ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

VISUALIZADOR DE ESTADO DE RED

TESINA DE SEMINARIO

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

INGENIERO EN TELEMÁTICA

PRESENTADO POR:

LÓPEZ ENCALADA DIEGO ANDRES

SERRANO PEREZ MARÍA VERÓNICA

GUAYAQUIL ECUADOR
2011

AGRADECIMIENTOS

Agradezco a mis padres por brindarme siempre su apoyo, y darme la fortaleza necesaria para formarme profesionalmente.

A mi hermano José Enrique que estuvo presente desde el momento en que ingrese a esta prestigiosa institución y me supo aconsejar en los momentos que más lo necesite.

DIEGO ANDRÉS

Agradezco a toda mi familia por apoyarme a cada momento de mi vida estudiantil, por brindarme fortaleza y cariño. A mis dos mamás, Anita Victoria porque a base de paciencia y esfuerzo supo encaminarme y formarme como persona, a Marcela Isabel por día a día pese a la distancia darme su amor incondicional y fortaleza.

A mis hermanos Anita María, Fausto José, Katyta, Fausto Javier, AnitaMaria Fernanda y Fausto Alejandro por ser mi alegría diaria, ayudarme a levantarme después de cada caída y ser mi ejemplo a seguir.

Agradezco también con mucho amor a mi esposo Andrés Abad y a mi hijo Andrés Fernando por ser mi inspiración diaria y mi razón de querer salir adelante.

A todos muchísimas gracias.

MARIA VERÓNICA

DEDICATORIAS

A mis padres Freddy y Mercy, por su guía permanente y su apoyo incondicional.

A mis hermanos Freddy Augusto, José Enrique,
María Elena y Sofhía Daniela, por su confianza y afecto.

DIEGO ANDRÉS

Agradezco a mi familia Anita Victoria, Marcela Isabel, Anita María, Fausto José, Katyta, Fausto Javier, AnitaMaria Fernanda por darme sus cuidados y amor y convertirme en quien soy ahora.

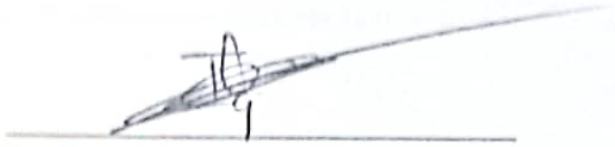
A Andres Abad y Andres Fernando Abad Serrano por darme toda la felicidad y motivación que necesito en esta nueva etapa, mi vida profesional.

Y de forma especial esta tesina y profesión se la dedico a mi papá Fausto José Serrano García, quien desde el cielo aplaude este logro, quien nunca me ha dejado sola, me cuida y protege desde arriba llenándome de bendiciones y guiando cada uno de mis pasos.

Para usted papito, le dedico con todo mi corazón mi carrera, mi esfuerzo y cada uno de mis logros, los que he obtenido y los que vendrán.

MARIA VERÓNICA

TRIBUNAL DE SUSTENTACIÓN



MSc. Ignacio Marín García
Profesor Materia de Graduación

Ana Bruno Tapia

MBA. Ana Tapia Rosero
Profesora Delegada del Decano

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de esta Tesina de Grado, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral".

(Reglamento de Graduación de la ESPOL)

Diego Andrés López Encalada

Mat. No 200529113

María Verónica Serrano Pérez

Mat. No 200618874

RESUMEN

El siguiente trabajo de tesis está basado en el área de redes de computadoras, consiste en la implementación de un aplicativo que permita de forma visual y gráfica ver la topología de una red e identificar los protocolos y conexiones existentes que existan dentro de ella. Para la realización de este trabajo vamos a poner en práctica los conocimientos adquiridos a lo largo de nuestra carrera, y en especial en el dictado de la materia de graduación “Seguridad en Redes”.

Nos proponemos desarrollar un aplicativo que permita a un usuario en forma gráfica y detallada ver qué se esconde en cada uno de los protocolos capturados en una red. El software se podrá ejecutar en cualquier máquina, ya que se va a utilizar el lenguaje JAVA para su implementación, la aplicación se centrará en el análisis de los siguientes protocolos: Paquetes IP, ICMP y SNMP, segmentos TCP y UDP.

ÍNDICE GENERAL

ÍNDICE FIGURAS

ÍNDICE TABLAS

ABREVIATURAS

ARP.- Protocolo de Resolución de Direcciones (Address Resolution Protocol). Usado por equipos de redes para correlacionar una dirección lógica con una dirección física.

ARPANET.- Red de la Agencia de Proyectos de Investigación Avanzada (Advanced Research Projects Agency Network). Red de conmutación de paquetes desarrollada a principio de la década de los setenta por ARPA que se considera el origen de la actual red Internet.

ASN.1.- Notación Sintáctica Abstracta 1 (Abstract Syntax Notation One). Norma para representar datos independientes de la máquina que se esté usando y sus formas de representación internas.

ATM.- Modo de Transferencia Asíncrona (Asynchronous Transfer Mode). Método de transmisión de celdas de tamaño fijo (53 bytes) utilizada en redes de Banda Ancha. ATM puede transferir datos a tasas desde 25 Mbps hasta 622 Mbps.

DNS.- Sistema de Nombre de Dominio (Domain Name System). Servicio que devuelve una dirección lógica, ejemplo IP, en base a un nombre, por ejemplo URL.

FTP.- Protocolo de Transferencia de Ficheros **(**File Transfer Protocol). Permite a los usuarios recibir y enviar ficheros a través de una red.

FDDI.- Interfaz de distribución de datos de fibra óptica (Fiber Distributed Data Interface). Fue diseñada para cumplir los requerimientos de redes

individuales de alta velocidad, y conexiones de alta velocidad en redes individuales. El estándar FDDI lo desarrolló el comité de estándares acreditado X3T9.5 que está reconocido por el ANSI. Las principales razones para seleccionar FDDI son la distancia, la seguridad y la velocidad.

HTTP.- Protocolo de Transferencia de Hipertexto (HyperText Transfer Protocol). Utilizado para transferir archivos en formato de hipertexto a través de la red.

HTTPS.- Protocolo Seguro de Transferencia de Hipertexto (Hypertext Transfer Protocol Secure). Es la aplicación segura mediante SSL, para crear un canal cifrado (cuyo nivel de cifrado depende del servidor remoto y del navegador utilizado por el cliente) más apropiado para el tráfico de información sensible que el protocolo HTTP. Opera en la capa de aplicación del modelo TCP/IP.

ICMP.- Protocolo de Mensajes de Control de Internet (Internet Control Message Protocol). Se utiliza para comprobar la conectividad y cualquier mensaje de control.

IP.- Protocolo de Internet (Internet Protocol). Protocolo no orientado a conexión usado tanto por el origen como por el destino para la comunicación de datos a través de una red de paquetes conmutados no fiable de mejor entrega posible sin garantías. Opera principalmente en la capa de Internet del modelo TCP/IP.

LAN.- Red de Área Local (Local Area Network). Es una red que cubre el área de un edificio, una planta de edificio u oficina.

MAC.- Control de Acceso al Medio (Medium Access Control). Identificador de 48 bytes que corresponde de forma única a una tarjeta o dispositivo de red. Se la conoce como dirección física, y es única para cada dispositivo.

MIB.- Base de Información Gestionada (Management Information Base). Es un tipo de base de datos que contiene información jerárquica estructurada en forma de árbol de todos los dispositivos gestionados en una red.

MO.- Objetos Administrados (Management Object). Es una manera de ver los recursos para los propósitos de administración.

NMS.- Sistema de Administración de Redes (Network Management System). Es un terminal a través del cual los administradores pueden llevar a cabo tareas de administración.

PDU.- Unidades de Datos del Protocolo (Protocol Data Units). Unidad que define el empaquetamiento de los datos en un protocolo del modelo TCP/IP.

QoS.- Calidad de Servicio (Quality of Service). Se refiere a la calidad en la transmisión y recepción de información a través de una red de datos. Su objetivo es proporcionar un mejor servicio a ciertos flujos de datos.

RARP.- Protocolo de Resolución de Dirección Inverso (Reverse Address Resolution Protocol). Es lo inverso a ARP, en base a una dirección física devuelve una dirección lógica.

SMDS.- Servicio de Datos Conmutado Multimegabit (Switched Multimegabit Data Service). Es un servicio capaz de proporcionar un transporte de datos transparente “no orientado a la conexión” entre locales de abonados utilizando accesos de alta velocidad a redes públicas.

SMI.- Estructura de Información de Administración (Structure of Management Information). Describe cómo se definen los objetos gestionados contenidos en el MIB.

SMTP.- Protocolo Simple para Transferencia de Correo (Simple Mail Transfer Protocol). Se usa para enviar y retransmitir correo electrónico entre servidores.

SNMP.- Protocolo Simple de Administración de Red (Simple Network Management Protocol). Protocolo situado en la capa de aplicación del modelo TCP/IP, que facilita el intercambio de información de administración entre dispositivos de red

TCP.- Protocolo de Control de Transmisión (Transmission Control Protocol). Protocolo encargado de controlar el flujo de la transmisión de datos entre redes de iguales o diferentes tipos.

TFTP.- Protocolo para Transferencia Trivial de Archivos (Trivial File Transfer Protocol). Es una implementación sencilla del protocolo FTP.

TMN.- Administrador de Redes de Telecomunicaciones (Telecommunications Management Network). Modelo de protocolo definido por la ITU-T para la gestión de los sistemas abiertos en una red de comunicaciones.

UDP.- Protocolo de Datagrama de Usuario (User Datagram Protocol). Es un protocolo situado en la capa de transporte del modelo TCP/IP, que realiza las funciones parecidas al TCP, pero no asegura la transmisión de datos, tampoco realiza control de flujo.

WAN.- Red de Área Amplia (Wide Area Network). Red de equipos que abarca un área superior a una ciudad.

ITU.- Unión Internacional de Telecomunicaciones (International Telecommunication Union). Organismo especializado de la Organización de las Naciones Unidas encargado de regular las telecomunicaciones a nivel internacional entre las distintas administraciones y empresas operadoras.

OSI.- Interconexión de Sistemas Abiertos (Open System Interconnection). Modelo de red descriptivo creado por la Organización Internacional para la estandarización en el año 1984.

SSL.- Capa de Conexión Segura (Secure Sockets Layer). Proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de cifrado de información.

GLOSARIO

Administración de Redes: Es el conjunto de actividades que se basan en el acoplamiento de diversas tecnologías para planificar, modelar, diseñar, configurar y desarrollar redes informáticas con la finalidad de obtener un óptimo desempeño a un costo razonable y con la máxima eficiencia; dando así la capacidad de mantener, corregir, contabilizar, evaluar y expandir los recursos que la componen.

Checksum: Una suma de verificación o checksum es una forma de control de redundancia muy simple para proteger la integridad de datos, verificando que no hayan sido corrompidos.

Escáner de Puertos: Herramienta que permite analizar el estado de los puertos de una máquina conectada a una red de comunicaciones.

Ethernet: Protocolo de capa de enlace de datos, que brinda acceso al medio por contención mediante la técnica CSMA/CD.

Host.- Término usado en informática para referirse a las computadoras conectadas a una red, que proveen y utilizan servicios de ella. Los usuarios deben utilizar hosts para tener acceso a la red.

IPv4: Versión 4 del Protocolo de Internet, ampliamente utilizado y que en la actualidad está siendo reemplazado por su sucesor Ipv6, debido al agotamiento de direcciones en Ipv4, entre otros problemas.

Java: Lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina funcionalidades de bajo nivel.

Manager: Es el rol que se asigna al dispositivo donde se encuentra instalado el NMS. El manager se encarga de hacer consultas al agente.

Ping: Es una herramienta que permite probar la conectividad entre dos hosts. Se basa en el envío de mensajes ICMP Echo Request para conocer si un dispositivo se encuentra activo.

POP3: Es un protocolo de la capa de aplicación del modelo OSI que se encarga del manejo de correos entrantes, utilizando el puerto 110 por defecto para establecer una conexión TCP.

rlogin: (Remote Login) Es una aplicación, que permite una sesión de terminal remoto sobre el anfitrión especificado como host.

Sniffer: Programa que permite la captura de los paquetes que transitan por la red y el análisis de los protocolos que componen el mismo.

Telnet: Es un protocolo que permite acceder en forma remota a los sistemas informáticos de la misma manera en que se puede hacer con las terminales conectadas en forma directa.

Tracert: Es una utilidad que permite observar la ruta entre dos hosts. El rastreo genera una lista de saltos alcanzados con éxito a lo largo de la ruta. Al igual que la herramienta ping, se basa en el envío de mensajes ICMP

Echo Request variando el campo TTL o Hop Limit (según la versión IP) para conocer cada uno de los saltos por los que transita un paquete para llegar a su destino.

Frame Relay: Es una técnica de comunicación mediante retransmisión de tramas para redes de circuito virtual.

Proxy ARP: Es una técnica para usar el ARP para proporcionar un mecanismo de enrutamiento.

RUP: Metodología para el proceso de desarrollo de software, que se la puede dividir en 4 fases: inicio, elaboración, desarrollo y cierre de un proyecto de software.

Modelo TCP/IP: Se basa en los protocolos estándares que se han desarrollado, y se las ha llegado a organizar en cuatro capas relativamente independientes: aplicación, transporte, internet y acceso a la red.

INTRODUCCIÓN

En la actualidad la masificación de la comunicación, unida a la amplia gama de medios para el intercambio de información, ha creado la necesidad de agrupar las distintas formas de comunicación en una arquitectura común. Hoy en día la seguridad en redes desempeña un papel primordial en las empresas, más aún cuando se debe precautelar la protección de los datos, por esta razón se debe crear nuevas herramientas de software para combatir las nuevas amenazas que existe a través de Internet.

Pese a que conocemos que nunca lograremos alcanzar obtener una red absolutamente segura, podemos tomar medidas que nos ayuden a aumentar la seguridad e integridad de nuestra red; Es por esto que el software propuesto en el presente proyecto proporciona una forma de hacerlo, la cual consiste en realizar un escaneo de redes, puertos, protocolos y tráfico que transita por dicha red.

CAPÍTULO I
ANTECEDENTES

Una vez expuesta la vulnerabilidad de las redes debido a la rápida evolución de los ataques informáticos y de la problemática en la que se convierte para los administradores de red, intentaremos dar a conocer en el presente capítulo de una forma breve y sencilla la necesidad de implementación de un programa como el nuestro y justificar la solución planteada en él.

1.1 PLANTEAMIENTO DEL PROBLEMA

Desde que surgió Arpanet para permitir la comunicación general entre varias computadoras en la década de los setenta, que reconoceríamos como la base del moderno Internet, han existido los problemas de seguridad. Sin embargo, para mitigar los ataques que se encontraban no era necesario tener

conocimientos técnicos avanzados; por otro lado los ataques desde el interior de una red se basaban en la alteración de los permisos para modificar la información del sistema, mientras que los ataques externos se daban debido al conocimiento de las contraseñas necesarias para acceder a los equipos de red.

Una forma de dividir o diferenciar los ataques sufridos en una red son en dos categorías: **activos** por la falta de conocimiento del usuario acerca de las funciones del sistema, algunas de las cuales pueden ser dañinas para el sistema, y **pasivos** cuando el administrador (o el usuario) de un sistema no está familiarizado con los mecanismos de seguridad presentes en la red.

1.2 JUSTIFICACIÓN DEL TEMA

Nuestro proyecto tiene como función principal el proveer una herramienta gráfica de fácil manejo, que va a monitorear una red de acuerdo a su funcionamiento. Esto ayudará a los encargados de la seguridad en red de una empresa a disminuir la vulnerabilidad de los datos que se transmiten, conociendo qué tipos de protocolos y puertos son los que están en

funcionamiento en los diferentes sistemas operativos, ayudando a comprender el funcionamiento del modelo TCP/IP.

1.3 DESCRIPCIÓN DEL PROYECTO

El proyecto es una aplicación en JAVA, ejecutada en una PC conectada a una red, que monitorea diferentes dispositivos interconectados, y permite de forma visual y gráfica ver la topología de dicha red e identificar los protocolos y conexiones existentes en la red.

1.4 OBJETIVO GENERAL

El objetivo general de nuestro proyecto es desarrollar una aplicación en JAVA de fácil manejo, que sirva para escanear redes a su alcance, ver, analizar y basados en los resultados proporcionados, mitigar las diferentes vulnerabilidades de seguridad que existan en una determinada red.

1.5 OBJETIVOS ESPECÍFICOS

Para poder cumplir nuestro objetivo general se deben cumplir los siguientes objetivos específicos:

- Entender el funcionamiento del modelo TCP/IP.

- Investigar la programación orientada a la captura de paquetes en redes de datos locales (LAN).
- Desarrollar una aplicación que presente datos detallados y que su análisis pueda ser de la mejor manera.
- Diseñar un manual de usuario comprensible para que el software sea explotado en todo su potencial.

