utilizando como protocolo de transporte *SSL (Secure Sockets Layer*).

- *Páginas HTML:* son todas las páginas Web con una interfaz para el cliente que no tienen procesamiento en el lado del servidor. Cuando el servidor Web recibe una petición de una página *HTML* simplemente recupera la página y la envía al cliente solicitante.

- *Página servidor*: son todas las páginas Web que de alguna forma realizan procesamiento en el servidor. Normalmente estas páginas residen en el servidor (Active Server Pages, Java Server Pages, etc.). Estas páginas, a diferencia de los applets, tienen acceso potencial a los recursos disponibles en el servidor incluyendo componentes de la lógica de negocio, bases de datos, etc.

- *Servidor de aplicación*: es la herramienta principal para ejecutar la lógica de negocio en el lado del servidor. Es el responsable de ejecutar el código de las páginas servidor. Utilizamos el servidor de aplicación Apache – Tomcat, aunque hay otros muchos disponibles en el mercado (Jrun, JSWDK, LWS, Sun's Java Web Server, etc.).

- *Base de datos*: utilizamos la base de datos para conseguir la persistencia de los datos del sistema de información. La base de datos utilizada es *MySQL* y el mecanismo usado para conectarla al sistema es *Java Database Connectivity* (*JDBC*).

- *Capa de correspondencia con la base de datos*: este componente nos proporciona un servicio de correspondencia entre los objetos y las tablas relacionales de la base de datos. Una explicación más detallada del funcionamiento de esta capa de correspondencia aparece cuando en la implementación hablamos de la interacción con la base de datos.

- *Script cliente*: usamos JavaScript embebido dentro de algunas páginas *HTML* para conseguir una interfaz mejorada, añadiendo funcionalidad como la validación de formularios, e incrementando el aspecto gráfico de las páginas.

CAPITULO IV

IMPLEMENTACIÓN DE LA APLICACIÓN

4. IMPLEMENTACION DE LA APLICACIÓN

4.1 Plataforma de desarrollo

Para el desarrollo de la siguiente aplicación se utilizo el entorno JDK de Java, debido principalmente a sus características de lenguaje, la aplicación está desarrollada para que funcione bajo el Sistema Operativo Linux o Windows.

La plataforma usada para el desarrollo de la aplicación es la siguiente:

Sistema Operativo

Linux Fedora Versión 11.

Lenguaje y entorno de programación

JSP, JSF, JDK 5.0-14

Herramientas Case para modelación del sistema

Poseidón for UML 4, DIA

Entornos de desarrollo integrados

Netbeans 6.5.1

Sistema de Base de datos

MySQL 5.0.51

Mapeador objeto – relacional

Hibérnate

Conector de Java para MySQL

JDBC MySQL – conector – java – 3.1.7. – bin

4.2 Política desarrollo de la aplicación

- Las Imágenes a utilizar tienen el formato de archivo gráfico .gif, .jpg, jsf y Woodstock (imágenes, iconos, animaciones y cajas campos).

- Debido a que la aplicación esta en entorno web, se hace necesario que los menú se desplieguen y tengan un formato html.

- Para el desarrollo de la aplicación se siguió algunos lineamientos de codificación: las clases comienzan con mayúsculas, seguida de minúsculas, si el entorno de la clase es compuesto por dos palabras, se sigue lo anterior con la diferencia que la primera letra de la segunda palabra es con mayúscula seguida de minúsculas.

4.3 Documentación de la aplicación

4.3.1 Guía de instalación

Para la instalación de Unlmonitoreo, debe contar su equipo con un Sistema Operativo Windows, Linux, ya sea Fedora u otro equivalente con sus últimas versiones. Por ejemplo para Windows en sus versiones Xp, Vista. Y para Linux puede ser Fedora en su versión 10 y 11, en Ubuntu Hardy Heron 8.04, y Ubuntu Jaunty 9.10. Que son en los que ha sido probada la aplicación.

4.3.2 Requerimientos mínimos de hardware

Características mínimas del equipo sobre el cual se ejecutara la aplicación es el siguiente:

- Procesador Intel Core 2 Duo de 2.8 GHz o su equivalente en otras máquinas.
- 2 GB de memoria Ram.
- 80 Gb de almacenamiento en disco
- 1 tarjetas de red 100/1000
- Monitor SVGA con una resolución de 1024 x 768

4.3.3 Requerimientos mínimos de software

El Computador necesita un mínimo de programas para que funcione y realice las tareas para las cuales fue adquirido, entre ellos snmp, MySQL, apache Tomcat y java. El software está compuesto principalmente por el Sistema Operativo, los lenguajes de comunicación y los programas de aplicación.

Los requerimientos de software para las máquinas con sistema operativo Linux deben cumplir con lo siguiente:

Apache Tomcat con sus respectivas dependencias.

- Java 5 Runtime Environment-JRE versión 1.5
- MySQL versión 5.0 con sus respectivas dependencias
- SNMP versión 2.0 con sus respectivas dependencias

4.3.4 Procedimientos de instalación de Unlmonitoreo

Para que el sistema funcione correctamente se debe instalar y configurar el paquete.

- **Apache-Tomcat version 6.0.24**

<u>Instalar Apache tomcat en Windows</u>

Apache-tomcat- version .exe
jdk- version -windows-i586-p.exe

Tenga en cuenta que el espacio requerido para la instalación del Tomcat es 11.4 MB

El puerto que se debe seleccionar para la conexión es el 8080 y el usuario de administración puede ser admin y password puede ser vació o la clave que desee

Para la instalación de la maquina virtual de java debemos hacerlo en la siguiente ruta C:\Program Files\Java\jdk version

Luego abrimos el explorador de Internet y le damos la siguiente ruta
http://localhost:8080/

Instalar Apache Tomcat en Linux

Descomprimimos la estructura:
tar xvfz apache-tomcat versión .tar.gz

y podemos moverla dentro de /usr/local/ como el caso de jdk
mv apache-tomcat versión /usr/local/tomcat versión

Ejecutamos tomcat con el comando:
/usr/local/apache-tomcat version /bin/startup.sh

Y asignara las variables de entorno dando como resultado:
Using CATALINA_BASE: /usr/local/apache-tomcat versión
Using CATALINA_HOME: /usr/local/apache-tomcat versión
Using CATALINA_TMPDIR: /usr/local/apache-tomcat versión /temp
Using JRE_HOME: /usr/local/jdk versión/jre

- **MySQL version 5.0**

La Base de datos es necesaria para que la aplicación pueda almacenar los datos.
Es preciso ejecutar el script adjunto a la aplicación denominado monitoreo.sql.

Instalación de MySQL en Windows

Ejecutaremos SETUP.EXE, en la próxima ventana, nos pide en que carpeta queremos instalar MySQL, nosotros dejaremos c:\mysql.

Ahora se realizará la copia de archivos y la instalación finalizará.

Abriremos una ventana de MS-DOS (Inicio > Ejecutar > cmd *) y escribiremos los siguientes comandos:

cd ... (Hasta que estemos en c)
cd mysql
cd bin
mysql –console

Mientras esta ventana esté abierta podremos conectar con MySQL.

Instalación de MySQL versión 5.0.51 en Linux

rpm -i MySQL- version i386.rpm

RPM hace lo siguiente para obtener MySQL para ejecutarlo en el sistema:

- Copia los ficheros binarios de MySQL a los lugares apropiados de su sistema (por lo general, los binarios de ir a / usr / bin y / usr / sbin, mientras que las bases de datos y las tablas se almacenan en / var / lib / mysql)

- Añade un usuario mysql / grupo al sistema para manejar todos los relacionados con el funcionamiento y las tareas administrativas de MySQL

- Altera la propiedad de los binarios de MySQL para que sean propiedad del usuario mysql / grupo

- Crea e inicializa la concesión tablas MySQL

- Añade entradas adecuadas para los scripts de arranque de su sistema para que el servidor MySQL se inicia automáticamente al arrancar

- Inicia el servidor para que pueda empezar a utilizarlo inmediatamente

Ahora instale los RPMs restantes de una manera similar:

- *Mysql-server-VERSION.* i386.rpm (Obligatorio)
- *mysqlclient9-VERSION.* i386.rpm (bibliotecas de objetos compartidos)
- *mysql-devel-VERSION.* i386.rpm (C incluyen los archivos y bibliotecas para desarrolladores de software)
- *php-mysql-VERSION.* i386.rpm (Para acceder a la base de datos de MySQL desdephp)

Para manipular el Unlmonitoreo debe estar siempre ejecutándose mysql y apache-tomcat.

4.4 Instalación de la Herramienta

- Ingrese a un Browser y en la Dirección ingrese la siguiente línea: http://localhost:8080 (8080 es el puerto donde se instala el Apache-Tomcat) y si todo es correcto se presentará la pagina de Apache-tomcat.

- Ingrese al Apache Tomcat y presione el botón examinar para buscar el archivo unlmonitoreo.war.

- Luego ejecute el botón “aceptar” y automáticamente este le indicará que el despliegue fue exitoso con el mensaje OK.

Si se desea conectar a la red desde otro equipo conectado, se debe cambiar localhost por la dirección IP del servidor y además de que el puerto que escucha apache-tomcat es diferente, debe ser modificado también, por defecto es el 8080.

- Sabiendo que es una aplicación web se debe solo copiar el archivo war dentro del directorio donde se encuentre instalado el weabpp de Tomcat

- Se debe ejecutar el script de la base de datos indicado anteriormente.

4.5 Ejecución de la aplicación

Una vez que se ha subido la herramienta al servidor se debe ejecutar desde cualquier browser:

http://localhost:8080/unlmonitoreo/

4.6 Programación o Código Fuente

El código de la aplicación para el Administrador como del Usuario se encuentra en el CD, adjuntado a este trabajo de Tesis.

4.7 Activación de SNMP Versión 2 para Monitoreo

Para habilitar el servicio de SNMP en Windows Xp o 2003

1.- En Windows Xp y Windows 2003, click en Inicio, luego en Panel de Control después en Agregar y quitar programas. Cuando se abra el cuadro de dialogo click en Agregar Componentes de Windows.

2.- En Windows XP y 2003, da click en Herramientas de Administración y supervisión y selecciona el Protocolo simple de administración de redes (SNMP).

3.- ahora solo damos aceptar en ok y esperamos que instale los servicios tal vez nos pida insertar el disco de instalación y reiniciar el equipo.