CAPÍTULO II

FUNDAMENTOS TEÓRICOS

Para comprender el funcionamiento de nuestro proyecto, es necesario conocer conceptos básicos teóricos sobre redes y su administración. Durante el presente capítulo explicaremos brevemente los protocolos que se utilizan para la administración de redes, enfocándonos sobretodo en el protocolo TCP/IP, así como el funcionamiento, la utilidad, importancia y los componentes básicos de la administración de redes.

2.1 ARQUITECTURA DE PROTOCOLOS TCP/IP

La arquitectura usada en la realización de nuestro proyecto es la del protocolo TCP/IP, la cual es la más adoptada para la interconexión de sistemas, y es el resultado del trabajo llevado

a cabo por la Agencia de Investigación de Proyectos Avanzados de Defensa (DARPA por sus siglas en inglés).

A principio de los setenta, después de la construcción de la pionera ARPANET, DARPA comenzó a trabajar en un gran número de tecnologías de transmisión de datos, por lo que el Internet está basado en los protocolos TCP/IP que creó DARPA para mejorar ARPANET [1].

No existe un modelo oficial de referencia TCP/IP, sino que se basó en los protocolos estándares que se han desarrollado, y se las ha llegado a organizar en cuatro capas relativamente independientes (Figura 2.1).

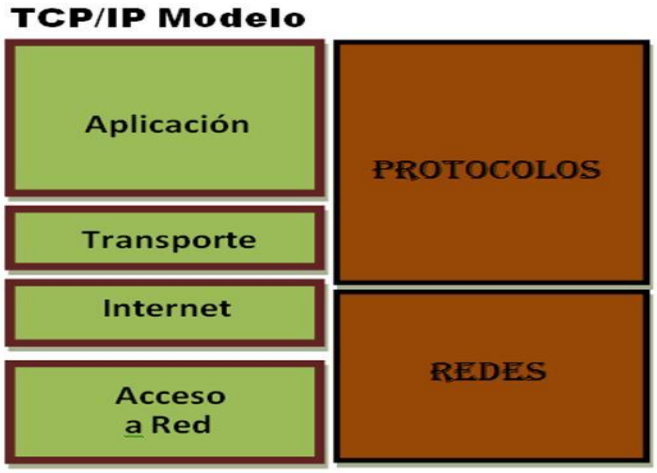


Figura 2.1 Arquitectura del modelo TCP/IP (ANEXO A)

2.2 ADMINISTRACIÓN DE REDES INFORMÁTICAS

Las redes informáticas comprenden una gran variedad de tecnologías ofrecidas por servicios y productos de múltiples fabricantes. El creciente número de usuarios de sistemas informáticos que se apoyan sobre las redes de comunicaciones actuales, han dado origen al campo de la administración de redes para controlar en términos de desempeño, capacidad, utilización, configuración y rendimiento las redes de comunicaciones.

La administración de redes se basa en un conjunto de técnicas que nos permite tener una red operativa, eficiente, segura, constantemente monitoreada con una planeación adecuada y propiamente documentada, siendo estas técnicas de seguridad la función principal del software desarrollado para nuestro proyecto.

Adicionalmente la administración de redes se basa en el acoplamiento de diversas tecnologías, para planificar, modelar, diseñar, configurar y desarrollar redes informáticas con la finalidad de obtener un óptimo desempeño a un costo razonable con la máxima eficiencia, dando así la capacidad de mantener,

corregir, contabilizar, evaluar, y expandir los recursos que la componen.

Para llevar a cabo estas actividades de una forma eficiente la administración de redes emplea un tipo de software (sistema informático para la administración de redes) y hardware (equipos de red) que permiten evaluar, estudiar y monitorear el estado de la red en cualquier momento, para de esta manera dar solución a cualquier tipo de anomalía que se encuentre en la red y dar un buen servicio a los usuarios.

La importancia de la administración de redes se da principalmente en la capacidad de unificar coordinadamente distintas áreas (planificación, modelado, diseño, configuración, desarrollo y mantenimiento), de tal forma que se tenga una red estructurada que nos permita la optimización de los procesos que intervienen en su funcionamiento.

La capacidad de unificar distintas áreas nos permite hacer un análisis y planificación para la ubicación, instalación y configuración de cada uno de sus componentes, la monitorización de la red para conocer las propiedades y el

estado de ésta en un periodo de tiempo deseado con la finalidad de prevenir situaciones de error y reestructurar su diseño debido a la evolución o expansión de la red.

Para que una red sea administrada de la mejor manera, hay que contar con las herramientas necesarias que permitan monitorear la red, para poder visualizar su comportamiento en cada uno de los estados de transmisión de información (Figura 2.2).

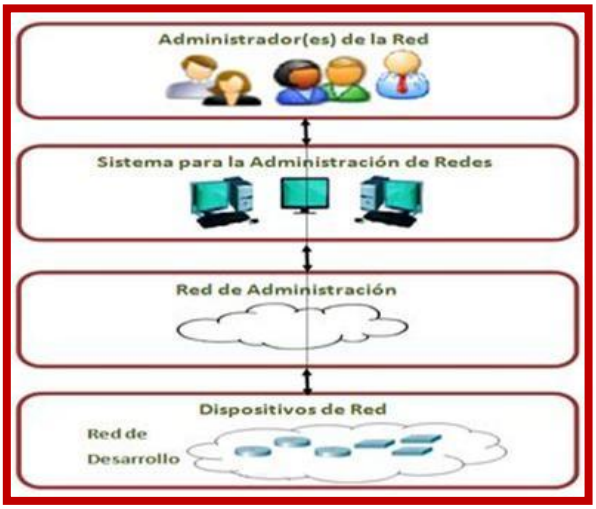


Figura 2.2 Componentes básicos para la administración de redes

Los dispositivos de red representan el componente principal de este esquema, los cuales deben ser administrados, y forman parte del dominio de la red. Éstos, además de su función

principal, proveen soporte para comunicarse mediante protocolos de administración.

El NMS es el software utilizado por el administrador de la red para controlar y monitorizar el comportamiento del dominio administrado, utilizando servicios para obtener información de la red y/o modificar un valor de alguno de los elementos de la misma, ya sea como resultado de una petición explícita o de manera automatizada. Dentro de las funciones que puede desempeñar un NMS para lograr los objetivos mencionados anteriormente, se encuentran: capturar y analizar el tráfico que pasa por la red, recolectar y guardar información diversa de la red, ejecutar diversas tareas programadas, monitorizar, correlacionar, detectar alarmas en la red y descubrir su origen, realizar operaciones de mantenimiento a la red.

Los sistemas para la administración de redes y los dispositivos de red, comúnmente son referenciados por sus roles, “manager” y “agente”, respectivamente. Adicionalmente, hay un agente de administración que está compuesto de tres partes conceptuales: *La interfaz de administración* es la encargada de la comunicación entre el manager y el dispositivo de red. Para

esto, debe soportar el protocolo de administración que define las reglas para el diálogo. Por ejemplo SMTP.

La base de información para la administración (MIB), es un almacén de datos lógico, en donde se guarda información del dispositivo administrado. La información almacenada en la MIB es una abstracción de dicho dispositivo, en la cual se omiten detalles y solamente se exponen datos relativos a la administración.

La lógica central del agente de administración tiene como función principal tomar los datos recibidos por la interfaz de administración, hacer la traducción de la petición en la MIB y posteriormente, convertir esta información en una operación interna del dispositivo, la cual podrá ser leída o asignada [4].

Por todo lo antes mencionado, la principal característica de un NMS es la de brindar una interfaz clara y comprensible a los administradores y traducir las solicitudes de éstos en peticiones que son usadas por los protocolos, para comunicarse con los distintos agentes presentes en el dominio administrado; siendo esta última característica el rol que cumplen los NMS dentro del dominio de red. Dicho rol es conocido como manager.

2.3 ADMINISTRADOR DE LA RED Y ÁREAS FUNCIONALES

Como se aprecia en la Figura 2.3 el administrador de red desempeña muchas funciones, como:

La documentación de la red es llevar control de la topología y configuración inicial de la red registrando todos los cambios en el software y en la topología de la red.

El respaldo confiable y políticas de restauración consiste en configurar un respaldo periódico y apropiado para cada tipo de dato soportado y establecer políticas para la recuperación de los mismos.

Figura 2.3 Funciones del administrador de red

El establecimiento de procesos y políticas operacionales se encarga de la documentación de procedimientos operacionales. *La configuración de parámetros de seguridad* incluye la creación de políticas de seguridad apropiadas y parámetros de acceso dentro de la red [4].

En muchos de los casos, las funciones concernientes a la administración de redes, se dividen en cinco áreas funcionales conocidas bajo el acrónimo inglés FCAPS (Fallas, Configuración, Cuenta, Desempeño y Seguridad por sus siglas en inglés Failure, Configuration, Account, Performance, Security) (Figura 2.4).

Figura 2.4 Áreas funcionales de la administración de redes

Cada una de estas áreas se encarga de un conjunto de las actividades establecidas dentro de la administración de redes:

La Administración de Fallas ***(F)*** se relaciona con el conjunto de mecanismos que permiten la detección, el aislamiento y la corrección de las operaciones anormales de una red o de un sistema de comunicaciones. Nuestro software es capaz de cumplir con esta función.

La Administración de Configuración ***(C)*** abarca las operaciones realizadas para inicializar y modificar las propiedades de configuración de los equipos que componen la red.

La Administración de Contabilidad ***(A)*** contempla las actividades que permiten la identificación de los recursos y cuantificación de los costos de servicios, elaboración de facturas y seguimiento de los cobros y pagos.

La Administración de Desempeño ***(P)*** necesaria para optimizar la calidad de servicio (QoS). Nuestro software es de mucha utilidad en lo que respecta a calidad de servicio y desempeño de una red, el cual se rige por parámetros de rendimientos.

La Administración de Seguridad ***(S)*** se encuentra relacionada con todos los aspectos que puedan afectar la seguridad de la red y su objetivo principal es brindar soporte al sistema de administración de redes en cuanto a las políticas de seguridad.

En esta área se deben distinguir tres tópicos fundamentales que generalizan las funciones de la administración: seguridad física, seguridad de acceso y seguridad de datos, la cual es la parte fundamental de nuestro software [5].

El patrón común de interacción entre managers y agentes, se rige por un modelo asíncrono de petición/respuesta, que es independiente del protocolo de administración de red utilizado. En la mayoría de los casos, el manager realiza una petición (con el fin de obtener información de administración, cambiar propiedades de configuración o realizar actividades de mantenimiento), y posteriormente el agente envía una respuesta con la información solicitada.

SMI y MIBs: La Estructura de Información de Administración (SMI), especifica una manera para definir los objetos administrados y su comportamiento, utilizando el lenguaje ASN.1 para la definición de los objetos. Por su parte, la MIB, es la definición de una base de datos conceptual en donde se mantiene información de todos los MO presentes en el agente.

Operaciones en SNMP: En SNMP, el manager y el agente se comunican a través de mensajes. Un mensaje SNMP consiste en un identificador de versión, un nombre de comunidad y una PDU. Los tipos de PDU definen un conjunto de cinco métodos primitivos sobre los cuales se basa toda la administración. Estos métodos primitivos son: *GetRequest, GetNextRequest, GetResponse, y SetRequest* (ANEXO B).

Los mensajes SNMP tienen una estructura conformada por tres partes: *Versión* que es el número de versión SNMP, *Community*(string) que debe coincidir con el que se ha configurado en el agente, ya que éste funciona como contraseña en el proceso de autenticación de SNMP, *PDU SNMP* el cual consiste en la codificación de la operación SNMP que se desea realizar.

A continuación se explica dichas codificaciones, las cuales también rigieron las operaciones SMNP en nuestro software. (Figura 2.5).

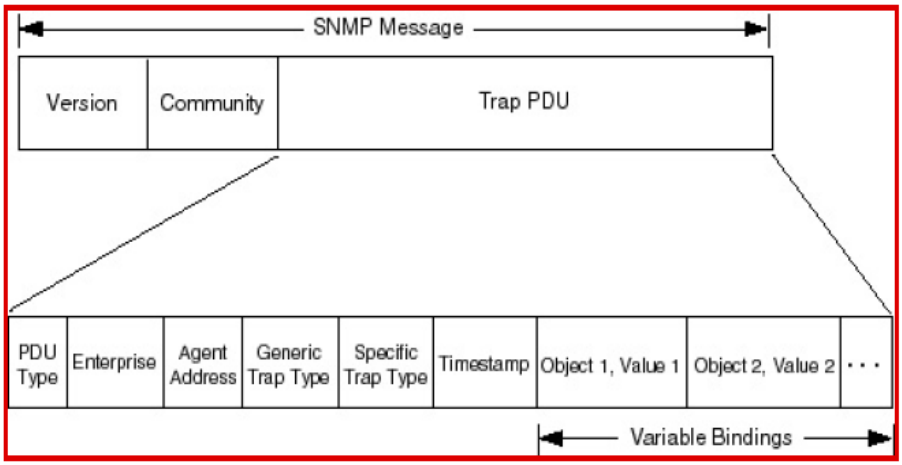


Figura 2.5: PDU Trap [5]

PDU Type: indica el tipo de operación que se está realizando según el valor asignado: 0 para *GetRequest*, 1 para *GetNextRequest*, 2 para *GetResponse* y 3 para *SetRequest*.

RequestID: es un campo de tipo integer, el cual es asignado aleatoriamente y relaciona la petición del manager con la respuesta del agente.

Error Status: es un campo de tipo integer, que indica si la operación ha finalizado exitosamente o si ocurrió algún error, teniendo además cinco condiciones de error predefinidos.

Error Index: Identifica el índice del error que haya ocurrido. En caso que no haya errores, este campo se inicializa en 0 en la respuesta (GetResponse).

Los traps presentan un formato de PDU distinto al de las operaciones mencionadas anteriormente. Los campos que conforman dicha PDU son:

PDU Type: indica el tipo de PDU, el cual es 4 para un trap.

Enterprise: campo de tipo object identifier, que representa la empresa de administración que emitió el trap. Dicho objeto se encuentra definido en el árbol global.

Agent Address: contiene la dirección IP del agente.

Generic Trap Type: provee información acerca del tipo de evento generado (Anexo F).

Specific Trap Type: provee información del trap o evento ocurrido para fabricantes privados, o para ampliar la información del trap genérico (Generic Trap Type).

Timestamp: contiene un valor que representa la cantidad de tiempo transcurrido desde la última vez que fue reiniciado el agente hasta la ocurrencia del trap.

Variable Bindings: presenta una lista de variables bindings, las cuales son una lista de pares nombre/valor que representa los objetos que están siendo solicitados.

CAPÍTULO III

IMPLEMENTACIÓN

En el presente capítulo daremos a conocer la metodología RUP que hemos planteado en nuestro proyecto, la cual propone realizar un cronograma de actividades para dar seguimiento al proyecto; utilizar el entorno de programación NetBeans 6.9 para aplicaciones JAVA; hacer uso de librerías de libre acceso ya existentes en Internet para la captura de paquetes en una red.

3.1 ADAPTACIÓN DE LA METODOLOGÍA DE DESARROLLO

La metodología de desarrollo plantea un esquema de trabajo que hemos adaptado a 4 fases, las cuales son: Análisis, Diseño, Programación y Pruebas (Figura 3.1).

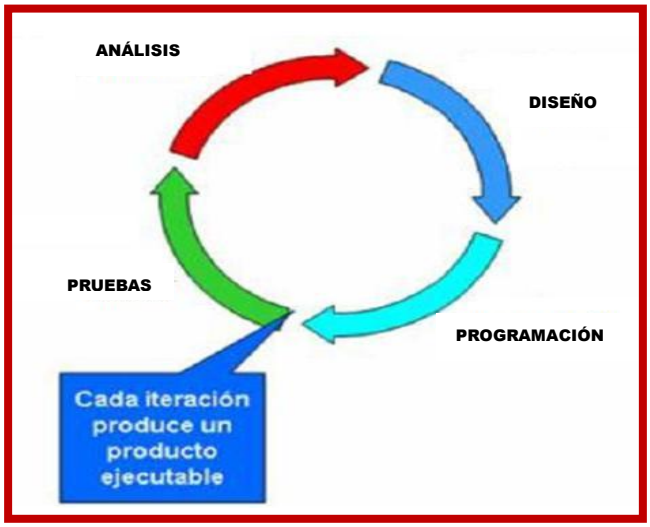


Figura 3.1 Metodología de Desarrollo de software [16]

En cada una de las fases de desarrollo se tomaron en cuenta los siguientes aspectos: Manejo de interrupciones; Comunicación y sincronización entre tareas; Gestión de procesos concurrentes; Respuestas oportunas ante eventos externos; Datos continuos o discretos; Sistemas en tiempo real.

3.2 ANÁLISIS Y PLANIFICACIÓN

Durante la fase de análisis se definieron los requerimientos para cada iteración. Por ser este un proceso iterativo toda fase de análisis es antecedida por una fase de pruebas de la sección o módulo del proyecto que se estaba desarrollando. Dichas pruebas pueden implicar el desarrollo de nuevos requerimientos si no se obtuvieron el resultado o comportamiento esperado. Los resultados fueron analizados en una nueva iteración. Cuando la fase de prueba arrojó los resultados esperados, se procedió a la ejecución de una nueva iteración que incluyó el análisis de requerimientos del nuevo módulo que implementamos.

La fase de análisis y planificación fue documentada mediante la creación y especificación de diagramas de casos de uso, los cuales mostraron por nivel de abstracción todas las funcionalidades (requerimientos) que debieron incluirse en cada módulo.

Se determinó de manera global por cada uno de los módulos de cada iteración los principales requerimientos de la aplicación, los cuales debieron cubrir los objetivos definidos en el Capítulo

1 para dar solución al problema. A partir de este punto, se creó el diagrama de casos de uso, que reflejó las principales funcionalidades de la aplicación y cómo debe interactuar la misma con el usuario para el logro de cada uno de los objetivos. Se estructuró la lista de requerimientos de la aplicación definiendo una interfaz gráfica de usuario basada en el prototipo general de interfaz que se fundamentó en principios de su uso; Se contó también con un manager SNMP; Se dio soporte SNMP del protocolo; Se incorporó utilidades para administración de redes tales como escáner de puertos y sniffer y finalmente se dio soporte para el protocolo IPv4.

Posteriormente a la definición de la lista de requerimientos, se generó la documentación apropiada que incluyó la creación de los diagramas de casos de uso y su especificación. Éstos son mostrados por niveles de acuerdo al grado de abstracción aplicado para cada funcionalidad del sistema.

En la Figura 3.2 se puede observar el nivel 0 del diagrama de casos de uso, el cual refleja el sistema a crear y adicionalmente muestra la interacción con un actor que es llamado “Usuario”. Un Usuario será todo aquel que interactúe con el sistema y

utilice la aplicación, el cual puede ser desde un usuario inexperto hasta un administrador de redes.

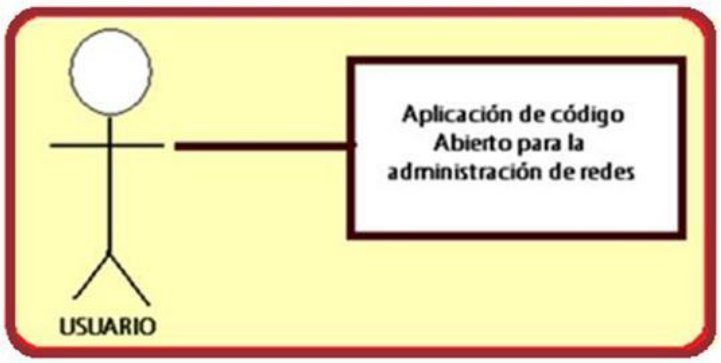


Figura 3.2 Diagrama de casos de uso nivel 0

Durante la iteración uno o General se definieron las principales funcionalidades que debe cumplir la aplicación, obteniéndose 7 casos de uso o módulos principales.

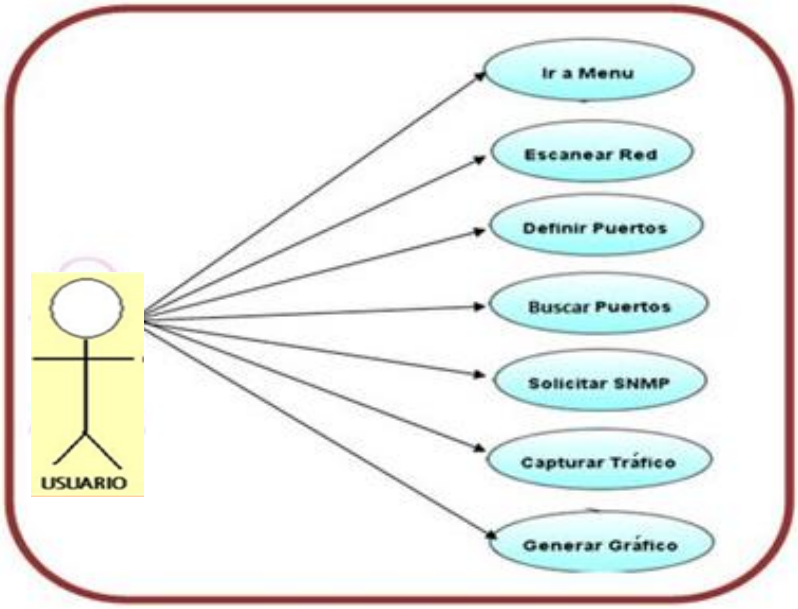


Figura 3.3 Diagrama de Casos de Uso Nivel 1

En la Figura 3.3 puede observarse el diagrama perteneciente al nivel 1 de abstracción, se presenta la especificación de cada caso de uso descrito en el nivel 1. (Tabla 3.1 y 3.2 respectivamente).

Tabla 3.1 Especificación caso de uso (1) Ir a Menú.

Caso de Uso	1. Ir a Menú
Actor	Usuario
Descripción	Presenta el menú de la aplicación, el cual muestra todas las opciones de la aplicación
Flujo Básico	○ Acceder a la aplicación ○ Escoger una opción del menú

En el caso 1 *Ir a Menú* despliega visualmente todas las opciones contenidas en el menú al usuario para proceder a escoger. Se enlaza con la ejecución de cada una de las funciones contenidas en el sistema.

Tabla 3.2 Especificación caso de uso (2) Escanear Red (ANEXO D)

Caso de Uso	2. Escanear Red
Actor	Usuario
Descripción	Muestra todos los equipos que se encuentran conectados en nuestra red.
Flujo Básico	○ Acceder a la aplicación ○ Ejecutar la opción Escanear Red

En el caso 2 *Escanear Red* inicia el Escaneo de la Red, se ejecuta al ser escogida dicha opción del menú de usuario. Tiene como resultado listar las redes activas que se encuentren dentro del rango de alcance.

3.3 DISEÑO

Se creó la estructura lógica del módulo a desarrollar. Para ello, se definieron las clases que iban a interactuar, así como los métodos que fueron incluidos dentro de cada una de las clases. Esta fase fue documentada mediante el uso de diagramas de clases definidos en lenguaje UML. A continuación, nombramos y explicamos algunas de las que consideramos eran las más relevantes.

Clase JnetscanApp1: Clase principal de la aplicación en donde se inicializa la interfaz gráfica del usuario. Como se puede apreciar en la Figura 3.4.

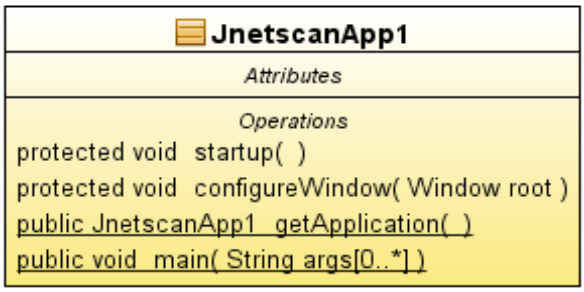


Figura 3.4 Clase JnetscanApp1 (ANEXO E)

Clase JnetscanView **:** Clase que dibuja las diferentes ventanas de las opciones del software, es la parte donde se ejecuta las diferentes funciones que invoca la aplicación principal.

(Figura 3.5)

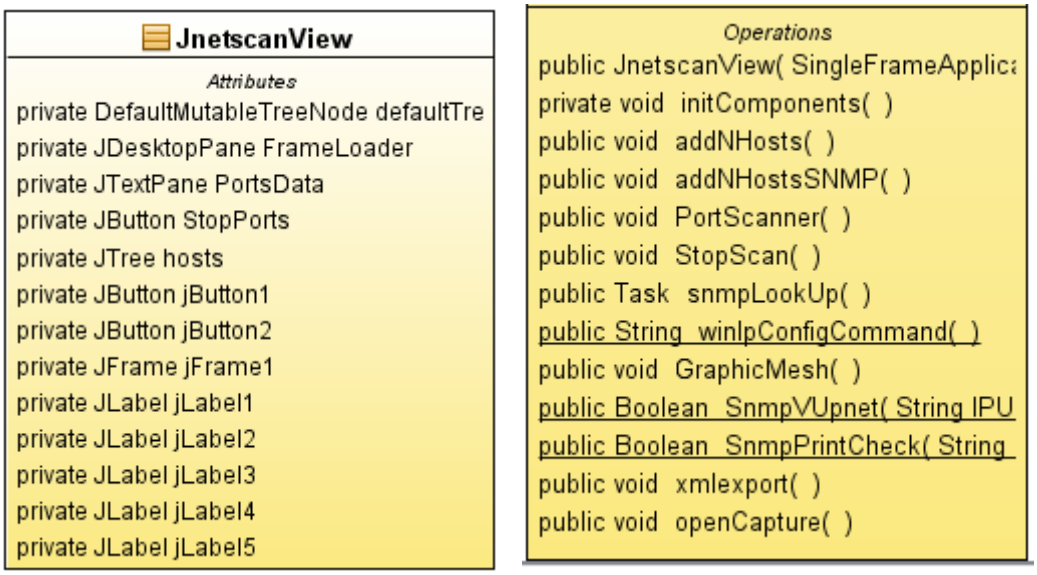


Figura 3.5 Clase JnetscanView (ANEXO E)

3.4 PROGRAMACIÓN

Una vez analizados los requerimientos y diseñada la solución para cubrirlos, se procedió a codificar la solución a fin de implementar las clases diseñadas y se desarrollaron cada uno de los métodos definidos para las clases. Durante esta fase se documentaron los aspectos de codificación más importantes por módulo. Algunos de los métodos implementados son:

El método Startup crea al inicio y muestra la estructura principal de la aplicación.

El método *configureWindow(java.awt.Window root)* inicializa la ventana especificada mediante la inyección de recursos. Ventanas que se muestran en nuestra aplicación están totalmente inicializadas con el constructor de interfaz gráfica de usuario, de modo que esta configuración adicional no es necesaria (ANEXO F).

protected void startup() en el inicio crea y muestra la estructura principal de la aplicación, *protected void configureWindow java.awt.Window root)* este método inicializa la ventana especificada mediante la inyección de recursos. Ventanas que

se muestran en nuestra aplicación están totalmente inicializadas con el constructor de interfaz gráfica de usuario, de modo que esta configuración adicional no es necesaria (ANEXO F).

3.5 PRUEBAS Y ANÁLISIS DE RESULTADOS

Toda iteración culmina con la verificación de la solución creada para asegurar que se cumplan con los requerimientos planteados al inicio y determinar si a partir del desarrollo de la solución se desprenden nuevos requerimientos.

- **ESCANEO DE HOSTS POR SNMP MAPPING**

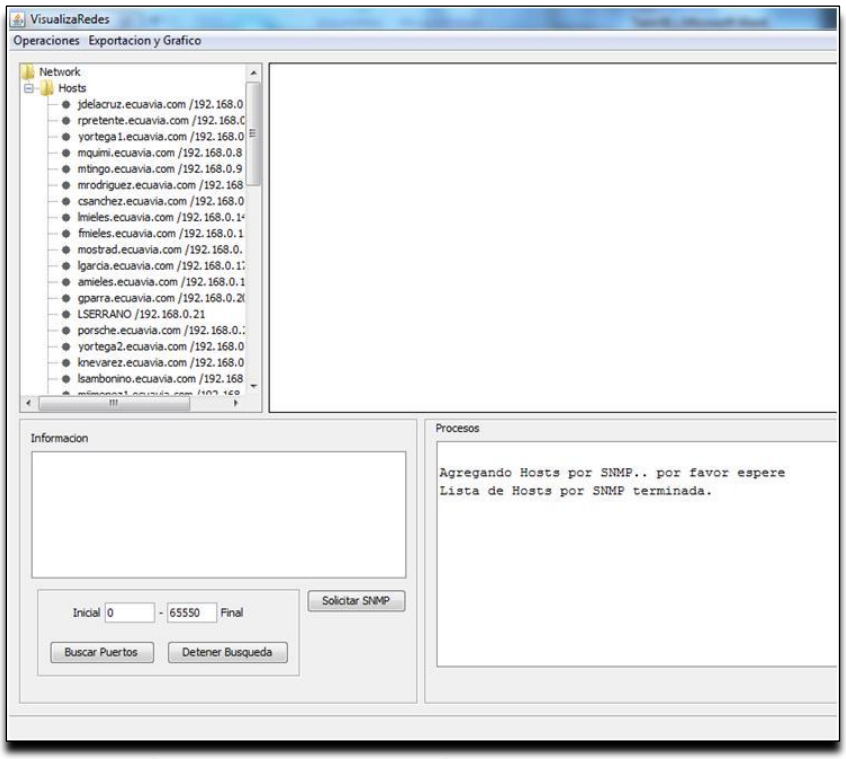


Figura 3.6 Escaneo de Hosts por SNMP

En la parte izquierda de la Figura 3.6 podemos observar que al escanear los hosts por SNMP Mapping se genera una carpeta titulada “host” en la cual se ubican la lista de hosts encontrados durante el escaneo por SNMP, todos estos hosts se encuentran dentro del alcance radial y podremos analizar los puertos abiertos, solicitar SNMP, y generar gráfico de la topología de la red tanto jerárquicamente como de forma radial, este método de escaneo tiene como requerimiento principal deshabilitar el firewall de nuestro computador.

Al probar el software en otro computador se mostró un mensaje de error” Error en la obtención de datos SNMP al host: 192.168.1.1. Ex: java.net.SocketTimeoutException: Receive timed out”, este error se solucionó habilitando todas las funciones SNMP en el computador.

- **BÚSQUEDA DE PUERTOS ABIERTOS**

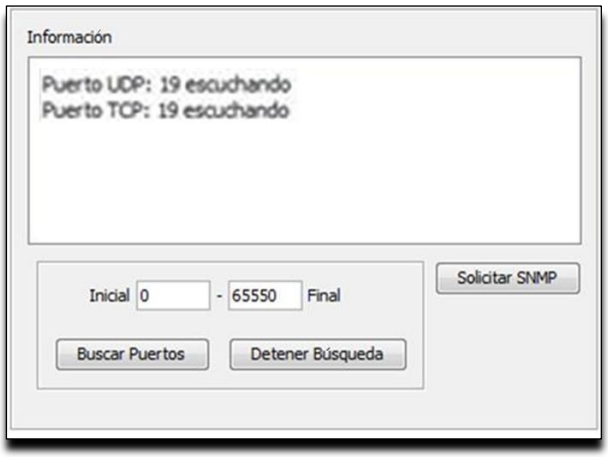


Figura 3.7 Búsqueda de Puertos Abiertos

Al escoger un host dentro de la lista de los hallados por el escaneo SNMP Mapping, podemos observar en la Figura 3.7 que en la sección “Información” ubicada en la parte izquierda de la pantalla podemos buscar los puertos abiertos de la red escogida escribiendo previamente un rango de búsqueda para evitar pérdidas en exceso de tiempo, en este caso se lista el puerto abierto de la red “ecuavia” el número 19 perteneciente a TCP, en todas las pruebas tuvimos un resultado positivo.

- **ESCANEO DE HOSTS POR ICMP**

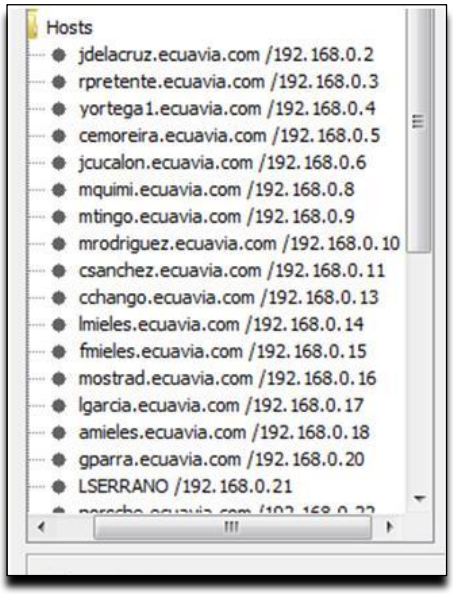


Figura 3.8 Escaneo de Hosts por ICMP

Como se observa en la Figura 3.8 en la parte izquierda de la pantalla se genera al realizar el escaneo de hosts mediante ICMP, una carpeta titulada “host” donde se lista las redes halladas como se indica en la sección “Procesos” ubicada en la parte derecha de la pantalla.