Para habilitar el servicio de SNMP en Windows Vista

1.- En Windows Vista, click en Inicio, luego en Panel de Control después Programas y Características después de eso a la izquierda de la pantalla da click en Activar o Desactivar características de Windows.

si sale un cuadro de dialogo da clic en Continuar.

En Windows Vista selecciona el Característica SNMP.

2.- Ahora solo damos aceptar en ok y esperamos que instale los servicios.

Configurar el Servicio de SNMP

1.- Damos click en Inicio luego Panel de Control **después** Herramientas Administrativas **después** abrimos la consola de **Servicios**.

2.- Localizamos el servicio llamado Servicio SNMP damos click derecho sobre él y después en Propiedades.

3.- Nos ubicamos en la pestaña de Capturas en Nombre de la comunidad pondremos “public” y después damos click al botón de Agregar a la lista.

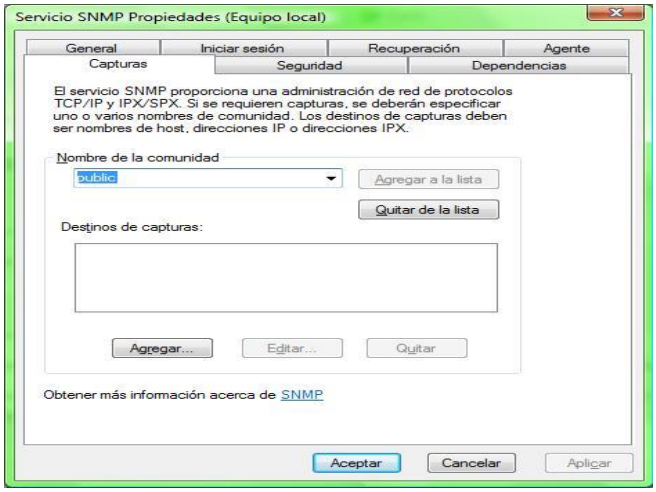


FIGURA 4.1 CONFIGURAR COMUNIDAD

4.- Ahora nos vamos a la pestaña de Seguridad en Nombre de comunidad aceptados agregamos public como SOLO LECTURA y seleccionamos Aceptar paquetes desde cualquier host.

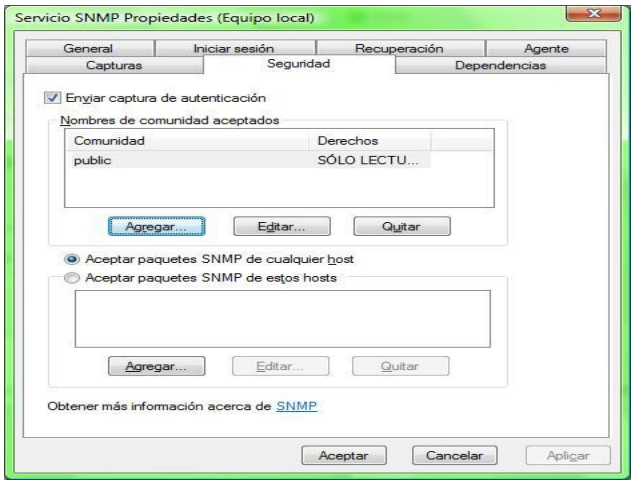


FIGURA 4.2 ACEPTAR PAQUETES DESDE CUALQUIER HOST

para que se guarden los cambios es necesario Reiniciar el equipo.

<u>Para habilitar el servicio SNMP en Linux (Fedora o Ubuntu)</u>

Software requerido:

yum -y install net-snmp net-snmp-utils

Para activar SNMP se debe:

1. configurar el nombre de comunidad en el archivo /etc/snmp/snmp.conf

```
Com2sec paranoid default        public
#com2sec readonly  default        public
Com2sec readonly   192.168.0.109  public
Com2sec readwrite default        private
```

Donde 192.168.0.109 es la IP desde donde se tendrá acceso al equipo

2. configurar el archivo /etc/default/snmpd para tener acceso desde otro equipo que no sea el local (127.0.0.1)

```
#SNMPDOPTS='-Lsd -Lf /dev/null -u snmp -I -smux -p /var/run/snmpd.pid 127.0.0.1'
SNMPDOPTS='-Lsd -Lf /dev/null -u snmp -I -smux -p /var/run/snmpd.pid'
```

CAPITULO V

VALIDACIÓN DEL SISTEMA

5.1. DISEÑO DEL PLAN DE PRUEBA

El objetivo primordial de realizar un plan de pruebas es para identificar las falencias y limitaciones que pueden existir en el Software Unlmonitoreo, mediante la aplicación de este plan realizaremos la corrección de algunos errores que se presenten en el momento de ejecutar la aplicación desarrollada y presentar un software sin errores.

5.2. PERSONAL SELECCIONADO PARA VALIDAR LA APLICACIÓN

El software de monitoreo **Unlmonitoreo,** tiene las características para ser implementado sobre un servidor administrador de una red y su objetivo es facilitar la administración de la red y verificar el buen funcionamiento de la misma, y para que una persona pueda validar el software es necesario que sea administrador o jefe de cómputo, o tenga un cargo a fin.

Por esta razón, se ha seleccionado al **Ing. Patricio Villamarín** quien ahora tiene a su cargo la Coordinación de la **U.D.I.** para hacerle la encuesta sobre la funcionalidad de la aplicación ya que cumple con la función de Administrador de la red del Área.

La encuesta para la validación del software “Unlmonitoreo” se fundamentó en las siguientes preguntas:

1. Tuvo algún problema para ingresar remotamente al software “unlmonitoreo”

   Si ☐

   No ☐

Porque_____
_____
_____

2. La interfaz del programa “unlmonitoreo” le parece amigable.

   Si ☐

   No ☐

Porque_____
_____
_____

3. Del menú Administrar pudo crear, modificar, eliminar en Usuarios, Parámetros y Catálogos

Si ☐

No ☐

Porque_____
_____
_____

4. Del menú Herramientas pudo escanear los equipos de la Red?

Si ☐

No ☐

Porque_____
_____
_____

5. La opción de Configurar le parece adecuada para adaptarse a nuevos requerimientos?

Si ☐

No ☐

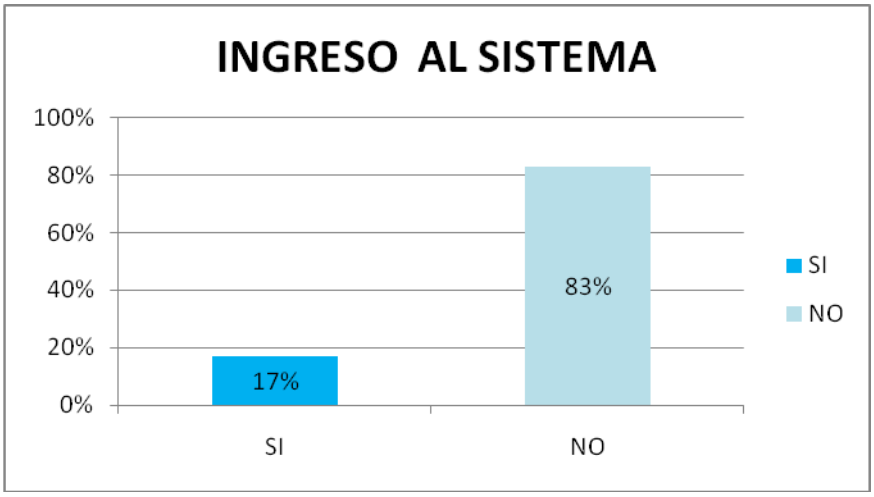
Porque_____
_____
_____

PRUEBA DE VALIDACIÓN APLICADA A LOS DOCENTES

(Ing. ketty Palacios, Patricio Villamarín, Edison Coronel, Lorena Conde, Hernán Torres, Lic. Cecilia Zúñiga) DE LA CARRERA DE INGENIERIA EN SISTEMAS.

INTERPRETACIÓN DE RESULTADOS

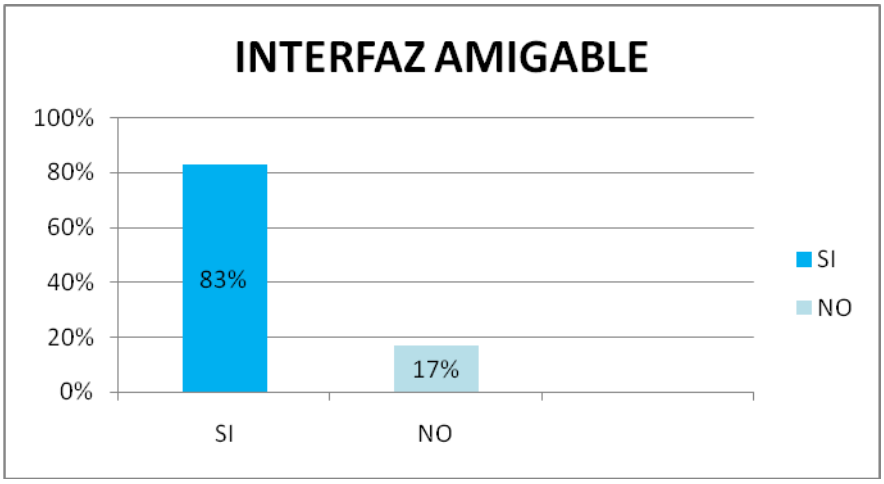
1. Tuvo algún problema para ingresar remotamente al software "unlmonitoreo"



Interpretación:

Con respecto a la pregunta se puede comentar que el 17% de los encuestados manifestaron que al ingresar remotamente al SOFTWARE este presente una dificultad leve que fue corregida de inmediato, y un 83% no tuvo ningún problema al ingresar, esto demuestra que el programa se ha ejecutado con una relativa normalidad.

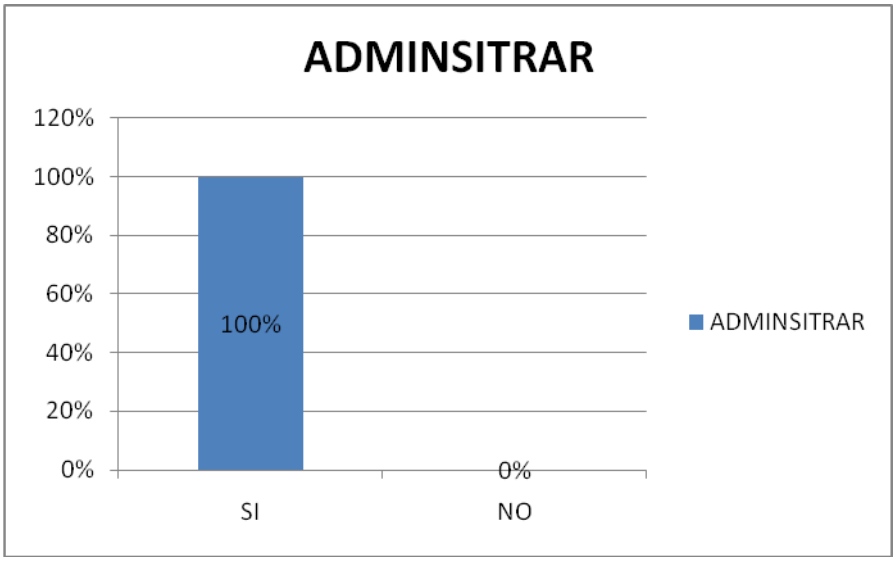
2. La interfaz del programa "unlmonitoreo" le parece amigable.



Interpretación:

Con relación a esta pregunta podemos observar que el 17% ha respondido que la Interfaz se podría mejorar, mientras que el 83% según los encuestados se manifiestan que esta es amigable.

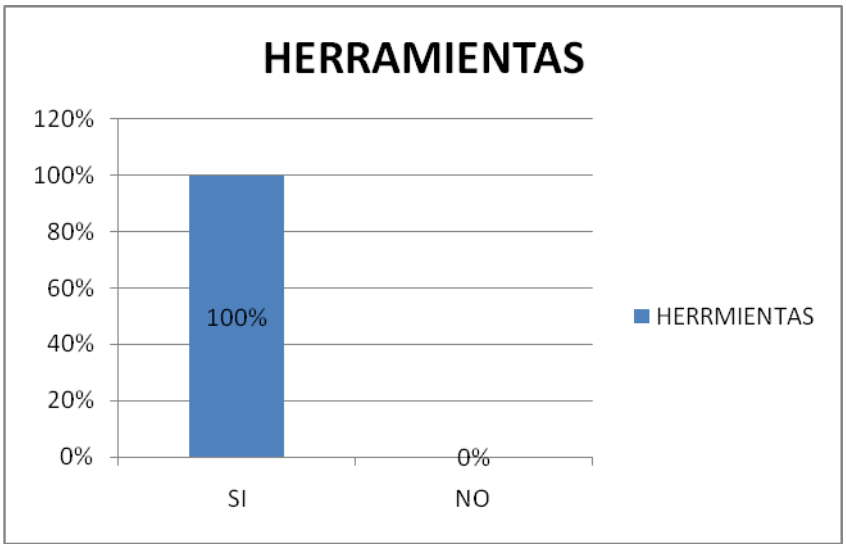
3. Del menú Administrar pudo crear, modificar, eliminar en Usuarios, Parámetros y Catálogos



Interpretación:

En esta pregunta un 100% de los encuestados se manifestaron que es fácil administrar, usuarios, parámetros y catálogos, pero se podría mejorar los mensajes

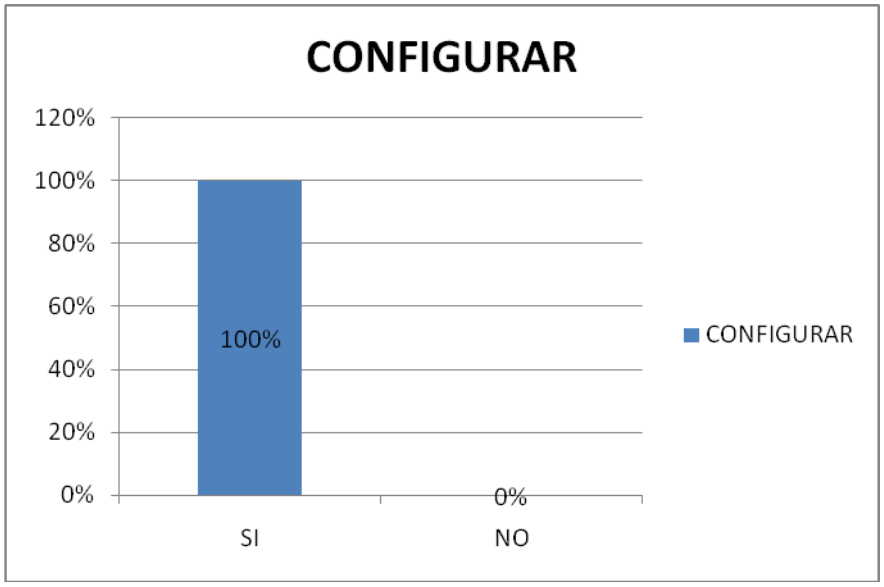
4. Del menú Herramientas pudo escanear los equipos de la Red?



Interpretación:

En relación a esta pregunta un 100% de los encuestados ha manifestado que si se ha podido escanear los equipos de la red con facilidad.

5. La opción de Configurar le parece adecuada para adaptarse a nuevos requerimientos?



Interpretación:

La pregunta 5 tiene como respuesta un 100%, la cual manifiesta que si es fácil la Configuración del SOFTWARE, por lo tanto para manejarlo este no presenta mayor dificultad.

Después de haber aplicado las encuestas los usuarios hicieron las siguientes recomendaciones:

- Con respecto a la segunda pregunta que se podría mejorar en la interfaz del programa, faltan mensajes que indiquen al usuario que debe realizar.

- En la tercera pregunta en modificar un usuario del sistema no muestra ningún mensaje que debe llenar todos los campos para poder guardar.

VI

CONCLUSIONES

Y

RECOMENDACIONES

6.1. CONCLUSIONES

- El software unlmonitoreo realiza el análisis del tráfico de la red desde cualquier computador debido a que todos cuentan con navegador y el sistema desarrollado soporta cualquier Sistema Operativo.

- El software unlmonitoreo nos permite conocer las características de los equipos analizados como su ip, nombre, uso de cpu y otros utilizando las oid que se ingresan en el sistema.

- Esta aplicación permite a los estudiantes identificar las variaciones en el tráfico de la red de acuerdo a las funciones que realice cada computador.

- Este software optimiza el análisis de los equipos ya que no es necesario ir a cada computador a monitorear su tráfico si no que se lo puede realizar por esta aplicación.

- Según los resultados del análisis del tráfico obtenido en el sistema, el administrador reduce el ancho de banda para un determinado usuario si lo cree conveniente.

- El software unlmonitoreo basado en el protocolo snmp interactúa sin ningún problema con dispositivos que utilizan el sistema operativo Windows o Linux.

- El software desarrollado soporta las versiones 1 y 2 de snmp, permitiendo el monitoreo de los equipos que soporten las versiones de este protocolo.

6.2. RECOMENDACIONES

- El servidor principal donde va a correr la aplicación debe tener instalado MySQL versión 5.0 o superior, Apache Tomcat versión 6.0.24 o superior para su correcto funcionamiento.

- Es recomendable que todas las computadoras a ser monitoreadas tengan habilitadas el servicio de SNMP para conocer sus características.

- Se recomienda herramientas de software libre con el propósito de reducir los costos que permiten un control eficaz del monitoreo de red.

- El protocolo SNMP de la versión uno tiene problemas de seguridad ya que trabaja con la comunidad “public”, mientras que en la versión snmp dos se tendrá un cierto grado de seguridad autenticándose a través del nombre de la comunidad pública y privada, por lo que es recomendable trabajar con la versión 2 o posteriores.

- Se recomienda que todos los elementos de red a ser monitoreados soporten el protocolo SNMPv2, puesto que permitirá monitorear con el software unlmonitoreo.

- Se recomienda que el software unlmonitoreo se implemente en el UDI (Unidad de Telecomunicaciones de Información) porque es la base donde se encuentra el servidor principal de todas las redes.

BIBLIOGRAFIA

Libros:

- Moliner López, Francisco Javier. 2005. Informáticos Generalitat Valenciana. España, Apress. 138

- Chávez Ruíz, Grisel; Rivera Lozano, Jesús Enrique; Osornio Valdez, Salvador 2005. Análisis del Protocolo SNMP.360

- Stallings, William. 1996. SNMP, SNMPv2, SNMPv3, and RMON 1 and 2. 1996. 3 Edición. 619

- Stallings, William. 2004. Fundamentos de Seguridad en Redes Aplicaciones y Estándares. Madrid, Apress. 432

Sitios Web:

- **URL:** cisco.com/en/US/docs/internetworking/technology/handbook/SNMP.html
  Descripción: Protocolo Simple de Administración de Red (SNMP) [consulta, 6 de Junio del 2007]

- **URL:** coit.es/publbit/bit102/quees.htm
  Descripción: J. Manuel Huidrovo, Snmp. Un Protocolo Simple de Gestión, [consulta, 24 de Octubre del 2006]

- **URL:** david-guerrero.com/papers/snmp/lj.es.html
  Descripción: Administración de Redes con Linux [consulta, 10 de Noviembre de 2009]

- **URL:** es.kioskea.net/contents/internet/protip.php3
  Descripción: Protocolo IP [consulta, 19 de Noviembre del 2007]

- **URL:** es.kioskea.net/contents/internet/tcp.php3
  Descripción: Protocolo TCP [consulta, 29 de Noviembre del 2008]

- **URL:** es.kioskea.net/contents/internet/tcpip.php3
  Descripción: El modelo Osi [consulta, 7 de Junio del 2009]

- **URL:** hibernate.org
  Descripción: Pagina oficial del framework Hibernate [consulta, 6 de Julio del 2007]

- **URL:** mailxmail.com/curso-que-son-redes/que-es-tcp-ip
  Descripción: Que son las Redes [consulta, 23 de Febrero del 2007]

- **URL:** neutron.ing.ucv.ve/revista-e/No6/BriceñoMaria/SNMPv3.html
  Descripción: SNMPV3 [consulta, 26 de Noviembre del 2008]

- **URL:** Programación.net/java/tutorial/jsf-intro/
  Descripción: información sobre JSF [consulta, 6 de Junio del 2006]

- **URL:** radioutn.org.ar/comunicaciones/default.html
  Descripción: TANNEMBAUN, Redes Para Ordenadores [consulta, 23 de Marzo del 2008]

- **URL:** timat.unican.es/siteadmin/submaterials/89.pdf
  Descripción: Formato de las Tramas SNMP [consulta, 18 de Julio del 2009]

ANEXOS

ANEXO A

CERTIFICACIÓN DE LA VALIDACION DEL SOFTWARE

UNIVERSIDAD NACIONAL DE LOJA
AEIRNNR

Loja, 27 de Mayo de 2010

CERTIFICA:

Que el señor ROLANDO ANTONIO VIÑAN RIOFRIO, Egresado del la carrera de Ingeniería en Sistemas instalo el software **Herramienta de Monitoreo para la Red de la Universidad Nacional de Loja,** en la Unidad de Desarrollo Informático el cual se encuentra ejecutándose y escaneando en la red con total normalidad.