- **GENERACIÓN DE GRÁFICO**

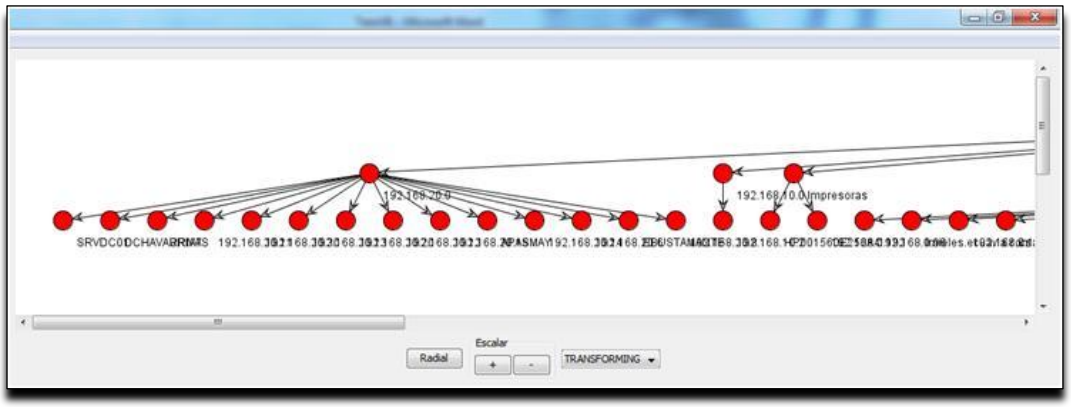


Figura 3.9 Generar Gráfico de Topología

Podemos observar en la Figura 3.9 el gráfico de la red, para lo cual se requiere previamente crear el archivo XML, ingresar un nombre para el archivo del gráfico generado, y ubicar la carpeta donde se lo desea guardar.

En lo que respecta al gráfico de la topología, para asegurarnos que los resultados sean comprendidos aún por personas sin amplio conocimiento en redes, le preguntamos en manera de opinión a 4 personas cercanas a nosotros si el gráfico era lo suficientemente sencillo de entender, obteniendo respuestas positivas en su totalidad, cumpliendo así uno de nuestros objetivos que era realizar una interfaz amigable para el usuario.

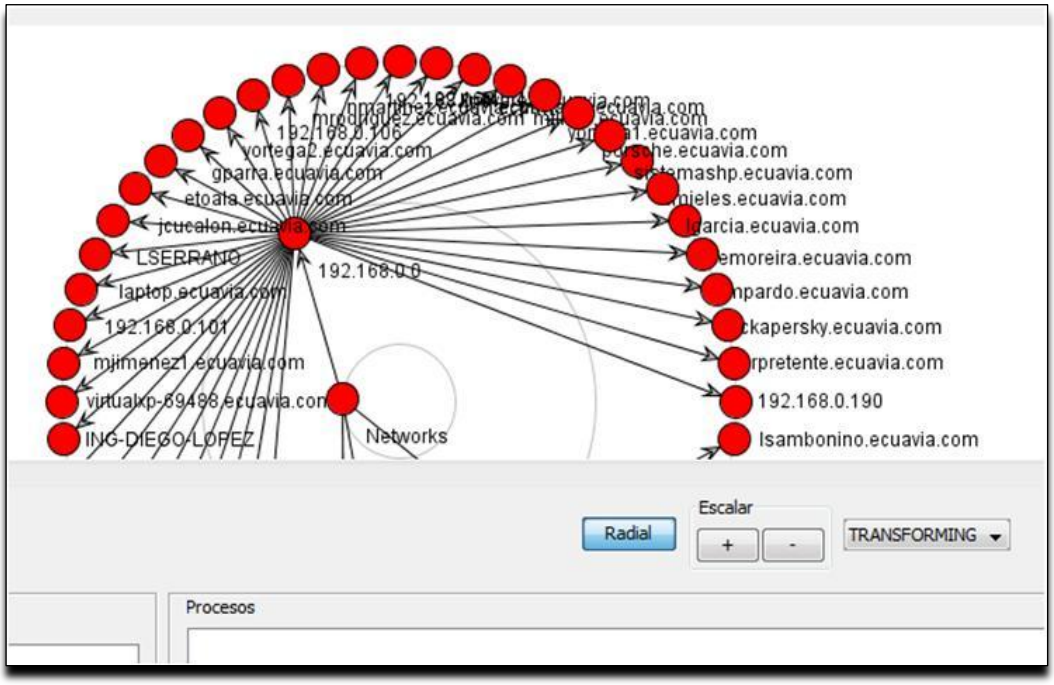


Figura 3.10 Generar Gráfico Radial

Podemos observar en la Figura 3.10 el gráfico de la red en forma radial, el cual se puede alejar o acercar mediante los botones de escalar ubicados en el centro de la pantalla.

- **CAPTURAR PAQUETES DE TRÁFICO**

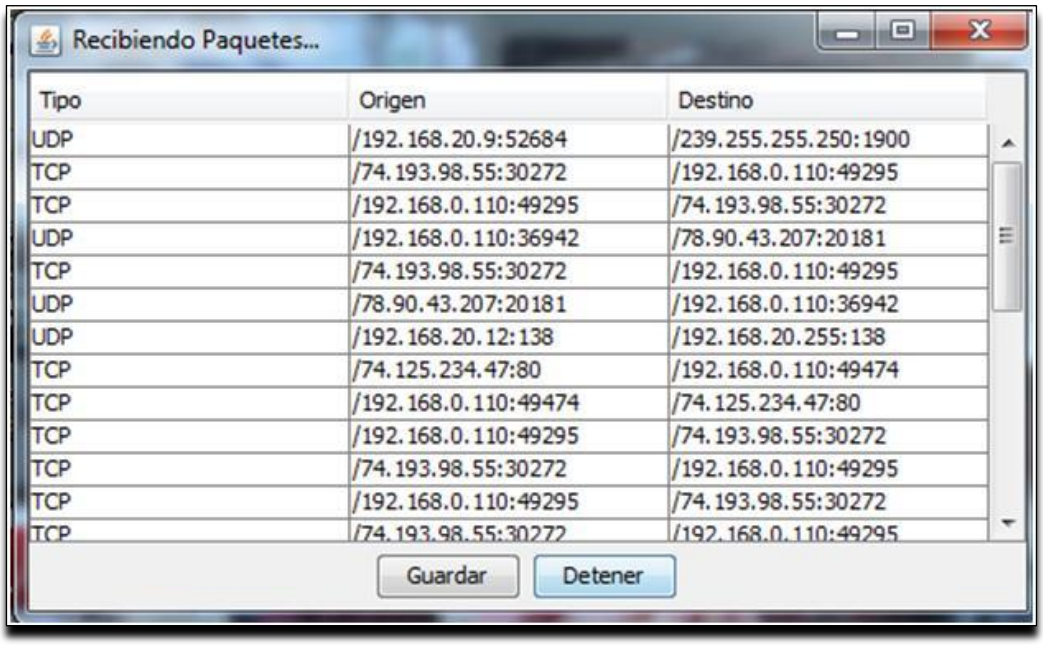


Figura 3.11 Capturar Paquetes de Tráfico

Como podemos observar en la Figura 3.11 se listan los paquetes de tráfico, indicando el tipo de protocolo con el que se transmite, el origen y el destino, del host seleccionado, para lo cual se solicita el SNMP y escogiendo posteriormente la opción “Capturar paquetes de Tráfico” dentro del menú Operaciones

- **CAPTURAR PAQUETES DE TRÁFICO**

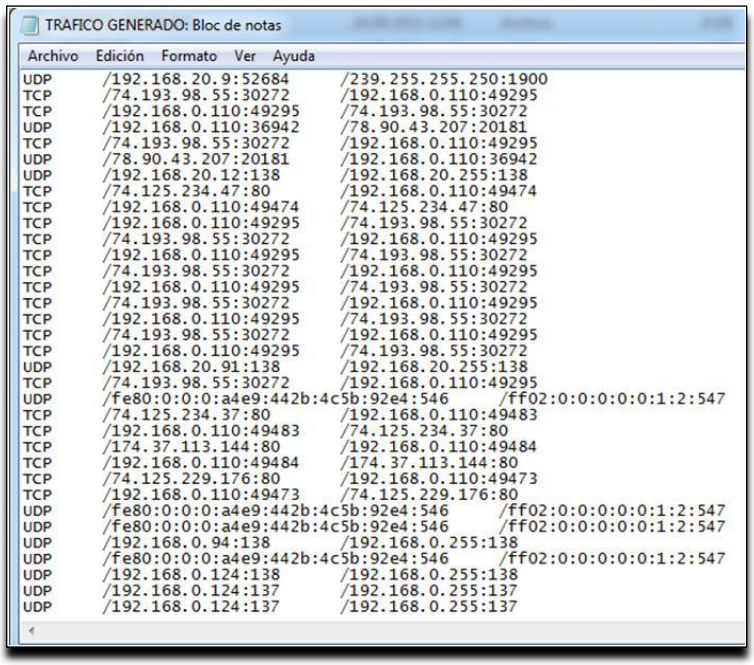


Figura 3.12 Generar Documento de Información de Tráfico

Documento generado de tipo texto con la información del tráfico entrante y saliente de la interfaz de red de la máquina en donde se ejecuta el software.

3.6 PROTOTIPO GENERAL DE INTERFAZ

Con el fin de obtener lineamientos estándar de interfaz gráfica entre los diferentes módulos de la aplicación, se diseñó un prototipo de interfaz general que estableció dichos lineamientos y que se seguirán a lo largo del desarrollo de la aplicación para procurar la usabilidad y mantener la consistencia. Para ello se la ha dividido en 4 secciones (Figura 3.13):

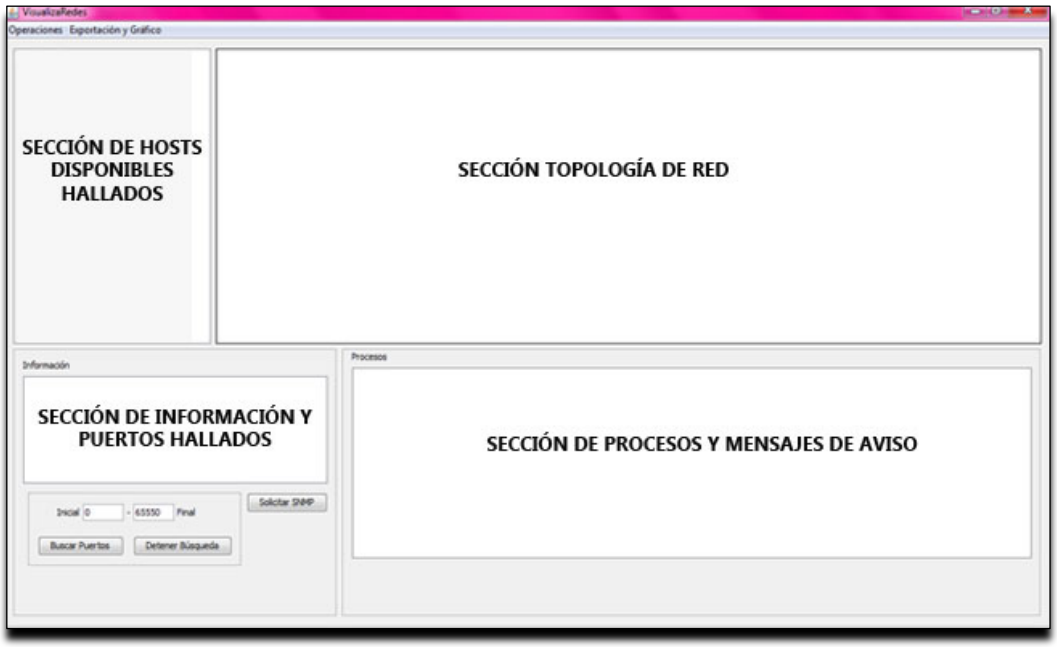


Figura 3.13 Prototipo Interfaz Ventana Principal

Sección de información y puertos hallados en donde se muestra los resultados obtenidos en el escaneo de puertos o saber qué tipo de hardware es mediante el SNMP.

Sección de hosts disponibles hallados que muestra el listado de los hosts encontrados una vez terminado el escaneo sea por SNMP o por ICMP, ya que éstos vienen enumerados por la IP como identificador.

Sección topología de red donde se muestra un gráfico interactivo y escalable en base a los datos que se almacenaron en el archivo XML.

Sección de procesos que indica paso a paso el estado del proceso ejecutado, mensajes de error y mensajes de espera.

CONCLUSIONES

1. Después de extensas investigaciones sobre plataformas que nos permitieran programar un software que cumpla con todos los requerimientos solicitados, decidimos trabajar con Java, ya que es un lenguaje de programación que cuenta con un API extenso, del cual se usó Swing para generar interfaces gráficas basadas en criterios de usabilidad.

2. La utilización de las librerías de código abierto como SNMP en su versión 1 y 2c, JFreeChart, así como la herramienta Jpcap, fue determinante a la hora de poder producir resultados requeridos.

3. Es importante destacar, que durante la implementación se encontró que el protocolo ICMP no es soportado por Java, por lo que se solventó dicha limitación creando algunas herramientas bajo el lenguaje C++, que aparte

de ser muy amplias en cuanto a las funcionalidades que ofrecen, pudieron ser integradas perfectamente a la aplicación.

4. Para lograr capturar los paquetes que transitan por la red que previamente fue escogida de la lista de resultados del escaneo de redes, y para realizar el análisis de los protocolos que los componen utilizamos un sniffer como paquete para generar interfaces gráficas.

5. Decidimos basarnos en la metodología RUP al estructurar el proyecto por plantear una forma de desarrollo de software bien establecido para la programación, la cual consta de 4 fases: Inicio; Elaboración; Desarrollo y Cierre de un proyecto, lo cual nos permitió llevar una excelente organización en el desarrollo del proyecto.

6. Al comparar el funcionamiento de software con funcionalidades similares a nuestro proyecto como es el caso de BuduIP, pudimos observar que al contrario de este, nosotros cumplimos correctamente con una de los desafíos más grandes, el lograr que la interfaz manejada por el usuario del analizador de red sea totalmente amigable y sencilla de usar.

7. Nuestro software ofrece la opción de escanear los protocolos de red especificando previamente el rango de protocolos que se desea buscar, evitando así desgaste innecesario de recursos, convirtiéndolo en más eficiente.

RECOMENDACIONES

1. Verificar que en el computador donde se ejecutará el software tenga habilitado todos los componente SNMP para evitar errores con el escaneo por SNMP mapping.

2. Deshabilitar el firewall de seguridad en el computador, de lo contrario no nos permitirá la correcta ejecución del programa.

3. Evitar colocar un rango extenso de puertos a escanear, ya que aproximadamente el software tarda 3 segundos por cada 5 puertos a escanear, haciendo que se desperdicie innecesariamente tiempo.

4. Una mejora del programa en futuras implementaciones podría ser un entorno gráfico 3D para mejorar el aspecto gráfico y mejorar la base de datos de puertos.

ANEXO A

CAPAS DEL MODELO TCP/IP

CAPA DE APLICACIÓN

Proporciona a la interfaz y servicios que soportan las aplicaciones de usuario. También se encarga de ofrecer acceso general a la red.

- Transferencia de Archivos: TFTP, FTP, NFS
- Correo Electrónico: SMTP
- Conexión Remota: Telnet, rlogin
- Administración de Red: SNMP
- Gestión de Nombres: DNS

CAPA DE TRANSPORTE

Es la encargada de mantener y controlar el flujo de datos entre los nodos que establecen una comunicación, los datos no solo deben entregarse sin errores, sino además en la secuencia que proceda. La capa de transporte se ocupa también de evaluar el tamaño de los paquetes con el fin de que estos tengan el tamaño requerido por las capas inferiores del conjunto de protocolos. El tamaño de los paquetes lo dicta la arquitectura de red que se utilice: TCP orientado a conexión y UDP No orientado a conexión [3].

CAPA DE INTERNET

El propósito de la capa de Internet es seleccionar la mejor ruta para enviar paquetes por la red. El protocolo principal que funciona en esta

capa es IP. Los siguientes protocolos operan en la capa de Internet TCP/IP: ICMP, IP, ARP, RARP.

El Protocolo de mensajes de control en Internet (ICMP) suministra capacidades de control y envío de mensajes. El protocolo de resolución de direcciones (ARP) determina la dirección de la capa de enlace de datos, la dirección MAC, para las direcciones IP conocidas. El IP ejecuta las siguientes operaciones: Define un paquete y un esquema de direccionamiento. (Figura A.1).

Figura A.1 Estructura de una dirección IP

Porción de red.- Identifica la red a la cual pertenece un host.

Porción de host.- Identifica una dispositivo o estación de trabajo dentro de una red.