Es todo cuanto puedo certificar en honor a la verdad, facultándolo al interesado hacer uso del documento.

Ing. Germán Patricio Villamarín Coronel
RESPONSABLE DE LA UNIDAD DE DESARROLLO INFORMATICO

ANEXO B

ENCUESTA DE LA VALIDACION DE SOFTWARE

Universidad Nacional de Loja
Área de la Energía, las Industrias y los Recursos Naturales no Renovables.

Cuestionario de validación sobre la implementación y manejo del software "unlmonitoreo" implementado para la Red de la UNL.

1. **Tuvo algún problema para ingresar remotamente al software "unlmonitoreo"**

Si ☐

No ☐

Porque_____
_____
_____

2. **La interfaz del programa "unlmonitoreo" le parece amigable.**

Si ☐

No ☐

Porque_____
_____
_____

3. **Del menú Administrar pudo crear, modificar, eliminar en Usuarios, Parámetros y Catálogos**

Si ☐

No ☐

Porque_____
_____
_____

4. **Del menú Herramientas pudo escanear los equipos de la Red?**

Si ☐

No ☐

Porque_____
_____
_____

5. **La opción de Configurar le parece adecuada para adaptarse a nuevos requerimientos?**

Si ☐

No ☐

Porque_____
_____
_____

Gracias por su colaboración.

Nombre _____
Cargo _____
Firma _____

ANEXO C

PAQUETE SNMPJ4

SNMP4J - El objeto SNMP API orientada a Java para gestores y agentes

SNMP4J es una clase empresarial de código abierto libre y de tecnología de punta aplicación SNMP-estado para Java ™ 2SE 1,4 o posterior. SNMP4J compatible con la generación de comandos (gestores), así como comandos de responder (agentes). Su objeto de diseño orientado a limpiar se inspira en SNMP + +, que es un conocido API SNMPv1/v2c/v3 para C + + (ver http://www.agentpp.com).

El SNMP4J Java SNMP API ofrece las siguientes características:

- SNMPv3 con MD5 y autenticación SHA y DES y AES 128, AES 192 y AES 256 privacidad.
- Pluggable modelos de procesamiento de mensajes con las implementaciones de MPv1, MPv2c y MPv3
- Todos los tipos de PDU.
- Pluggable asignaciones de transporte UDP y TCP cuentan con el apoyo fuera de la caja.
- Pluggable modelo de tiempo de espera.
- Sincrónicas y asincrónicas peticiones.
- Comando generador, así como el apoyo de respuesta de comando.
- Libre de código abierto con la licencia de Apache modelo
- Java ™ 1.4.1 o posterior
- Inicio de sesión basada en Log4J
- Fila asincrónica basada eficiente de recuperación de la tabla con GetBulk.
- El soporte multi-threading.
- pruebas JUnit (estará disponible en la versión 2.x y posterior)

API and comes with: El SNMP4J-Agente Java puro agente SNMP API añade comandos de respuesta incluyendo la notificación iniciador y el apoyo reenviador al núcleo SNMP4J API y viene con:

- Las implementaciones de SNMP-MIB-TARGET, SNMP-MIB-NOTIFICACIÓN, SNMP-MIB PROXY, SNMP-MIB-MARCO, SNMPv2-MIB, de comunidad SNMP-MIB, SNMP-Basado en el usuario--MIB SM, SNMP-VIEW- BASE-ACM-MIB, SNMP y MIB--MPD.

- SNMPv1, v2c, v3 lingual como agente de apoyo multi-, incluyendo la autenticación MD5 y SHA, así como DES, 3DES y AES (128, 192, 256) privacidad.
- IPv4/IPv6 UDP and TCP support. IPv4/IPv6 UDP y TCP de apoyo.
- La generación de código a partir de especificaciones MIB se proporciona a través AgenPro 2 que es un lenguaje y API independiente generador de código de la plantilla base con la generación de las instalaciones de viaje de todo el año.

El SNMP4J-AgentX API Java puro AgentX añade soporte para el maestro y 1,0 AgentX subagente del protocolo definido por RFC 2741 y 2742. SNMP4J-AgentX extiende SNMP4J-Agente:

- Lleno AgentX 1,0 compatibilidad con el protocolo, incluyendo los contextos, para compartir las tablas, la asignación de índice, PDU ping, manejo de tiempo de espera de conexión, etc.
- Aplicación de la AgentX-MIB para el agente principal.
- Mapeo de transporte TCP para el protocolo AgentX.
- La generación de código a partir de especificaciones MIB se proporciona a través AgenPro 2 que es un lenguaje y API independiente generador de código de la plantilla base con la generación de las instalaciones de viaje de todo el año (véase también SNMP4J-agente).

El SNMP4J-AgentJMX puro Java 1.5 añade soporte API preparación de mapas descriptivos de JMX MBean instrumentación a escalares SNMP, mesas, y las notificaciones.

org.snmp4j

Proporciona clases e interfaces para crear, enviar y recibir mensajes SNMP.

El org.snmp4j clases son capaces de crear, enviar y recibir mensajes SNMPv1/v2c/v3.. Un mensaje SNMP está compuesta por su encabezado del mensaje y su carga útil PDU. Este paquete contiene tres grandes grupos de clases e interfaces:

- Las clases de mensaje SNMP y la creación de meta
- Las clases para el envío de mensaje SNMP (generación de comandos)

- Las clases para el envío de mensaje SNMP (comando de responder)

El siguiente paquete de diagrama UML muestra las dependencias entre los paquetes del núcleo SNMP4J API. Los usuarios de la API normalmente sólo se necesitan utilizar la org.snmp4j y el org.snmp4j.smi paquetes directamente.

La siguiente clase muestra el diagrama UML clases más importantes del paquete org.snmp4j y sus relaciones (relaciones con otros paquetes no se muestran):

Mensajes SNMP y metas

Facilitar el intercambio de un mensaje SNMP con un sistema remoto, que el sistema tenga que ser identificado, la retransmisión y la política de información de tiempo de espera sobre el intercambio de mensajes tiene que ser especificado. Un sistema remoto se especifica con SNMP4J mediante la creación de un Target instancia adecuada para el protocolo SNMP para ser utilizado.

- Para SNMPv1 y SNMPv2c la CommunityTarget se tenga que utilizar que proporciona información de la comunidad, además de la dirección, la retransmisión y la política de información de tiempo de espera definido por el Target de la interfaz.
- Para el SNMPv3 UserTarget debe ser utilizado en su lugar.. Se extiende la SecureTarget clase abstracta y establece lo siguiente de usuario basada en el Modelo de Seguridad (USM) la información del usuario: nombre de la seguridad, nivel de seguridad, modelo de seguridad (es decir, USM) y motor autoritativo ID.

Un mensaje SNMP consiste en la carga útil del mensaje, el SNMP Protocolo de la Unidad de datos (PDU) y un encabezado del mensaje. Simplificado, dijo, en SNMP4J la información del encabezado del mensaje está representado por Target instancias y la PDU es representada por una de las siguientes clases:

- PDUv1 (SNMPv1)
- PDU (SNMPv2c)
- ScopedPDU (SNMPv3)

Así, con el fin de poder enviar un mensaje SNMP con SNMP4J, un PDU de instancia y un Target instancia tiene que ser creado.

Ejemplos PDU

- SNMPv1/v2c PDU GETNEXT
-
- import org.snmp4j.PDU;
- import org.snmp4j.smi.*;
- ... ...
- PDU pdu = new PDU();
- pdu.add(new VariableBinding(new OID("1.3.6.1.2.1.1.1"))); // sysDescr
- pdu.add(new VariableBinding(new OID("1.3.6.1.2.1.2.1"))); // ifNumber
- pdu.setType(PDU.GETNEXT);
- ... ...

  SNMPv3 GETBULK PDU PDU GetBulk SNMPv3
-
- import org.snmp4j.ScopedPDU;
- import org.snmp4j.smi.*;
- ... ...
- ScopedPDU pdu = new ScopedPDU();
- pdu.add(new VariableBinding(new OID("1.3.6.1.2.1.2.1"))); // ifNumber
- pdu.add(new VariableBinding(new OID("1.3.6.1.2.1.2.2.1.10"))); // ifInOctets
- pdu.add (new VariableBinding(new OID ("1.3.6.1.2.1.2.2.1.16 "))); / / ifOutOctets
- pdu.setType(PDU.GETBULK);
- pdu.setMaxRepetitions(50);
- // Get ifNumber only once
- pdu.setNonRepeaters(1);
- / / Contexto que se define no por defecto de contexto (contexto predeterminado no es necesario establecer)
- pdu.setContextName(new OctetString("subSystemContextA"));
- / / Establecer el contexto no predeterminado IDENTIFICACION del motor (para usar objetivos motor autoritativo ID
- / / Uso de una biblioteca (vacía tamaño == 0) cadena de octeto)

- pdu.setContextEngineID(OctetString.fromHexString("80:00:13:70:c0:a8:01:0d"));
- ... ...

- SNMPv1 TRAP PDU SNMPv1 TRAP PDU
-
- import org.snmp4j.PDUv1;
- ... ...
- PDUv1 pdu = new PDUv1();
- pdu.setType(PDU.V1TRAP);
- pdu.setGenericTrap(PDUv1.COLDSTART);
- ... ...

- SNMPv2c/SNMPv3 INFORM PDU SNMPv2c/SNMPv3 INFORM PDU
-
- import org.snmp4j.ScopedPDU;
- ... ...
- ScopedPDU pdu = new ScopedPDU();
- pdu.setType(PDU.INFORM);
- // sysUpTime
- long sysUpTime = (System.currentTimeMillis() - startTime) / 10; pdu.add(new VariableBinding(SnmpConstants.sysUpTime, new TimeTicks(sysUpTime)));
- pdu.add(new VariableBinding(SnmpConstants.snmpTrapOID, SnmpConstants.linkDown));
- // Carga útil
- pdu.add(new VariableBinding(new OID("1.3.6.1.2.1.2.2.1.1"+downIndex),
- new Integer32(downIndex)));

Ejemplos de destino

- Community Target Comunidad de destino
-
- CommunityTarget target = new CommunityTarget();
- target.setCommunity(new OctetString("public"));

- target.setAddress(targetAddress);
  target.setVersion(SnmpConstants.version1);

Objetivo del usuario

- UserTarget target = new UserTarget();
- target.setAddress(targetAddress);
- target.setRetries(1);
- // establece en 500 milisegundos -> 2 * 500 ms = 1s total de tiempo de espera
- target.setTimeout(500);
- target.setVersion(SnmpConstants.version3);
- target.setSecurityLevel(SecurityLevel.AUTH_PRIV);
- target.setSecurityName(new OctetString("MD5DES"));

Envío de mensajes SNMP

Mensajes SNMP se envían con SNMP4J utilizando una instancia de la SNMP Sesión interfaz. La implementación predeterminada de esta interfaz es la Snmp clase.

Para configurar un Snmp ejemplo es suficiente para llamar a su constructor con un TransportMapping instancia. El mapeo de transporte es utilizado por el período de sesiones SNMP a enviar (y recibir) mensajes SNMP a un sistema remoto mediante un protocolo de transporte, por ejemplo, el Protocolo de datagramas de usuario (UDP).

Un SNMP4J Snmp ejemplo soporta SNMP v1, v2c y v3 por defecto. Por sub-clasificar Snmp otras combinaciones de las versiones del protocolo SNMP se puede apoyar.

Con SNMP4J, mensajes SNMP pueden ser enviadas *de forma sincrónica* (bloqueo) y *asíncrona* (no-bloqueo). El Snmp clase no utiliza una rosca interna para procesar las respuestas de solicitudes asincrónicas y sincrónicas. No obstante, utiliza los hilos del receptor de las asignaciones de transporte para procesar las respuestas.

Respuestas asincrónicas son devueltas por llamar a un método de devolución de llamada en una instancia de objeto que implementa la ResponseListener interfaz. La devolución de llamada se lleva a cabo en nombre de la rosca de la cartografía de transporte que recibió el paquete de respuesta del cable. Así, si el método llamado bloques, la entrega de mensajes síncronos y asíncronos recibidos en el puerto de

escucha de que la cartografía de transporte también serán bloqueados. Otros mapas de transporte no se verán afectados. El bloqueo se puede evitar, ya sea por medio de mensajes síncrono o sólo por la disociación del tratamiento en el método de devolución de llamada.

Ejemplo para el envío de un mensaje síncrono

```
import org.snmp4j.*;
... ...
Snmp snmp = new Snmp(new DefaultUdpTransportMapping());
... ...
ResponseEvent response = snmp.send(requestPDU, target);
if (response.getResponse() == null) {
   // Tiempo de espera agotado
   ... ...
}
else {
   System.out.println("Received response from: "+ response.getPeerAddress
   // Vuelca respuesta PDU
   System.out.println(response.getResponse().toString());
}
```

Ejemplo para el envío de un mensaje asincrónico

```
import org.snmp4j.*;
import org.snmp4j.event.*;
... ...
Snmp snmp = new Snmp(new DefaultUdpTransportMapping());
... ...
ResponseListener listener = new ResponseListener() {
   public void onResponse(ResponseEvent event) {
      PDU response = event.getResponse();
      PDU request = event.getRequest();
      if (response == null) {
         System.out.println("Request "+request+" timed out");
      }
      Else {
```

```
        System.out.println("Received response "+response+" on request "+
    }
};
snmp.sendPDU(request, target, null, listener);
... ...
```

Recepción de mensajes SNMP

SNMP4J recibe mensajes SNMP a través del puerto de escucha de las asignaciones de transporte. Con el fin de poder recibir las respuestas y solicitudes, que el puerto se debe establecer en modo de escucha. Esto tiene que hacerse mediante una llamada al listen () método de la TransportMapping ejemplo para iniciar las asignaciones de transporte interno escuchar hilo. La rosca interior se detiene y el puerto de escucha está cerrada mediante una llamada al close () método en el TransportMapping instancia o el correspondiente Snmp instancia.

El mapeo de transporte sólo recibe el mensaje SNMP como un flujo de bytes y reenvía el mensaje a asociados Message Dispatcher casos. De forma predeterminada, SNMP4J utiliza una instancia de la MessageDispatcherImpl clase para descifrar y envío mensajes entrantes. That instance is created and used internally by the Snmp class. Esa instancia es creada y utilizada internamente por el Snmp clase.

El Snmp clase procesos respuestas a las solicitudes pendientes y las PDU por delante de otros mensajes SNMP para registrados CommandResponder casos oyente. Para recibir mensajes SNMP por lo que es suficiente para:

1. Crear un TransportMapping e inicializar su puerto de escucha llamando TransportMapping. Listen ().
2. Crear un Snmp ejemplo de lo anterior TransportMapping.
3. Crear instancias de una clase que implementa la CommandResponder interfaz y registrarlo con la Snmp instancia llamando Snmp.addCommandResponder (CommandResponder).

Cuando un mensaje SNMP no controlada (por lo tanto un mensaje SNMP donde no existe solicitud pendiente correspondiente) se recibe, entonces el processPdu(CommandResponderEvent) método de la CommandResponder será llamado con el PDU decodificado e información adicional acerca del mensaje recibido

SNMP proporcionado por el procesamiento de mensajes modelo que ha descifrado el mensaje SNMP.

Ejemplo para recibir mensajes de SNMP

```
import org.snmp4j.*;
import org.snmp4j.smi.*;
import org.snmp4j.mp.SnmpConstants;
... ...
TransportMapping transport =
   new DefaultUdpTransportMapping(new UdpAddress("0.0.0.0/161"));
Snmp snmp = new Snmp(transport);
if (version == SnmpConstants.version3) {
   byte[] localEngineID

((MPv3)snmp.getMessageProcessingModel(MessageProcessingModel.MPv3)).createL
ocalEngineID();
USMusm=newUSM(SecurityProtocols.getInstance(),
OctetString(localEngineID), 0);
   SecurityModels.getInstance().addSecurityModel(usm);
   snmp.setLocalEngine(localEngineID, 0, 0);
   // Añadir el usuario configurado en la USM
   ... ...
}
snmp.addCommandResponder(this);
transport.listen();
... ...
public synchronized void processPdu(CommandResponderEvent e) {
   PDU command = e.getPdu();
   if (command != null) {
   ... ...
   }
 }
```

ANEXO D

ANTEPROYECTO DE TESIS

UNIVERSIDAD NACIONAL DE LOJA
AEIRNNR

UNIVERSIDAD NACIONAL DE LOJA

AREA DE ENERGIA, INDUSTRIAS Y RECURSOS NATURALES NO RENOBABLES

TEMA: HERRAMIENTA DE MONITOREO PARA LA RED DE LA UNIVERSIDAD NACIONAL DE LOJA.

AUTOR: ROLANDO VIÑAN

Coordinador:

LOJA –ECUADOR

2004

2. PLANTEAMIENTO DEL PROBLEMA

La seguridad de la red es uno de los factores más importantes que cualquier administrador o instalador de red. Debe tener en cuenta pero así mismo proporcionara todas las facilidades para que un usuario acceda a los datos codificando y decodificando con facilidad los mensajes sin intervención de terceras personas.

En las instalaciones de red los cambios son frecuentes, especialmente en su cableado, debido a la evolución de los equipos y a las necesidades de los usuarios de la red. Es así mismo que las redes en general consisten "compartir recursos", y uno de sus objetivos es hacer que todos los programas, datos y equipo estén disponibles para cualquiera de la red que así mismo lo solicite, sin importar la localización física del recurso y del usuario.

Por ello en el año de 1990 la UNIVERSIDAD NACIONAL DE LOJA específicamente en la parte administrativa se crea la Red General para bibliotecas que tenía una arquitectura ARNET. Su protocolo de comunicación era SPX/IPX. Luego a partir de 1998 se empezó a trabajar en el diseño del Backbone Universitario el cual tiene el objetivo de conectar a todas.

La Red con la que cuentan es FastEthernet a 100mbps, el protocolo de comunicación es el TCP/IP, el enlace con el que cuentan es satelital, su topología es estrella y su ancho de banda es de 512 de alta y 256 de baja, a demás se encuentran trabajando con una red LAN.

El servidor principal de Administración Central cuenta con un software de ayuda que lo obtuvo en el Internet llamado Visual Router, que le permite mostrar una tabla de ruteo como:
Tiempo en el que llega a su destino,
Ver si los servidores de las otras áreas están encendidos o apagados
Pérdidas de paquetes.
También la recopilación de datos para la elaboración de distintos tipos de estadísticas.
Verificar el ancho de banda de entrada / salida a una zona de la red entre los distintos tipos de protocolos de nivel de aplicación o entre las distintas máquinas de la zona en sus accesos externos.

La implementación de un software propio es necesario pues nos ayuda con el monitoreo de la red en cada una de las Áreas, porque el que tiene se caduca al mes, y su adquisición es muy costosa.

El monitoreo debe ser remoto y de tiempo real, ejecutado desde Administración Central con un sofisticado Centro de Monitoreo. Todo este proceso se ejecutará por medio de una conexión segura y con absoluta garantía de confidencialidad.
La red de datos tiene como propósito principal servir en la transformación e intercambio de información entre Organizaciones Académicas y de Investigación, mediante éstos y otros servicios tanto local, nacional e internacional, a través de conexiones con otras redes

El personal del centro de cómputo es el encargado de suministrar medidas de seguridad contra los intrusos o daños a la información almacenada en los sistemas, como la instalación de cualquier herramienta, dispositivo o versión de software.

El Centro de Operaciones es el único autorizado para monitorear constantemente el tráfico de paquetes sobre la red, con el fin de determinar y solucionar anomalías

Es por eso que para determinar el problema que hemos encontrado en Administración Central este ha tenido un seguimiento y un proceso el mismo que cuenta también con soluciones claras y precisas detallándolo al problema así:

UNA HERRAMIENTA DE MONITOREO PARA LA RED DE LA UNIVERSIDAD NACIONAL DE LOJA.

3. MARCO TEORICO REFERENCIAL

Las herramientas informáticas en redes, han dado un cambio radical en el mundo de las telecomunicaciones, por sus avances tecnológicos tanto en hardware como en software. Por tanto para elaborar una “Herramienta de monitoreo en la Red”, se debe tener conocimientos teóricos que ayuden a entender la evolución tecnológica en el presente tema. Es por ello que la evolución de las telecomunicaciones se la hizo a través de signos abstractos, dibujos en papel realizado en hojas de árboles, señales de fuego, maratón, el código Morse para telegrafía, telégrafo por cable, luego el teléfono, la radio, y la televisión

Después en 1969 nace el INTERNET gracias al desarrollo de la red de computadoras y a la industria de los ordenadores (computadores) que ha crecido sin precedente, así mismo la puesta en órbita de los satélites de comunicación.