En nuestro proyecto hace falta descifrar el formato de los paquetes que viajan para el intercambio de información a través de los distintos niveles del modelo TCP/IP. (Figura A.2)

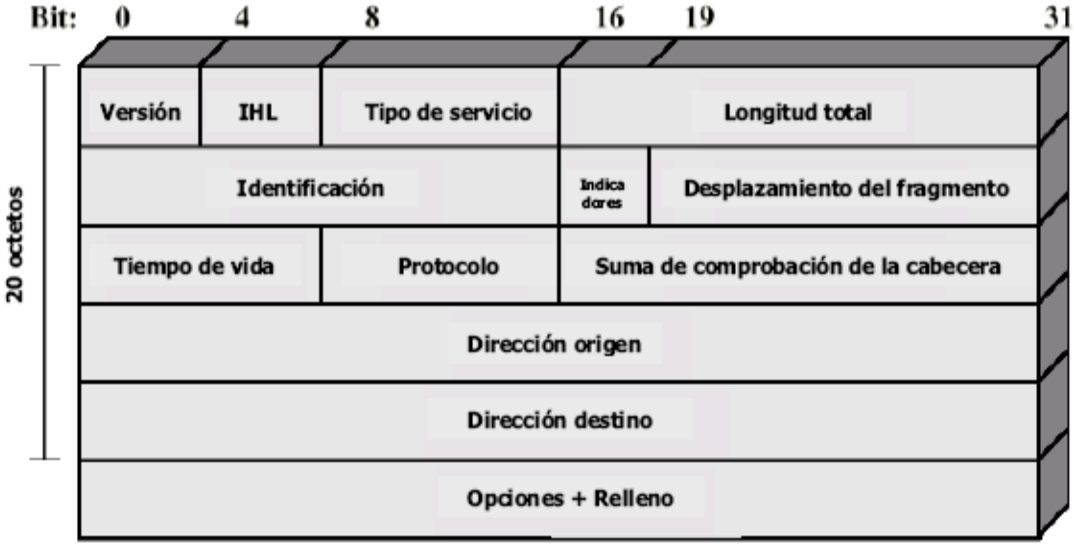


Figura A.2 Datagrama del protocolo IP [1]

Versión (4 bits).- Indica el número de la versión del protocolo, este puede ser versión 4 o versión 6.

Longitud de la cabecera Internet (IHL, Internet Header Length) (4 bits).- Es la longitud de la cabecera expresada en palabras de 32 bits, el valor mínimo es de 5 que corresponde a la longitud mínima de la cabecera de 20 octetos.

Tipo de Servicio (8 bits).- Especifica los parámetros de seguridad, prioridad, retardo y rendimiento.

Longitud Total (16 bits).- Longitud total del datagrama, en octetos.

Identificador (16 bits).- Un número de secuencia que permite identificar de forma única un datagrama.

Tiempo de Vida (8 bits).- Especifica cuanto tiempo en segundos se le permite al datagrama permanecer en la red.

Suma de Comprobación de la Cabecera (16 bits).- Código de detección de errores aplicado solamente a la cabecera**,** es la suma complemento a uno de todas las palabras de 16 bits, este valor se recalcula en cada dispositivo de enrutamiento.

Dirección Origen (32 bits).- Especifica la dirección de red y número de host origen del datagrama.

Dirección Destino (32 bits).- Especifica la dirección de red y número de host destino del datagrama.

CAPA DE ACCESO A LA RED

Es la capa que maneja todos los aspectos que un paquete IP requiere para efectuar un enlace físico real con los medios de la red. Esta capa incluye los detalles de la tecnología LAN y WAN. Las funciones de la capa de acceso de red incluyen la asignación de direcciones IP a las direcciones físicas y el encapsulamiento de los paquetes IP en tramas. Basándose en el tipo de hardware y la interfaz de la red, definirá la conexión con los medios físicos de la misma: Ethernet, Slip&PPP, FDDI, ATM, Frame Relay y SMDS, ARP/RARP, Proxy ARP.

ANEXO B

PDU para GetRequest, GetNextRequest,

GetResponse y SetRequest

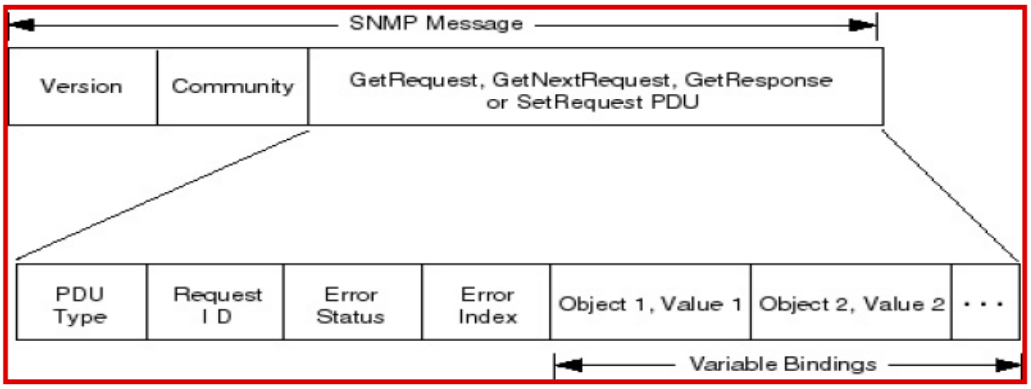


Figura B.1: PDU para GetRequest, GetNextRequest, GetResponse y SetRequest.

GetRequest: Esta operación es utilizada por el manager para recuperar uno o más valores del agente. La petición (*GetRequest*) tiene como parámetro una lista de variables bindings que especifican los objetos solicitados. En este caso, el valor del objeto será null, por tanto, no se conoce [4].

GetNextRequest: La operación *GetNextRequest* es utilizada para recuperar información de administración del agente. A diferencia de *GetRequest*, se especifica el objeto previo al deseado, por tanto, esta operación retornará el valor del OID que sigue en orden lexicográfico al objeto especificado.

SetRequest: La operación *SetRequest* es utilizada para asignar o modificar un valor a un objeto de la MIB que reside en el agente. Esta operación puede ser empleada de tres formas distintas, que son:

modificar algún valor de configuración de un objeto en el agente, crear o eliminar entidades lógicas en la MIB.

GetResponse: Esta operación es utilizada por el agente para responder a las peticiones realizadas por el manager (*GetRequest*, *GetNextRequest* y *SetRequest*). Devuelve el valor del objeto solicitado.

Trap: Los Traps son operaciones utilizadas por el agente de forma asíncrona (sin haberse realizado una solicitud previa), para indicar al manager que ha ocurrido un evento inesperado. Se han definido siete valores posibles. (TABLA B.1), para los diferentes eventos que pueden ocurrir en el agente.

Tabla B.1 TIPOS DE TRAPS

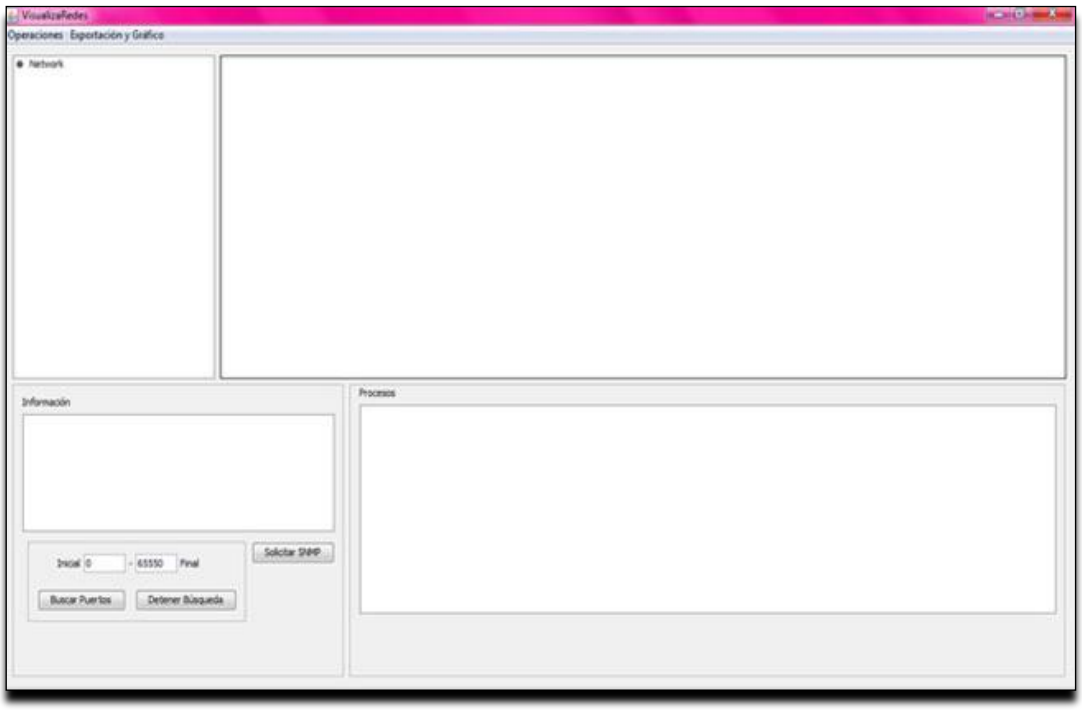
Valor	**Trap**	**Descripción**
0	coldStart	La entidad ha sido reiniciada indicando que la configuración del agente pudo haber sido alterada
1	warmStart	La entidad ha sido reiniciada, pero ni la configuración del agente ni la implementación de la entidad del protocolo ha sido alterada.
2	linkDown	La comunicación con un enlace ha fallado.
3	linkUp	La comunicación con un enlace se ha restablecido.
4	AuthenticationFailure	El agente ha detectado un fallo en la autenticación por parte del manager (comunidad incorrecta).
5	egpNeighborLoss	Un vecino EGP está caído.
6	enterpriseSpecific	Trap no genérico (específico de una empresa de administración).

ANEXO C

Manual de Usuario

Software Para Escaneo de Redes

Para Windows & Linux



Software Para Escaneo de Redes

Información de Contacto

Dirección:

Campus Gustavo Galindo “Escuela Superior Politécnica del Litoral”

Autores:

Diego López

Ma. Verónica Serrano

Soporte Técnico:

maveserr@espol.edu.ec

danlopez@espol.edu.ec

CONTENIDO

Bienvenida

Esta guía contiene información muy importante y necesaria para la apropiada instalación y uso del software, por favor lea bien las instrucciones antes de ejecutarlas.

Introducción del Producto

El software presentado tiene como función principal el proveer a los administradores de red una herramienta gráfica de monitoreo de red de fácil manejo, que le permita analizar y mitigar las diferentes vulnerabilidades de seguridad que existan en la red.

Requerimientos del Sistema

Para que el software se ejecute sin inconvenientes se requiere tener uno de los siguientes sistemas operativos: Windows Xp, Windows Vista, Windows 7 o Linux de todas las versiones que soporte Netbeans 6.9 o sus compatibles.

Especificaciones Técnicas

Programado en Java utilizando Netbeans IDE 6.9 MI, en un sistema operativo Windows Xp

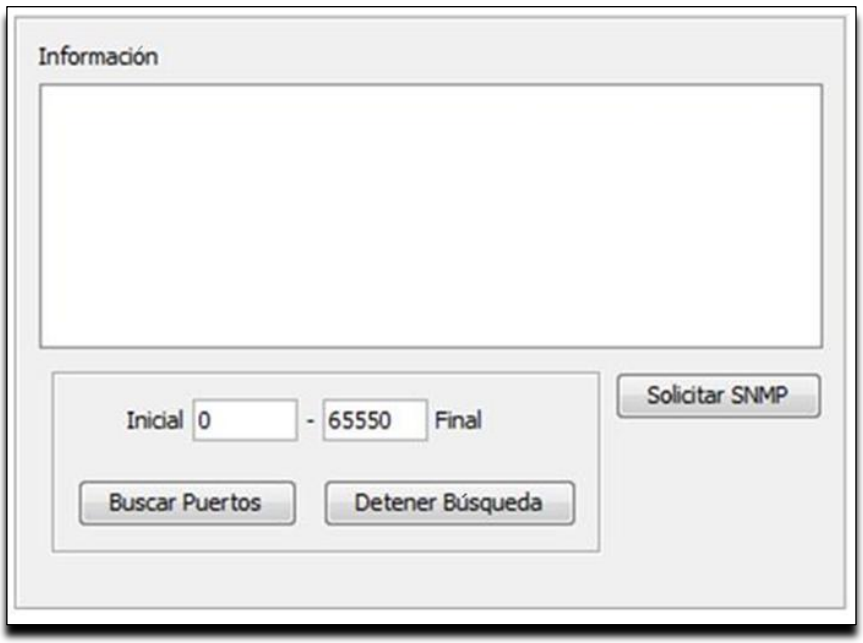
Instalación del Software

Se debe descargar la librería JPCAP de java para capturar y enviar paquetes de red de la página Web http://www.sf.net, para la captura de tráfico de la interfaz de red que tenga la máquina a ejecutar el software.

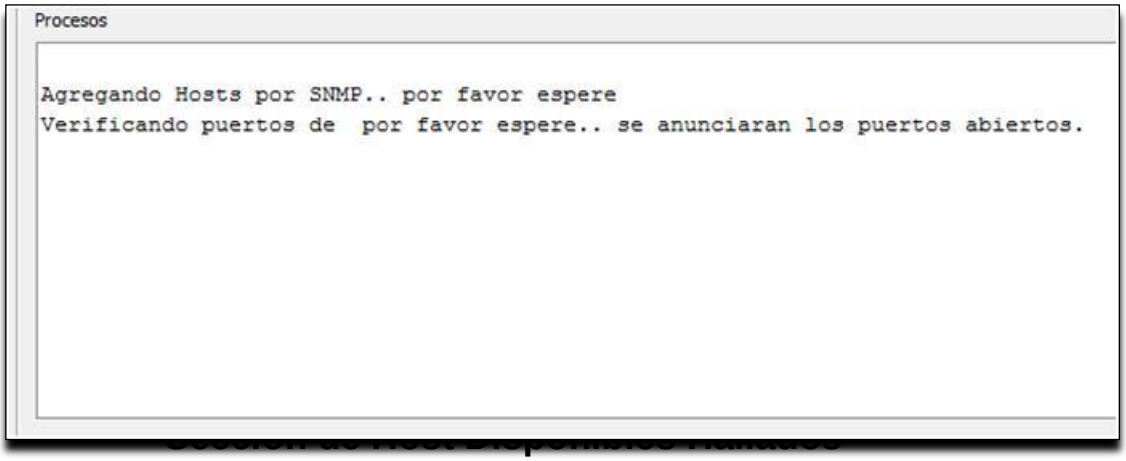
Secciones

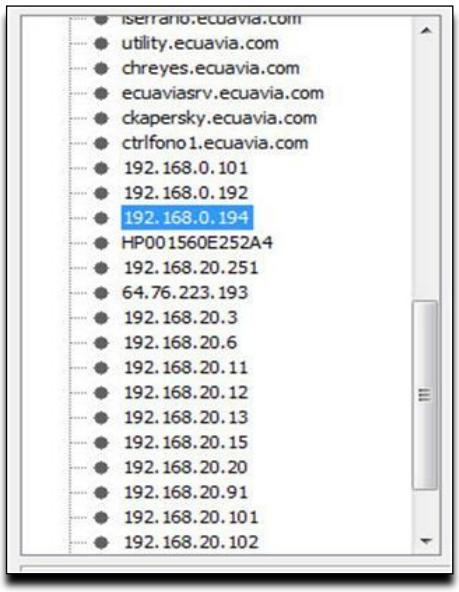
Barra de Menú

Sección de Información y Puertos Hallados

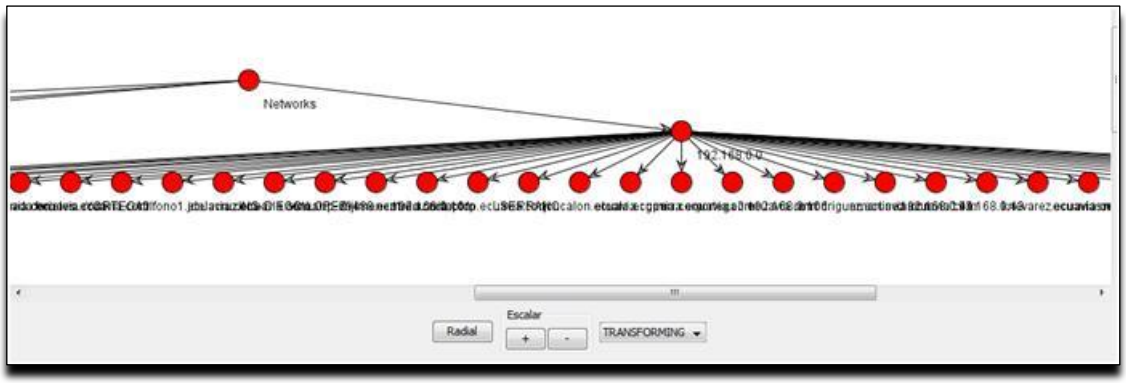


Sección de Procesos





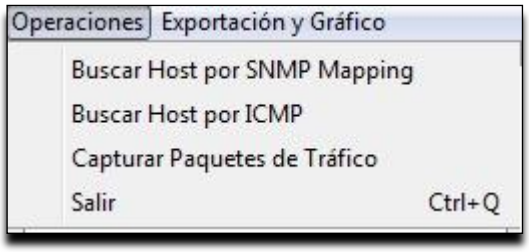
Sección Topología de Red



Operaciones

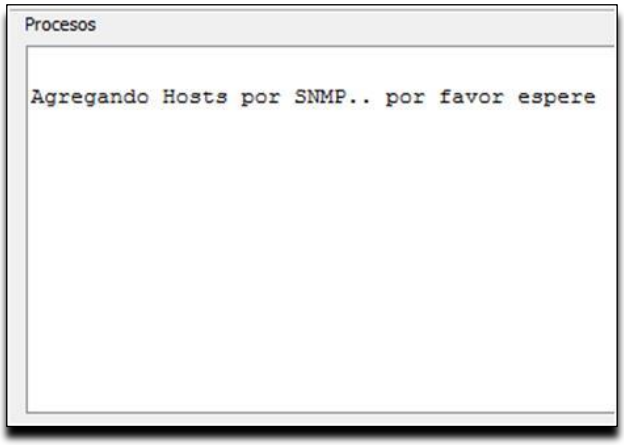
Dentro de la barra de Menú encontramos la opción OPERACIONES.