Estos sistemas, se conocen con el nombre de redes de ordenadores. Estas nos dan a entender una colección interconectada de ordenadores autónomos. Se dice que los ordenadores están interconectados, y son capaces de intercambiar información. Las redes en general, consisten en "compartir recursos", y uno de sus objetivos es hacer que todos los programas, datos y equipo estén disponibles para cualquiera red que así lo solicite, sin importar la localización física del recurso y del usuario..

También consiste en proporcionar una alta fiabilidad, al contar con fuentes alternativas de suministro. Es por eso que las redes se clasifican en tres tipos, según el área geográfica que abarquen. La Red LAN formada por computadoras que se encuentran en un mismo edificio, fábrica o campus universitario, es decir en un radio de unos pocos kilómetros cuadrados, MAN Red que abarca el área de una ciudad y la WAN Red que abarca países enteros y hasta todo el mundo.

Como se ha mencionado anteriormente, en la actualidad llamamos RED a un conjunto de computadoras interconectadas entre sí. La INTERNET, por ejemplo, es una red de redes. Toda la tecnología actual de Internet-working que se basa en la interconexión de redes rápidas. Las redes rápidas son las redes locales o redes de área local. Dentro de las LAN, hay varios tipos. Las más comunes son "Ethernet" y "Token-ring".

Las redes "Ethernet" se basan en una "topología de bus" donde todas las interfaces conectadas al cable "escuchan" lo que otras interfaces "escriben". Cuando una interface desea enviar un dato a través de la red, simplemente comienza a transmitir.

Las redes "token-ring" se basan en una "topología de anillo", donde los datos circulan a través del anillo, de interface en interface. En esta topología, las interfaces no transmiten cuando lo desean, sino que esperan a recibir el token. Este token, va circulando por todas las interfaces dándole a cada una la posibilidad de transmitir.

Para acceder a las redes existen, básicamente, dos formas de acceso tecnológicas distintas:
"Líneas dedicadas" Estas líneas tienen un costo fijo mensual y sobre ellas se puede transmitir sincrónica o asincrónicamente, analógica o digitalmente, con velocidades variando entre 300 baudios y alguna cantidad (muy alta) de Mbits/seg, dependiendo del tipo de material que se use como transmisor, así como de la forma de transmisión. Por ejemplo, en un línea dedicada analógica sobre cable de cobre común (utilizado por el sistema telefónico) y transmitiendo analógicamente, se pueden obtener 19200 Kbits/seg, mientras que la misma línea, transmitiendo en modo digital puede alcanzar los 128 Kbits/seg.

"Redes públicas": La conexión a estas redes se hace, en general, a través de líneas dedicadas, aunque en algunos casos se puede acceder vía el sistema telefónico conmutado. La diferencia con el caso anterior es que la línea dedicada conecta al usuario con el proveedor del servicio. El proveedor del servicio opera una WAN que interconecta a muchos usuarios entre sí, pero en una WAN, las conexiones son virtuales (comúnmente denominados circuitos virtuales). Esto significa que por la misma línea, una interface puede mantener varias conexiones simultáneamente como lo son:

Los hubs son repetidores que trabajan a nivel de la capa física regenerando la señal que reciben por un puerto y transmitiéndola por los demás por lo que la función principal es la de repetir la señal que ingresa por cada una de sus “puertas” hacia todas las otras “puertas”, realizando por tanto la “difusión” que requiere Ethernet (y que se daba naturalmente en las topologías de bus sobre cables coaxiales). Los hubs también monitorizan el estado de los enlaces de las conexiones a sus puertas, para verificar que la red funciona correctamente.

Los switches Trabajan a nivel de capa 2. Reciben la trama, y (generalmente) luego la transmiten por el puerto que corresponde. Cuando una estación envía una trama el switch “aprende” la ubicación de dicha estación y tramas dirigidas a ella serán enviadas solo por ese puerto, lo que mejora mucho la performance de la red. Pero los broadcasts siguen enviándose a todos los puertos.

Los rotures Para poder interconectar redes LAN distantes, mediante algún protocolo de WAN, es necesario disponer de equipos de “interconexión”, que cumplan varias funciones, en las que se destacan la Posibilidad de rutear tráfico, para disminuir el tráfico de WAN no deseado (Broadcasts, etc.), posibilidad de manejar protocolos de LAN y de WAN.

Estos servicios de datos a alta velocidad suelen denominarse conexiones de banda ancha. Se prevé que proporcionen los enlaces necesarios entre LAN para hacer posible lo que han dado en llamarse autopistas de la información.

Proceso distribuido:

Parece lógico suponer que las computadoras podrán trabajar en conjunto cuando dispongan de la conexión de banda ancha. ¿Cómo conseguir, sin embargo, que computadoras de diferentes fabricantes en distintos países funcionen en común a través de todo el mundo? Hasta hace poco, la mayoría de las computadoras disponían de sus propias interfaces y presentaban su estructura particular. Un equipo podía comunicarse con otro de su misma familia, pero tenía grandes dificultades para hacerlo con un extraño. Sólo los más privilegiados disponían del tiempo, conocimientos y equipos necesarios para extraer de diferentes recursos informáticos aquello que necesitaban.

En los años noventa, el nivel de concordancia entre las diferentes computadoras alcanzó el punto en que podían interconectarse de forma eficaz, lo que le permite a cualquiera sacar provecho de un equipo remoto. Los principales componentes son:

Cliente / servidor: En vez de construir sistemas informáticos como elementos monolíticos, existe el acuerdo general de construirlos como sistemas cliente / servidor. El cliente (un usuario de PC) solicita un servicio (como imprimir) que un servidor le proporciona (un procesador conectado a la LAN). Este enfoque común de la estructura de los sistemas informáticos se traduce en una separación de las funciones que

anteriormente forman un todo. Los detalles de la realización van desde los planteamientos sencillos hasta la posibilidad real de manejar todos los ordenadores de modo uniforme. **Tecnología de objetos:** Otro de los enfoques para la construcción de los sistemas parte de la hipótesis de que deberían estar compuestos por elementos perfectamente definidos, objetos encerrados, definidos y materializados haciendo de ellos agentes independientes. La adopción de los objetos como medios para la construcción de sistemas informáticos ha colaborado a la posibilidad de intercambiar los diferentes elementos. **Sistemas abiertos**: Esta definición alude a sistemas informáticos cuya arquitectura permite una interconexión y una distribución fáciles. En la práctica, el concepto de sistema abierto se traduce en desvincular todos los componentes de un sistema y utilizar estructuras análogas en todos los demás. Esto conlleva una mezcla de normas (que indican a los fabricantes lo que deberían hacer) y de asociaciones (grupos de entidades afines que les ayudan a realizarlo). El efecto final es que sean capaces de hablar entre sí. El objetivo último de todo el esfuerzo invertido en los sistemas abiertos consiste en que cualquiera pueda adquirir computadoras de diferentes fabricantes, las coloque donde quiera, utilice conexiones de banda ancha para enlazarlas entre sí y las haga funcionar como una máquina compuesta capaz de sacar provecho de las conexiones de alta velocidad.

Seguridad y gestión:

El hecho de disponer de rápidas redes de computadoras capaces de interconectarse no constituye el punto final de este enfoque. Quedan por definir las figuras del "usuario de la autopista de la información" y de los "trabajos de la autovía de la información".

Seguridad: La seguridad informática va adquiriendo una importancia creciente con el aumento del volumen de información importante que se halla en las computadoras distribuidas. En este tipo de sistemas resulta muy sencillo para un usuario experto acceder subrepticiamente a datos de carácter confidencial. La norma Data Encryption System (DES) para protección de datos informáticos, implantada a finales de los años setenta, se ha visto complementada recientemente por los sistemas de clave pública que permiten a los usuarios codificar y descodificar con facilidad los mensajes sin intervención de terceras personas. **Gestión:** La labor de mantenimiento de la operativa de una LAN exige dedicación completa. Conseguir que una red distribuida por todo el mundo funcione sin problemas supone un reto aún mayor. Últimamente se viene dedicando gran atención a los conceptos básicos de la gestión de redes distribuidas y heterogéneas. Hay ya herramientas suficientes para esta importante parcela que permiten supervisar de manera eficaz las redes globales.

Por lo que es importante tener el **Monitoreo de la red**: que analiza el tráfico de la red o la de una parte, como también examina paquetes de datos y recopilación de información sobre los tipos de paquetes, errores y tráfico de paquetes desde y hacia cada equipo. Un monitoreo de red es una herramienta útil que captura y filtra los paquetes de datos y analiza la actividad de la red.

Es así al momento de utilizar los Sistemas Operativos en la Red se distribuyen **Linux:** que es un sistema operativo flexible y poderoso con montones de programas provee un excelente conjunto de utilidades y lenguajes (tantos como 20 lenguajes de programación actuales, incluyendo Java, C++, C, Pascal, Fortran, Awk, Sed, Perl, Lisp, etc.), aquí es gratis. Linux también viene con depuradores, librerías, y toneladas de código fuente para facilitar el desarrollo de programas.

Linux no es sólo un simple mímica freeware del verdadero UNIX. Si no que es de ALTO CALIBRE que soporta todos los rasgos UNIX excelentemente. Esto incluye servicio Web, servidores de correo, POP3, y hasta el sistema Windows XP con una interfaz similar a Win95 (por defecto solamente en RedHat Linux), otros sistemas operativo que se utiliza en red son también **Windows 2000 Server /Profesional, Windows XP, NT y 95/ 98:** Un sistema operativo de red, conecta todos los equipos y periféricos, coordina las funciones proporciona seguridad controlando el acceso a los datos y a su vez funcionando sobre cualquier protocolo IP o IPX, LAN o WAN, red de marcado manual, VPN o en Internet..

Dentro de los sistemas operativos se pueden mencionar que hay **Herramientas de software** disponibles para que el administrador de red que pueda resolver los problemas de conectividad de la red. Estas herramientas pueden ayudar en el diagnóstico de fallas de las redes de área local, pero especialmente útiles para resolver los problemas de las redes de área amplia. Se analizarán los comandos disponibles para un administrador de red en la mayoría de los paquetes de software cliente. Entre estos comandos se incluyen: Ping, Tracert (Traceroute), Telnet, Netstat, (Traceroute), Telnet, Netstat, ARP y IPconfig (WinIPcfg).