En esta opción el usuario puede escoger una de las siguientes opciones: Buscar Host por SNMP Mapping, Buscar Host por ICMP, Capturar Paquetes de Tráfico, y Salir.

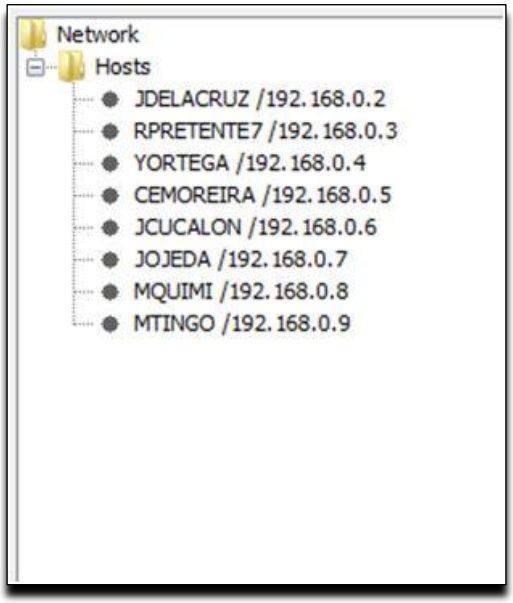


Buscar Host por SNMP Mapping

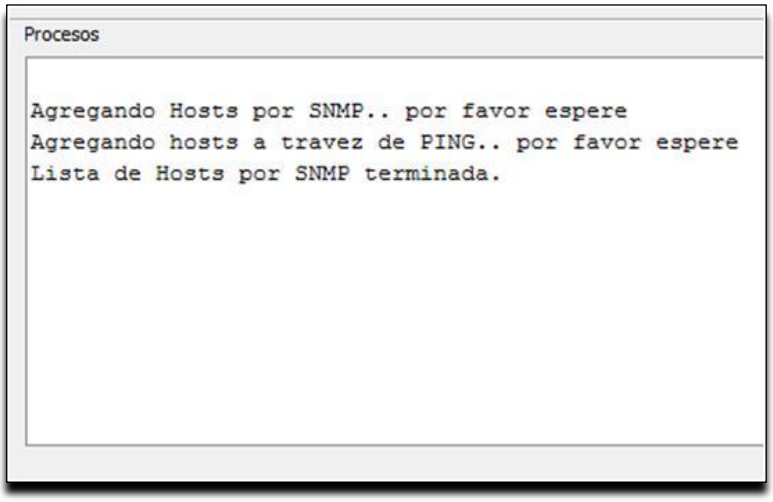
1. *Ir a la Barra de Menú y hacer un clic en la opción Operaciones.*
2. *Hacer un clic en la opción Buscar Host por SNMP Mapping.*
3. *En la ventana de procesos nos aparecerá un mensaje: "Escaneando hosts por SNMP… por favor espere"*



4. *En la ventana de hosts disponibles irá mostrando los hosts que se vayan encontrando a través de SNMP.*

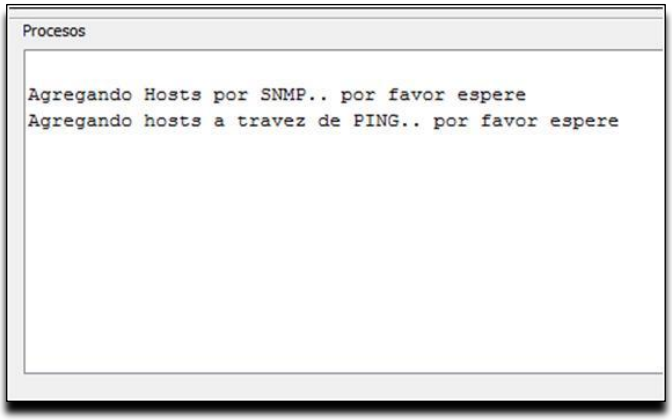


5. *Al finalizar el escaneo de la red, en la ventana de procesos aparecerá un mensaje "lista de Hosts por SNMP terminada."*

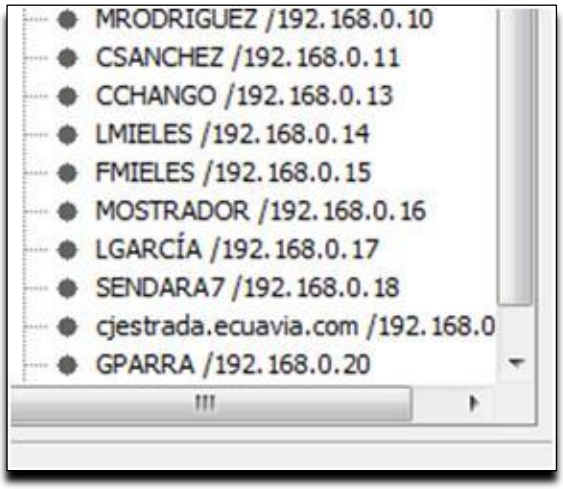


Buscar Host por ICMP

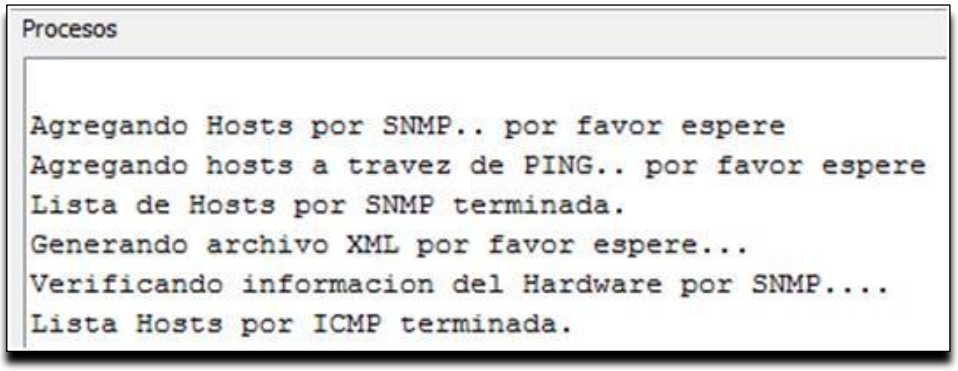
1. *Ir a la Barra de Menú y hacer un clic en la opción Operaciones.*
2. *Hacer un clic en la opción Buscar Host por ICMP*
3. *En la ventana de procesos nos aparecerá un mensaje: "Agregando hosts a través de PING.. por favor espere"*



4. *En la ventana de hosts disponibles irá mostrando los hosts que se vayan encontrando a través de PING*.

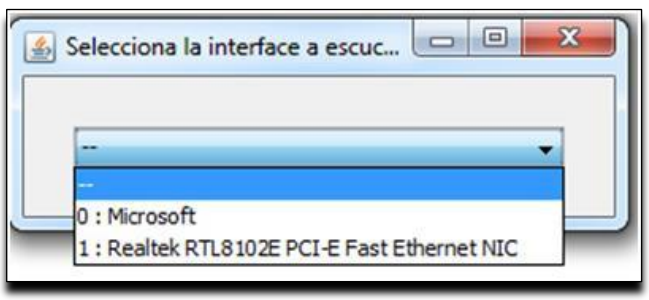


5. *Al finalizar el escaneo de la red, en la ventana de procesos aparecerá un mensaje "lista de Hosts por ICMP terminada."*

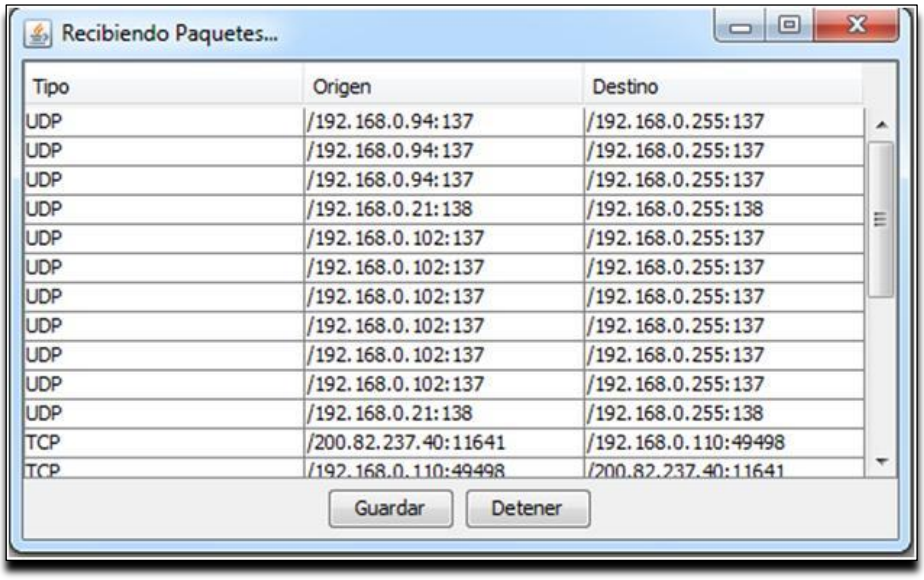


Capturar Paquetes de Tráfico

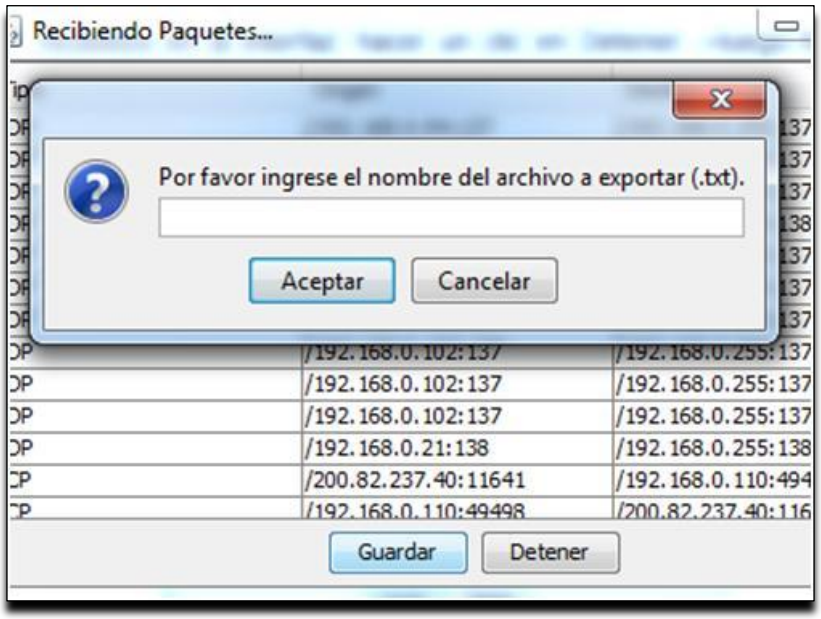
1. *Ir a la Barra de Menú y hacer un clic en la opción Operaciones.*
2. *Hacer un clic en la opción Capturar Paquetes de Tráfico*
3. Aparecerá una nueva ventana que pedirá seleccionar la Interfaz a escuchar
4. Seleccionar la interfaz a la que vamos a capturar el tráfico.



5. Se desplegará una nueva ventana que nos indicará el Tipo, Origen y Destino del tráfico de red.



6. Tenemos la opción de Guardar la información de los paquetes recibidos en la interfaz: hacer un clic en Detener ->luego hacer un clic en Guardar. Se pedirá colocar un nombre al archivo con extensión .txt con el que se va a almacenar.

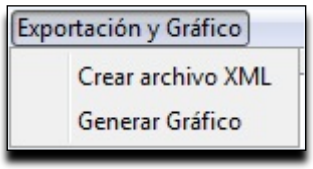


Salir

1. *Ir a la Barra de Menú y hacer un clic en la opción Operaciones.*
2. *Hacer un clic en la opción Salir para cerrar la aplicación.*

Exportación y Gráfico

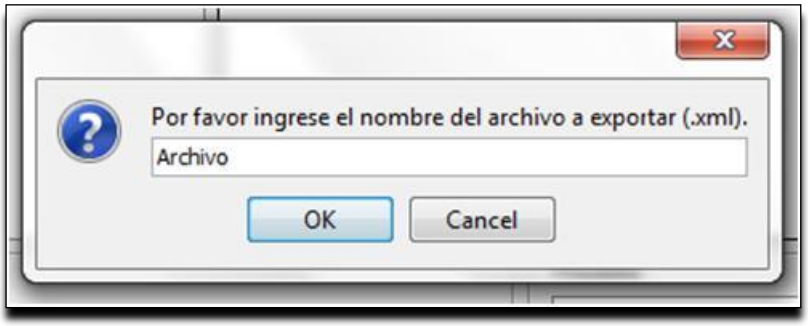
Dentro de la barra de Menú encontramos la opción Exportación y Gráfico



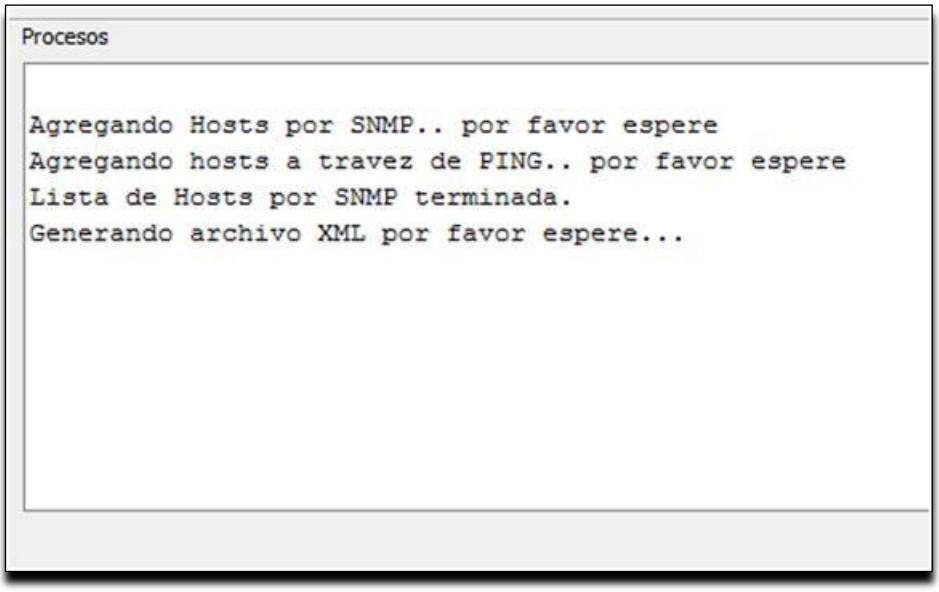
En esta opción el usuario puede escoger una de las siguientes opciones: Crear archivo XML y Generar Gráfico

Crear archivo XML

1. *Ir a la Barra de Menú y hacer un clic en la opción Exportación y Gráfico.*
2. *Hacer un clic en la opción Crear archivo XML.*
3. *Se desplegará una nueva ventana para agregar un nombre al archivo con extensión XML a crear*