Ping: Envía paquetes de eco ICMP para verificar las conexiones a un host remoto. Muestra si el ping fue exitoso. El resultado muestra la cantidad de paquetes a los que se respondió y el tiempo de retorno del eco. **Tracert (Traceroute):** Está utilidad muestra la ruta que siguió un paquete para alcanzar su destino. El resultado muestra el comando trace. **Telnet:** Este es un programa de emulación de terminal que le permitirá ejecutar comandos interactivos en el servidor Telnet. Hasta que se establece

una conexión, no pasa ningún dato y si la conexión se interrumpe, Telnet lo informa. Es bueno para probar parámetros de configuración de conexión a un host remoto. **ARP:** Se usa para reunir direcciones de hardware para los hosts locales, el Gateway por defecto se puede ver el caché ARP y verificar la existencia de entradas no válidas o duplicadas **IPconfig (Windows NT)/WinIPcfg (Windows 95-98):** Estas utilidades de Windows muestran información de direccionamiento IP para el adaptador(es) de red local o una NIC especificada. Estas son las herramientas que permiten que un administrador de red monitoree y controle la misma de forma remota.

Para poder trabajar dentro del monitoreo se necesita del Protocolo **TCP/ IP:** que es un protocolo DARPA que proporciona transmisión fiable de paquetes de datos sobre redes. El nombre TCP / IP que proviene de dos protocolos importantes de la familia, el Transmisión Control Protocol (TCP) y el Internet Protocol (IP).Todos juntos llegan a ser más de 100 protocolos diferentes definidos en este conjunto de los cuales se mencionara el más importante. **SNMP**: Simple Network Management Protocol, Protocolo simple de gestión de red, es un protocolo diseñado para dar al usuario capacidad de manejar remotamente otro ordenador, preguntándole y dándole valores y monitorizando los eventos que ocurren en la red. Está compuesto del MIB, del gestor (manager) y del agente gestionado (agent). SNMP funciona sobre TCP/IP en su nivel de aplicación. Existen dos versiones SNMPV1 y SNMPV2

Al usar SNMP el diseño es simple por lo que su implementación es sencilla en grandes redes y la información de gestión que se necesita intercambiar ocupa pocos recursos de la red. Además, permite al usuario elegir las variables que desea monitorizar sin más que definir. El SNMP es el único protocolo que existió en un principio y por ello casi todos los fabricantes de dispositivos como puentes y en caminadores diseñan sus productos para soportar SNMP. La posibilidad de expansión es otra ventaja del protocolo SNMP: debido a su sencillez es fácil de actualizar.

Los Lenguajes que se utilizan mayormente en el desarrollo de software para redes se encuentran:

Java: El cual ofrece toda la funcionalidad de un lenguaje potente, java trabaja con sus datos como objetos y con interfaces a esos objetos. Soporta las tres características propias del paradigma de la orientación a objetos: encapsulación, herencia y polimorfismo.

Java se ha construido con extensas capacidades de interconexión TCP/IP. Existen librerías de rutinas para acceder e interactuar con protocolos como http y ftp. Esto permite a los programadores acceder a la información a través de la red con tanta facilidad como a los ficheros locales.

El lenguaje **C++** es un lenguaje de programación de propósito general. Todo puede programarse con él, desde sistemas operativos y compiladores hasta aplicaciones de bases de datos y procesadores de texto, pasando por juegos, aplicaciones a medida, etc.

También hay que pensar que sistemas operativos como Linux, Unix o incluso Windows se escriben casi por completo en C.

Por otro lado las bases de datos se hacen necesarias para la implementación de aplicación de software de monitoreo de redes, entre las bases de datos más utilizadas están:

MySQL: Es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente.

Informix: proporciona fiabilidad superior, atendiendo las necesidades de las exigentes prácticas actuales del e-business-particularmente para aplicativos que requieran transacciones de alto desempeño. Soporta requisitos de procesamiento de transacción online, complejos y rigurosos. Optimiza capacidades de inteligencia del negocio competitivas. Maximiza operaciones de datos para el grupo de trabajo y para la empresa en total. Proporciona la firmeza de una administración de base de datos comprobada.

SQL: (Standar Query Lenguaje) Es un lenguaje estandarizado de base de datos, el cual nos permite realizar tablas y obtener datos de ella de manera muy sencilla.

ORACLE: Es el conjunto de datos que proporciona la capacidad de almacenar y acude a estos de forma consecuente con un modelo definido como relacional. Además es una suite de productos que ofrece una gran variedad de herramientas.

Postgres: Que es un sistema que permite la manipulación de datos de acuerdo con las reglas del álgebra relacional. Los datos se almacenan en tablas de columnas y renglones. Con el uso de llaves, esas tablas se pueden relacionar unas con otras. Una verdadera RDBMS debe soportar ACID que significa:

A-Atomicity. Mejor conocido como transacciones. Es cuando una serie de querys se ejecutan como un solo query. Si alguno de ellos falla, todos los demás relacionados no se llevan a cabo.

C-Consistency. Una Base de Datos tiene consistencia si se lleva de un estado válido a otro.

I-Isolation. Cuando una transacción se lleva a cabo no debe haber interferencia de otras transacciones.

D-Durability. Los cambios aplicados a la DB deben ser confiables y deben sobrevivir a las fallas.

Froufe Agustín, Java 2 Manual del Usuario y Tutorial, año 2000
"http://www.cert.org/advisories/CA-2002-03.html"
"www.monografias.com"

4. JUSTIFICACION

Los elementos que justifican la realización del presente trabajo son una parte a la contribución del conocimiento actual del área de estudio a través de un diagnóstico de los recursos y medios con los cuales puede contar.

Es por ello que se ha creído conveniente justificar el siguiente proyecto desde los siguientes puntos.

JUSTIFICACIÓN ACADEMICA

Concretamente el Estado, y la Universidad Nacional de Loja, en forma indirecta han invertido recursos financieros, humanos y materiales, siendo estos beneficios aprovechados, es por eso que hoy se ha presentado a la Universidad Nacional de Loja en su especialidad de Ingeniería en Sistemas, este anteproyecto con la finalidad de obtener el título y contribuir con el desarrollo del país.

JUSTIFICACIÓN TÉCNICA

Después de haber analizado las necesidades del Personal de Administración Central se ha determinado la Tecnología Hardware y Software para el monitoreo de la red, el cual contara con el lenguaje JAVA, el mismo que proporcionara repuestas claras y precisas mientras que, el protocolo que va a monitorear será el SNMP, el mismo que

nos permitirá obtener información de los distintos computadores asociados y a la vez el soporte que tienen sus equipos para trabajar con este protocolo.

JUSTIFICACIÓN ECONOMICA

El sistema a realizarse beneficiará en gran parte los problemas existentes en la U.N.L pues el paquete de Sistema Operativo que se utiliza actualmente representa costos excesivos y no duraderos y el software que se planea realizar estaría reduciendo costos y su reestructuración será de acuerdo a las necesidades del usuario.

Los costos que demandará el desarrollo del presente proyecto serán asumidos por el aspirante.

JUSTIFICACIÓN OPERATIVA

En estas circunstancias hay que tener en cuenta que los beneficios serán reales es decir que los servicios que éste prestara serán rápidos y seguros.

- El desarrollo de este proyecto, causara aspectos positivos para la institución dando una mejor agilidad en el monitoreo de cada una de las áreas.

OBJETIVOS

General

Diseñar una herramienta de monitoreo en la red para Administración Central de la Universidad Nacional de Loja basado en el protocolo TCP/ IP (SNMP), a fin de facilitar la administración de la Red.

Específicos

1. Permitir el análisis del tráfico de la red de cualquier computador de las Áreas.
2. Desarrollar una visión integrada de los computadores.
3. Determinar las características que tienen los dispositivos de red, disco duro y CPU.
4. Dar una facilidad de consulta a los estudiantes tanto teórico como practicó en el manejo del monitoreo de la red.
5. Contribuir con la implantación de un software en la Universidad Nacional de Loja, la misma que ayudara a resolver los diferentes problemas que se presentan al desarrollar sus actividades.
6. Optimizar los recursos humanos, técnicos y económicos en labores de administración de la red

6. METODOLOGÍA

La metodología para la resolución de este proyecto implica el establecimiento de técnicas, métodos y procedimientos de: recopilación, análisis e interpretación de la información; permitiendo la planificación de todas las fases de la investigación.
Es recomendable que se formule tomando en consideración los objetivos, que persigue el proyecto. Los métodos a utilizar son:

Métodos Deductivo.- Se necesitara conocer ciertas generalidades en cuanto a Redes para poder particularizar y así llegar al problema especifico que se va a resolver.
Para especificar el proceso de Implementación de un software para monitorear la red se seguirá las siguientes fases como son:

- Identificar los requerimientos de la red, siendo estos tecnológicos, en hardware y software.

- Realizar la distribución física de los equipos de comunicación y que se van a utilizar en el monitoreo de la red.
- Efectuar el estudio técnico detallado de la topología con que están trabajando y del protocolo a utilizarse.

Método Científico, ya que el mismo permitirá partir de conceptos, juicios y razonamientos, como también la de combinar ciertas reglas lógicas con el propósito de producir nuevas ideas y el de poder aportar a la ciencia.

Método Analítico, Se analizara como está funcionando toda la red para poder hacer pruebas de la implementación de software a desarrollar.
El Marco Metodológico para el presente Proyecto constará de tres etapas, siendo estas las siguientes:

1.- ORGANIZACIÓN

En la Etapa de Organización se llevará adelante las siguientes actividades:

1) Modelamiento del Proyecto

En esta dimensión se busca sentar las bases del Proyecto, así como determinar su Factibilidad dentro de una primera instancia. También se determinan los Objetivos, se vislumbran las metas, se describen las Principales Actividades y se señalan los Principales Productos, así como el Cronograma de Ejecución del Proyecto, otros.

2) Modelamiento de la Institución

En esta dimensión se busca la alineación del Proyecto con el Plan Estratégico de Sistemas de Información y el Plan de Tecnología. Además se busca organizar a las áreas de trabajo de la institución, para poder llevar adelante el Proyecto, esto debe entenderse en el sentido de que el personal de la institución debe colaborar con el proyecto.