4. En la ventana de procesos aparecerá un mensaje: “Generando archivo XML por favor espere…”

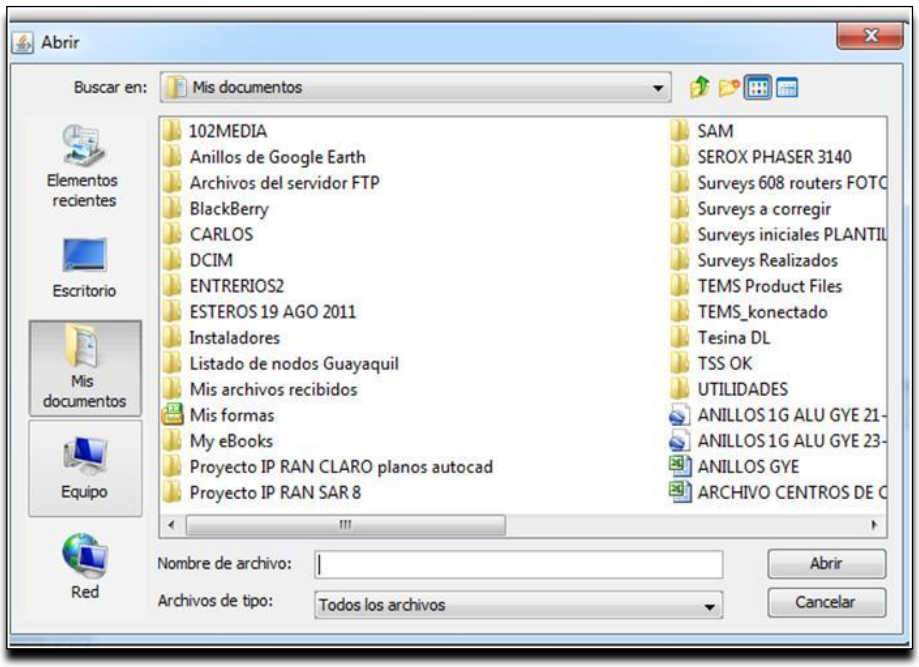


Generar Gráfico

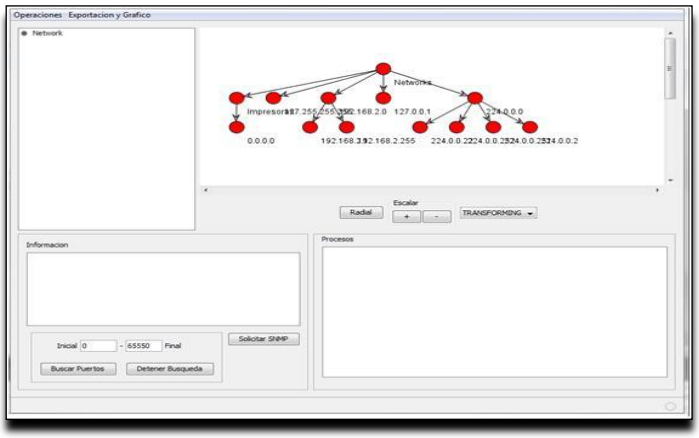
1. *Ir a la Barra de Menú y hacer un clic en la opción Exportación y Gráfico.*

2. *Hacer un clic en la opción Generar Gráfico.*

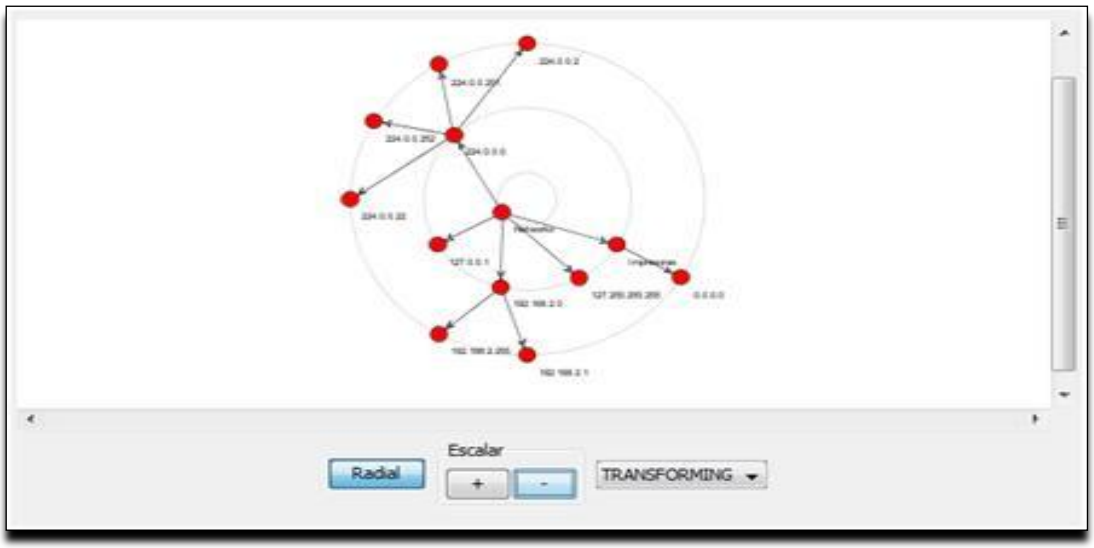
3. *Seleccionar el archivo XML generado.*



4. En la ventana Topología de red aparecerá el gráfico en forma jerárquica de la red escaneada.



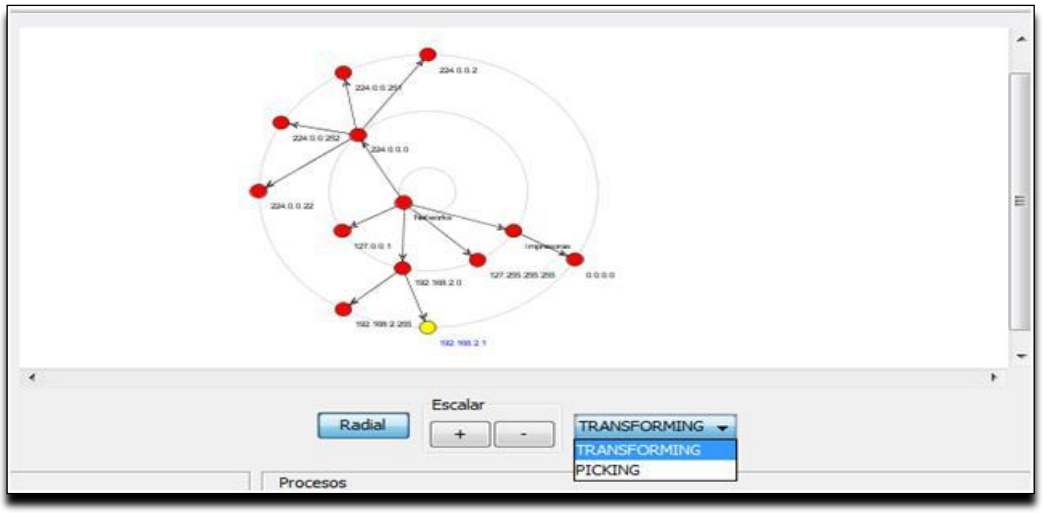
5. El software brinda la opción de mostrar gráficamente en forma de circunferencia la posición de los hosts al hacer clic en el botón radial.



6. Para interactuar con el gráfico tenemos las opciones:

TRANSFORMING Sirve para dirigir al usuario en el gráfico.

PICKING Sirve para seleccionar un Host en el gráfico el cual cambiará a color amarillo para mostrar que es el Host seleccionado.



Buscar Puertos

Se selecciona un host a cual se quiere buscar los puertos que tiene escuchando, para ello se debe indicar un rango a buscar y el resultado se va mostrando en la ventana información.

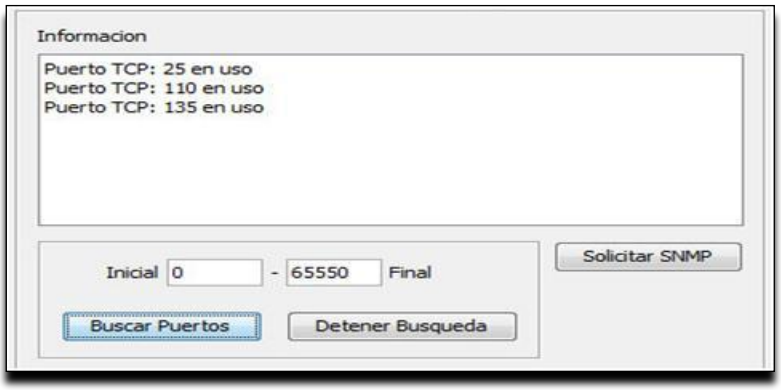


FIGURA C.13 Buscar Puertos

Solicitar SNMP

El usuario selecciona el host a cual desea saber qué tipo de hardware es mediante el SNMP. Cabe recalcar que a veces algunos Host tienen bloqueado el reconocimiento de Hardware mediante SNMP.

ANEXO D

CASOS DE USO

TABLA D.1 Especificación caso de uso (3) Definir Puertos

Caso de Uso	3. Definir Puertos
Actor	Usuario
Descripción	Se define el rango de puertos que se procederá a buscar.
Flujo Básico	o Limita la búsqueda al escaneo de red solo dentro del rango de puertos o Da paso a la opción Escanear Red

En el caso 3 *Definir Puertos* se pide al usuario ingresar los valores de rango de puertos los cuales limitará la búsqueda otorgándole al sistema la máxima eficiencia como es el economizar tiempo y recursos del ordenador. Tiene como resultado final establecer límites del proceso.

TABLA D.2 Especificación caso de uso (4) Buscar Puertos

Caso de Uso	4. Buscar Puertos
Actor	Usuario
Descripción	Nos permite analizar los puertos TCP y UDP que se encuentran activos en un Host
Flujo Básico	o Acceder a la opción escanear puertos o Iniciar el escaneo de puertos del rango seleccionado, indicando los nombres de los diez puertos más comúnmente utilizados.

En el caso 4 *Búsqueda de* Puertos se inicia el escaneo de puertos abiertos que se encuentren establecidos en el rango obtenido en el caso de uso previo**.**

TABLA D.3 Especificación caso de uso (5) Solicitar SNMP

Caso de Uso	5. Solicitar SNMP
Actor	Usuario
Descripción	Nos permite realizar consultas SNMP de un host seleccionado
Flujo Básico	○ Acceder a la opción SNMP ○ Especificar los parámetros de la consulta ○ Seleccionar la operación SNMP a realizar

En el caso 5 *Solicitar* SNMP acceden a las opciones SNMP, y simplemente esta opción lo que hace es ver información del equipo que se quiere consultar como su tipo de hardware, cabe recalcar que si el equipo a consultar tiene bloqueado el tráfico entrante por medio de su firewall esta información no se podrá visualizar.

TABLA D.4 Especificación caso de uso (6) Capturar Tráfico

Caso de Uso	6. Capturar Tráfico
Actor	Usuario
Descripción	Capturar tráfico de red, el cual se entiende como los paquetes entrantes y salientes.
Flujo Básico	○ Otorga una idea del tráfico que viaja por la red analizada, y el protocolo que se utilizó. ○ Analizar dichas capturas en hexadecimal para obtener todos los datos posibles de las cabeceras IP y segmentos TCP, además de comprender de forma más visual los conceptos TCP/IP.

En el caso 6 *Capturar Tráfico* se accede a una de las opciones del Menú, capturar paquetes de tráfico, luego se selecciona la interface de red, esto nos mostrará todo el tráfico entrante y saliente desde y hacia un host destino, nos muestra los puertos que se están utilizando para enviar información. Se puede guardar esa información colocando parar la captura de paquetes y luego guardar en un archivo de texto.

TABLA D.5 Especificación caso de uso (7) Generar Gráfico

Caso de Uso	7. Generar Gráfico
Actor	Usuario
Descripción	Nos permite generar el gráfico de la red previamente escaneada.
Flujo Básico	○ Visualizar las redes escaneadas ○ Definir los puertos y hosts encontrados dentro de la red.

En el caso 7 *Generar* Gráfico representará gráficamente lo obtenido en el caso de uso 5, el cual nos devolvió previamente el escaneo de las redes activas y detectadas dentro del rango de alcance. Se visualizará la información obtenida al final de la ejecución del sistema.

ANEXO E

DISEÑO

Clase JnetscanApp1： Clase principal de la aplicación en donde se inicializa la interfaz gráfica del usuario. Como se puede apreciar en la (Figura E.1)

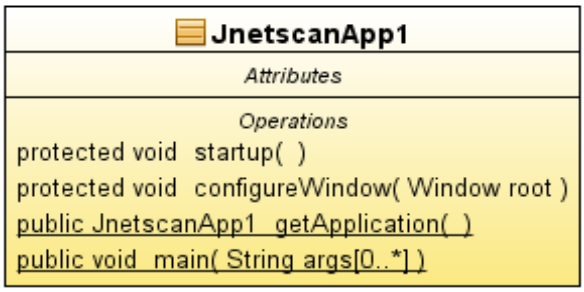


Figura E.1 Clase JnetscanApp1

Clase JnetscanView : Clase que dibuja las diferentes ventanas de las opciones del software, es la parte donde se ejecuta las diferentes funciones que invoca la aplicación principal. (Figura E.2)

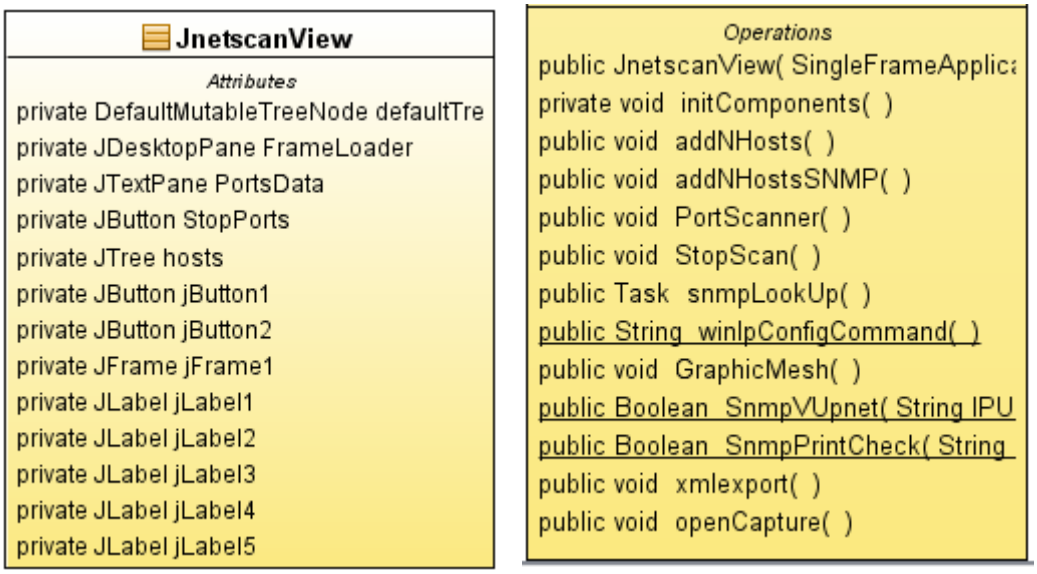


Figura E.2 Clase JnetscanView

Clase Selector: Clase que nos sirve para seleccionar los campos del árbol de los hosts y de esta manera darle los usos requeridos. (Figura E.3)

Figura E.3 Clase Selector

Clase SnmpLookUp: Clase que nos sirve para obtener la información del hardware a través del SysName del SNMP, cabe recalcar que si el firewall está activo no nos permitirá encontrar dicho host. (Figura E.4)

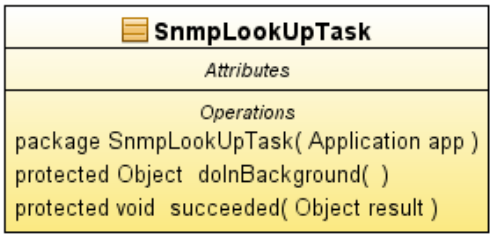


Figura E.4 Clase SnmpLookUpTask

Clase NetworkMesh: Clase en donde se encuentra el componente visual y el procesador de la gráfica, aquí es donde se realizan las diferentes funciones del gráfico como exportar gráfico, obtener información del gráfico, cambiar la vista del gráfico, manipular el gráfico. (Figura E.5)

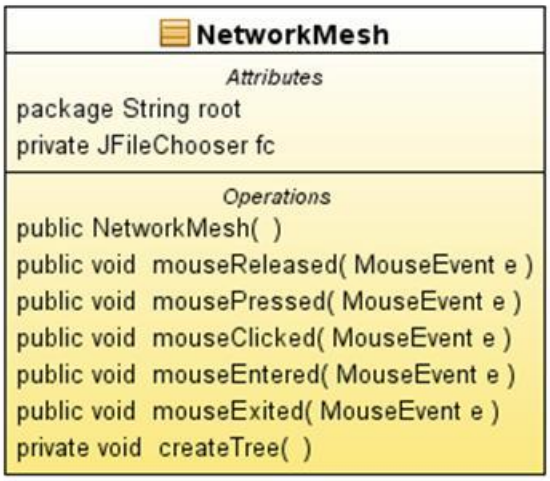


Figura E.5 Clase NetworkMesh

Clase XMLWriter

Clase en donde se almacenan todos los datos obtenidos de los host para poder realizar su gráfico, mediante el resultado de estos datos es cómo se realiza el gráfico de una forma más exacta. (Figura E.6)