3) Modelamiento del Requerimiento

En esta dimensión se busca la Definición de Requerimientos que deben ser satisfechos por el Proyecto de Red, Pisos, Áreas, Grupos de Trabajo, Puntos, Cableado, Otros.
En esta primera etapa una de las más importantes actividades es la de señalar los sistemas que van a trabajar en la Red; siendo estos sistemas los que van a justificar la viabilidad del Proyecto de Implantación de un software para monitoreo de la Red.

La idea es que los sistemas principales que justifican la implantación del software estén concluidos y probados al mismo momento (o poco antes) que la terminación, garantizando con ello un uso efectivo de dicha Red.

2. ANÁLISIS DE OBJETOS

Se contempla un modelo de lo que va hacer el sistema, el mismo que se expresará en términos de objetos y de relaciones, flujo dinámico de control y transformaciones funcionales, siguiendo los pasos o etapas del análisis tenemos en primer lugar:

1) Construcción de un modelo de Objetos**,** este modelo constará de diagrama de modelos de objetos más un diccionario de datos. Para la construcción de este paso se tomará en cuenta los siguientes puntos:

Se identificarán las clases de objetos a utilizarse en el desarrollo de la herramienta.

Se elaborara un diccionario que contendrá las descripciones de clases, atributos y asociaciones en la base de datos del software.

Se incluirán las asociaciones entre clases, como también los atributos de objetos y los enlaces que se deriven de estas relaciones.

Existirá la comprobación de las vías de acceso empleando escenarios e iterando los pasos anteriores cuando sea conveniente

2) Desarrollo de un Modelo Dinámico**,** el mismo que incluirá diagramas de estados más diagrama global de flujo de sucesos contando con los siguientes aspectos:
    - Se prepararán escenarios de secuencias típicas de interacción.
    - Se identificarán sucesos entre objetos y se preparará un seguimiento de sucesos para cada escenario.
    - Se preparará un diagrama de flujo de sucesos para el sistema.
    - Se desarrollará un diagrama de estados para cada clase que tenga un comportamiento dinámico importante.
    - Se comprobará la congruencia de los sucesos compartidos entre diagramas de estados.

3) Construcción de un Modelo Funcional**,** estará basado en diagramas de flujo de datos más restricciones, tomando en cuenta los siguientes puntos para su desarrollo:

- Se identificarán los valores de entradas y de salida.
- Se utilizarán diagramas de flujo de datos según sea necesario para mostrar las dependencias funcionales.
- Se describirá lo que hace cada función.
- Se identificarán las restricciones.
- Se especificarán los criterios de optimización.

4) Verificación de modelos**,** se verificará que las clases, asociaciones, operaciones y atributos sean congruentes y que estén completas dentro del nivel de abstracción seleccionado. Se comparará los tres modelos con la definición o descripción del problema y con el conocimiento relevante del dominio, empleando escenarios.

Se desarrollarán escenarios más detallados (incluyendo condiciones de error) como variaciones de los escenarios básicos con la finalidad de "qué pasaría si...." para verificar a un más los tres modelos.

3. DISEÑO DE OBJETOS

Para el diseño de objetos, se elaborará el modelo de análisis y se proporcionará una base detallada para la implementación, tomando las decisiones que sean necesarias para poder tener como objetivo de recolectar y analizar su comportamiento en diversos aspectos, ya sea en un momento en particular (tiempo real) o en un intervalo de tiempo. Esto permitirá tomar las decisiones pertinentes que se encuentren en el comportamiento del desarrollo de la herramienta de monitoreo.

1.- Diseño de algoritmos para implementar las operaciones que minimicen el costo de las operaciones, estructuras de datos adecuados para los algoritmos y nuevas clases y operaciones internas según sea necesario.

4. IMPLEMENTACIÓN

Se elaborará cada uno de los componentes de la herramienta de monitoreo en la cual se utilizara el lenguaje orientado a objetos (Java) que será almacenado en la base de datos Informix y que deberá ser implementado para el sistema operativo Windows o Linux los cuales servirán de soporte para generar una herramienta informática de calidad.

2.- Diseño de algoritmos para implementar las operaciones de algoritmos que minimicen el costo de implementar las operaciones, estructuras de datos adecuados para los algoritmos y nuevas clases y operaciones internas según sea necesario.

La correcta evaluación del Proyecto, debe permitir implantar correctivos que coadyuven al éxito del Proyecto, teniendo a los usuarios como principio y fin para el desarrollo exitoso del monitoreo de la Red.

8. RECURSOS Y PRESUPUESTOS.

8.1 RECURSOS.

RECURSOS HUMANOS.

Cant.	Descripción
1	Director de Tesis.
1	Aspirantes a Ingeniero en Sistemas.
1	Asesor para desarrollo de la tesis.

RECURSOS TÉCNICOS.

Cant.	Descripción
HARDWARE	
1	Computador : Procesador Intel P IV Memoria de 512 Mb Disco duro de 120 Gb Tarjeta de Red 10/100. CD - RW 52x24x52 Samsung
1	Impresora Canon S200 de inyección a tinta.
SOFTWARE	
1	Kit de Office para Linux
1	Licencia de Microsoft Project.
1	Kit de Software libre (Java, Linux), Mysql
COMUNICACIONES	
1	Hub, switch, rotures.
	Servicio de comunicación en Internet

RECURSOS MATERIALES.

Cant.	Descripción
1	Texto : S.O.Linux
1	Texto: Redes y servicios en las Telecomunicaciones.
1	Texto: Sobre SNMP.
1	Texto: Base de Datos (Mysql).
1	Texto: Programación en Java.
1	Texto: Sobre protocolos TCP/ IP.
3	Resmas de papel formato A4.
500	Copias (Aprox.)
6	Cartuchos de tinta
5	Empastados
1	Caja de diskette
2	CD-RW

8.2 PRESUPUESTO.

Cant.	Descripción	Duración	V./U.	V./T.
RECURSOS HUMANOS.				
1	Director de Tesis	-	-	-
1	Ingenieros en Sistemas	1100 horas	$5.00/h	$ 5500
1	Consultor para desarrollo de tesis	20 horas	$5.00/h	$ 100
			TOTAL	$5600
RECURSOS TÉCNICOS.				
HARDWARE.				
1	Adquisición de un computador	-	$750	$750
1	Impresora	-	$55	$ 55
TOTAL				$805
SOFTWARE.				

1	Licencia gratuita: Linux, java con la versión j2sdk1.4.0, Base de datos Mysql.	-		
TOTAL				
COMUNICACIÓN.				
1	Enlace en Internet	90horas	$ 0.80	$72
1	Arriendo de centro de computo	8 horas	$ 1.00	$ 48
TOTAL				$120
RECURSOS MATERIALES.				
Costo en bibliografía		-	-	$ 270
Suministros de Oficina		-	-	$ 200
Encuadernación		-	-	$ 25
TOTAL				$495
Total				*$ 7020*
Imprevistos 7%				$ 491.4
COSTO TOTAL PROYECTO				***$ 7551.4***

NOTAS:

1. Las Licencias en las que se desarrolla ya están adquiridas por la Universidad Nacional de Loja
2. Las horas que cobra un Ingeniero en Sistemas es de $ 5 dólares por cada hora
3. El alquiler de un Cyber es de $ 0.50 centavos la hora.
4. El Ingeniero en Sistemas realizara en 20horas diarias de lunes a viernes, 20 días al mes.

9. BIBLIOGRAFÍAS

Libros:

- Cox, B.J. y Novobliski, A. Programación orientada a objetos. Addison-Wesley Iberoamericana/Díaz de Santos. México, D.F.

- Date, C.J. Introducción a los sistemas de bases de datos. Addison-Wesley. México.

- Enciclopedia Microsoft(R). Redes de comunicación. Encarta(R) 98. (c) 1993-1997 Microsoft Corporation.

- PISCITELLI, Alejandro; Gustavo, Redes Electrónicas y el Proyecto Visión Buenos Aires, CLACSO, 1992.

- Wang, Henry H."*Telecommunications Network Management*". McGraw-Hill, 1999.

Sitios Web:

- **URL:** linuxpreview.org
  Descripción: Linux Preview como instalar y utilizar su computador [consulta, 15 de Noviembre del 2004]

- **URL:** monografias.com
  Descripción: Es un sitio de información y aprendizaje [consulta, 27 de Noviembre del 2004]

- **URL:** monografias.com
  Descripción: Introducción a Redes [consulta, 31 de Diciembre del 2004]

- **URL:** monografias.com
  Descripción: Linux [consulta, 19 de Marzo del 2005]

- **URL:** monografias.com
  Descripción: Redes [consulta, 19 Octubre del 2005]

- **URL:** monografias.com
  Descripción: Redes Inalámbricas [consulta, 19 de Mayo del 2005]

ANEXO A

DESCRIPCIÓN Y ALCANCE

En la actualidad podemos observar los avances tecnológicos en diferentes disciplinas especialmente en la computación, pues esta ha tenido grandes logros en las telecomunicaciones especialmente en la utilización de herramientas para monitorear la red.

El software que se creara permitirá monitorear a cada una de las áreas es decir que se podrá saber si los demás computadores están encendidos o apagados, determinando el tiempo en el que llegara a su destino.

El producto que se entregará será fiable, flexible y fácil de usar para el programador como para el usuario final.

Refiriéndose al tema planteado se especificara de la siguiente manera:

- El lenguaje de programación con el que se trabajara es Java, el cual se ha construido con extensas capacidades de interconexión TCP/IP, en el cual existen librerías de rutinas para acceder e interactuar con protocolos como http y ftp. Esto permite a los programadores acceder a la información a través de la red con tanta facilidad como a los ficheros locales
- Se trabajara con un sistema de base de datos que perita la manipulación de los datos que se almacenen.
- El protocolo de comunicación es el TCP/ IP(SNMP) que proporciona transmisión fiable de paquetes de datos sobre redes.
- Se Utilizara los computadores de las diferentes áreas para las pruebas.
- Se trabajara con hardware que soporte las versiones de agentes en SNMP: Switch 3com 1100/3300, cisco switch catalis 2950.

A continuación se detallará que ejecutara la aplicación a realizar:

Encendidos o apagados

- Realizar un ping a la maquina que deseamos ver.
- detallar todas las características de los switch y rotures.
- monitorear a las demás áreas.
- Un esquema de seguridad para acceso al sistema.

Tiempo en llegar al destino

- Dar el tiempo en milisegundo y el acceso a la otra máquina para ver si está funcionando.
- Presentar por cuantos computadores tuvo que pasar.

Es por eso que este software podrá ser implementado en el servidor de Administración Central o en cualquier red de la Universidad Nacional de Loja, permitiendo detectar y solucionar los problemas que se presenten en la misma.