Figura E.6 ClaseXMLWriter

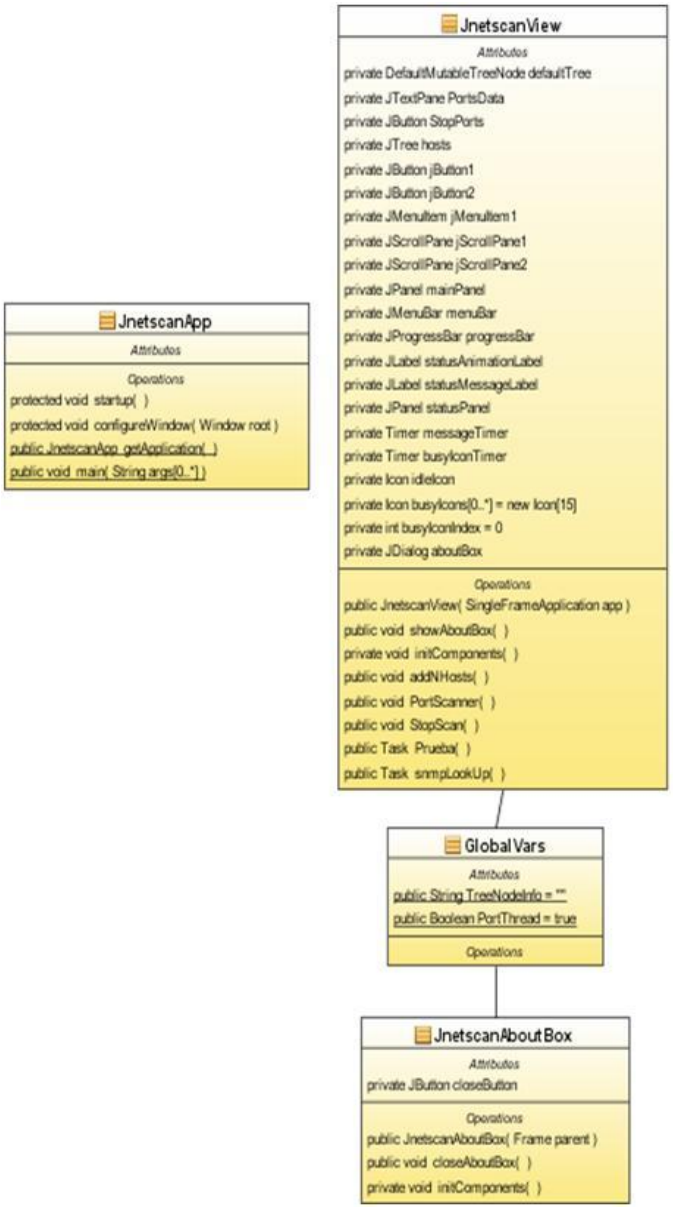


Figura E.7 Diagrama de clases

ANEXO F

MÉTODOS UTILIZADOS

public static *JnetscanApp1 getApplication():* Obtiene un estático conveniente para la instancia de la aplicación.

public static void *main(String[] args):* Método principal para ejecutar la aplicación.

public void *addNHosts():* Agrega los hosts dentro de la red interna a través del ICMP, si está bloqueado por firewall, este no aparecerá en la lista de hosts.

public void addNHostsSNMP() **throws** IOException: Agrega los hosts preguntando a la tabla de ipNetToMediaNetAddress en el router a través del SNMP. Si la opción de SNMP en el router está desactivada, no se podrán obtener datos.

public void *PortScanner():* Scanner de puertos TCP y UDP en un Host especifico requeridos por Rango.

public void *StopScan()* **throws** IOException: Para el escaneo de puertos TCP y UDP cuando se ha llegado al límite del rango especificado.

public *Task snmpLookUp():* Obtiene la información del Hardware a través del SysName del SNMP.

public static *String winIpConfigCommand():* Usa dos métodos diferentes para obtener el Gateway de nuestra máquina usando comandos nativos del OS (Windows ipconfig/netstat)

public void GraphicMesh() **throws** IOException: Llamada al API de generación del gráfico hacia un DesktopPane.

public static Boolean *SnmpVUpnet(String IPU):* Función para la verificación de redes superiores a la local y poder trazar una topología más correcta.

public static Boolean *SnmpPrintCheck(String IPU):* Revisa a través de la tabla de rutas del Hardware si es una impresora o print server.

public void *xmlexport()* **throws** *IOException:* Sirve para guardar un archivo XML, del cual se extraerá toda la información almacenada en el router para poder realizar el trazado en base a los datos proporcionados

public void *openCapture():* Permite capturar todos los paquetes que se transmiten en la interfaz en que se está corriendo el software, haciendo las funciones de un sniffer.

public *NetworkMesh():* Permite crear el gráfico permitiéndonos ver de qué manera están interconectados cada nodo en la red.

private void createTree(): Sirve para crear el árbol de todos los nodos escaneados y que están disponibles en nuestra red.

ANEXO G

PUERTOS BIEN CONOCIDOS

Los puertos representan el destino de las conexiones lógicas usadas en conversaciones de larga duración, por protocolos como TCP y UDP. La abstracción de puerto surgió con la necesidad de proveer servicios a usuarios desconocidos, para ello un puerto de servicio debe ser definido. La asignación de puertos es válida tanto para TCP y UDP. La cantidad de puertos asignados, actualmente se encuentra en el rango de 0 a 1023.

Listado de protocolos y puertos más importantes.

Keyword	Decimal	Description
echo	7/tcp	Echo
echo	7/udp	Echo
ftp-data	20/tcp	File Transfer [Default Data]
ftp-data	20/udp	File Transfer [Default Data]
ftp	21/tcp	File Transfer [Control]
ftp	21/udp	File Transfer [Control]
telnet	23/tcp	Telnet
telnet	23/udp	Telnet
smtp	25/tcp	Simple Mail Transfer
smtp	25/udp	Simple Mail Transfer
tftp	69/tcp	Trivial File Transfer
tftp	69/udp	Trivial File Transfer
www-http	80/tcp	World Wide Web HTTP
www-http	80/udp	World Wide Web HTTP
nntp	119/tcp	Network News Transfer Protocol
nntp	119/udp	Network News Transfer Protocol

SNMP es el protocolo para administración de redes más popular. Está diseñado para trabajar en la capa de aplicación y utiliza los servicios de transporte UDP, por medio de los puertos 161 para intercambio de datos y el

puerto 162 para alertas. Este protocolo está compuesto por un conjunto de funciones simples, las cuales proveen capacidades básicas de administración, tales como obtener y modificar el estado de los dispositivos administrados. Se basa en la interacción manager/agente descrita en el capítulo II y mantiene el modelo asíncrono para la comunicación entre éstos.

Desde sus primeras versiones hasta la actualidad, ha tenido gran aceptación debido a su sencillez. SNMP está pensado para administrar todo tipo de dispositivos que soporten el protocolo. SNMP trata a los recursos administrados como objetos administrados (MO).

ANEXO H

CAPTURAS DE PANTALLA DE PRUEBA

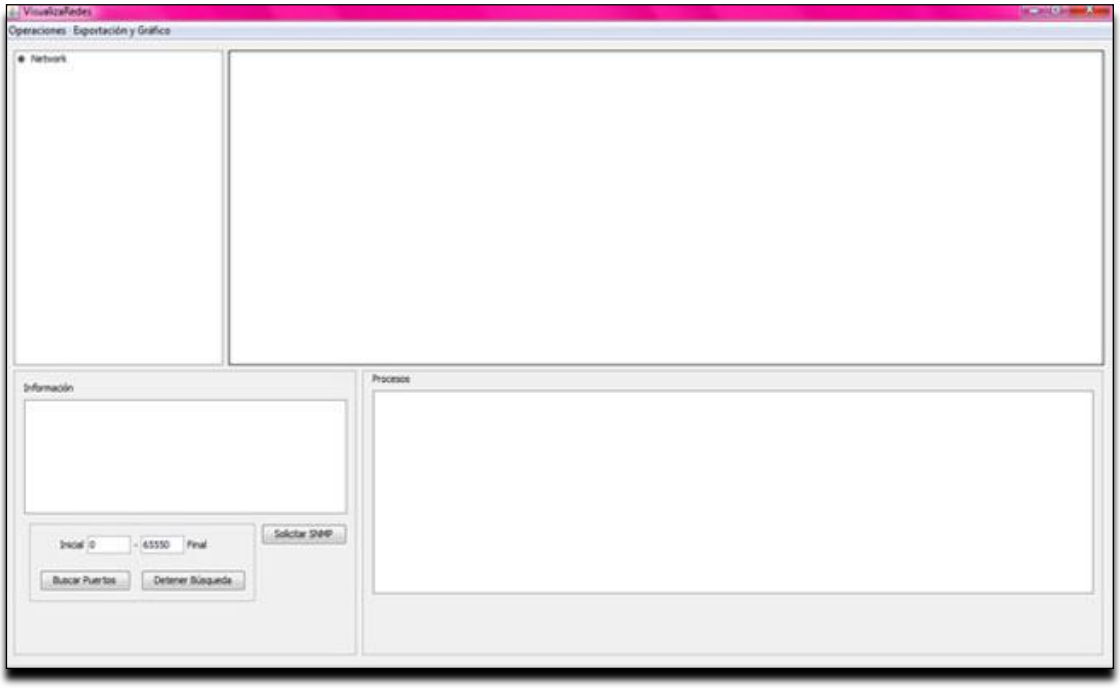


Figura H.1 Prototipo Interfaz Ventana Principal

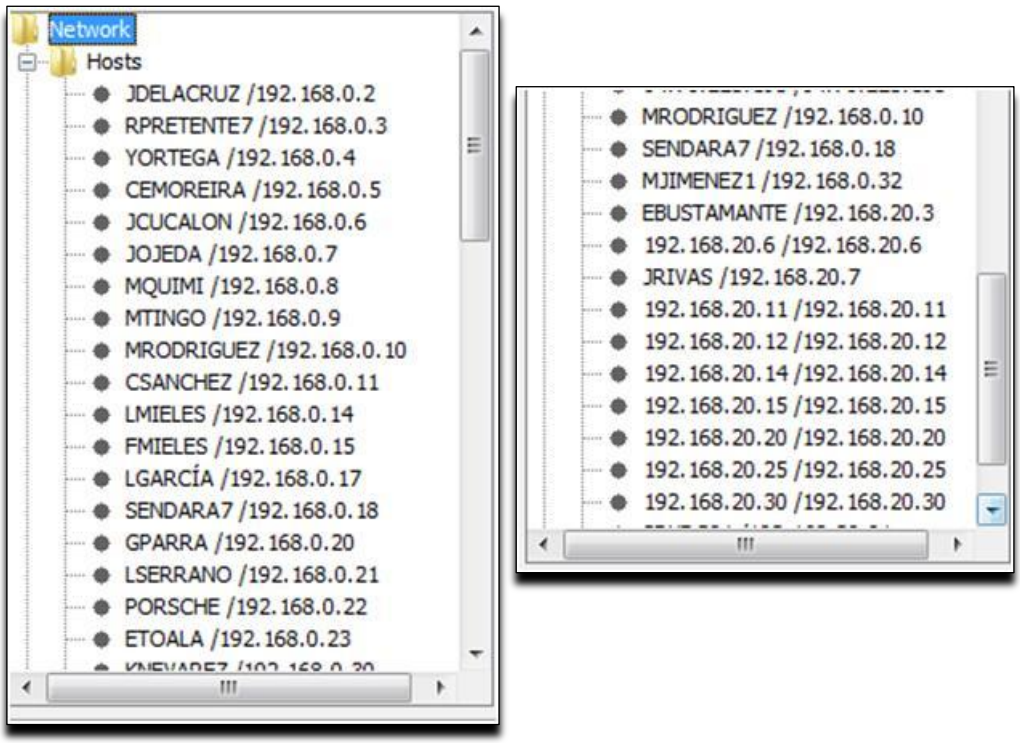


Figura H.2 Listado de Host Disponibles en una Red por SNMP

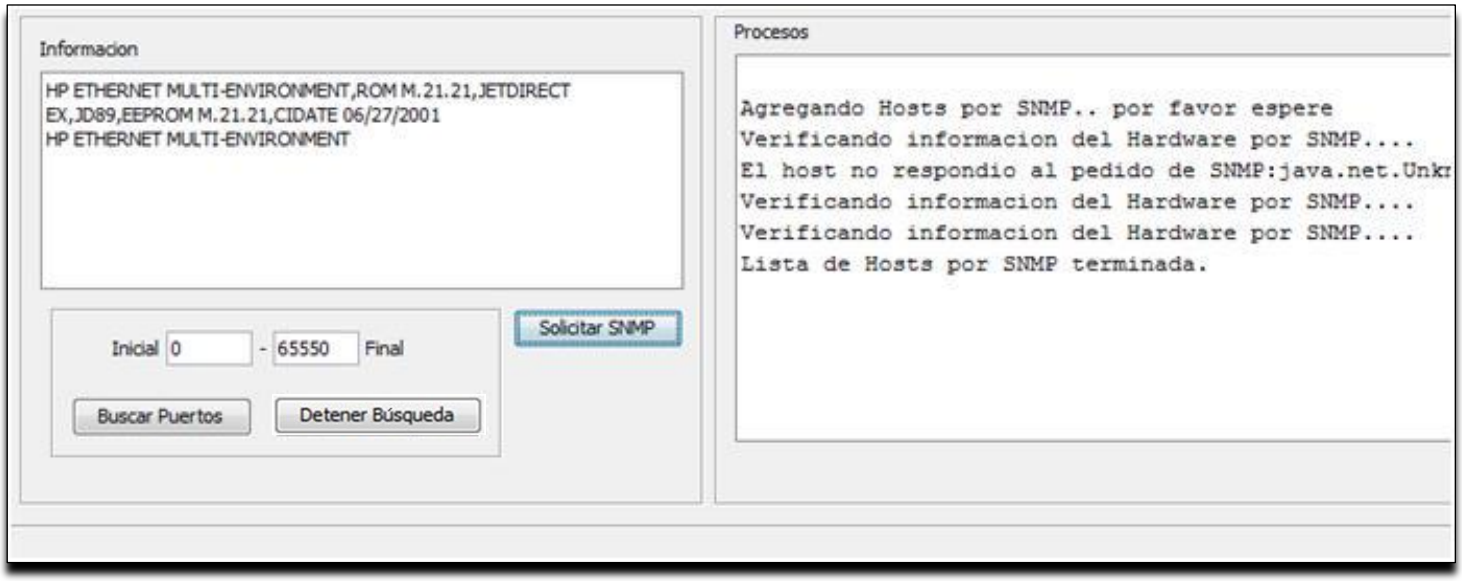


Figura H.3 Tipo de hardware de un host mediante SNMP

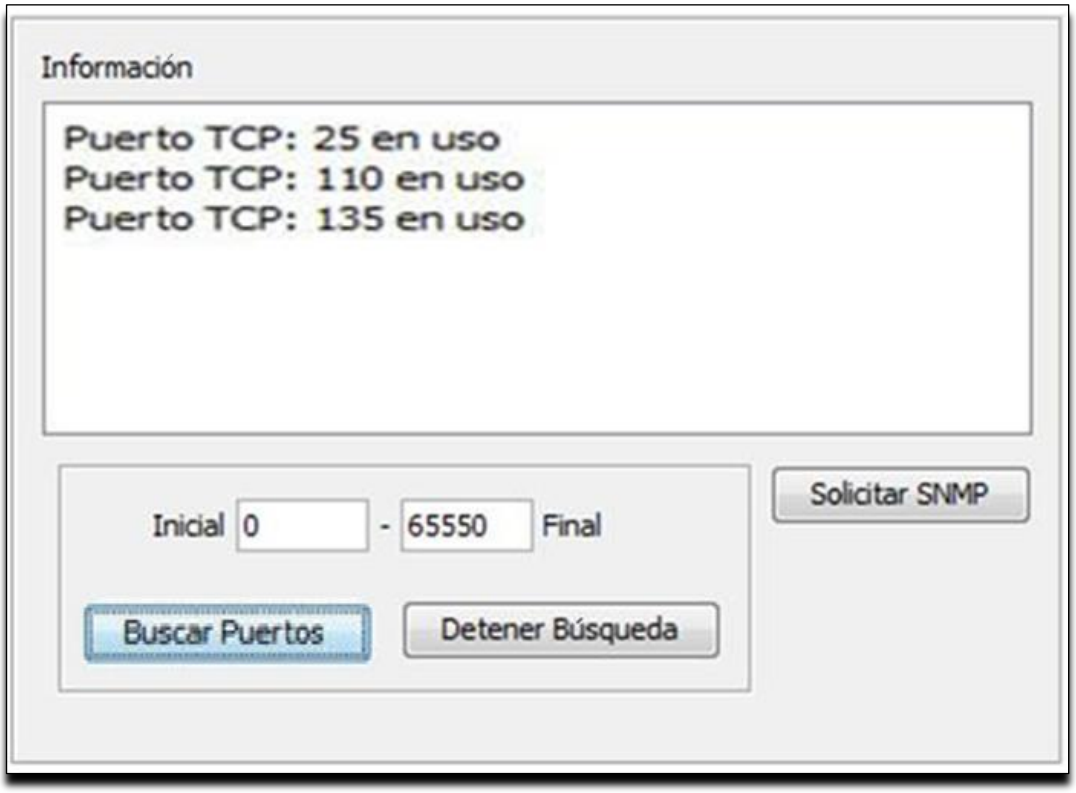


Figura H.4 Escaneo de puertos TCP y UDP de un host específico

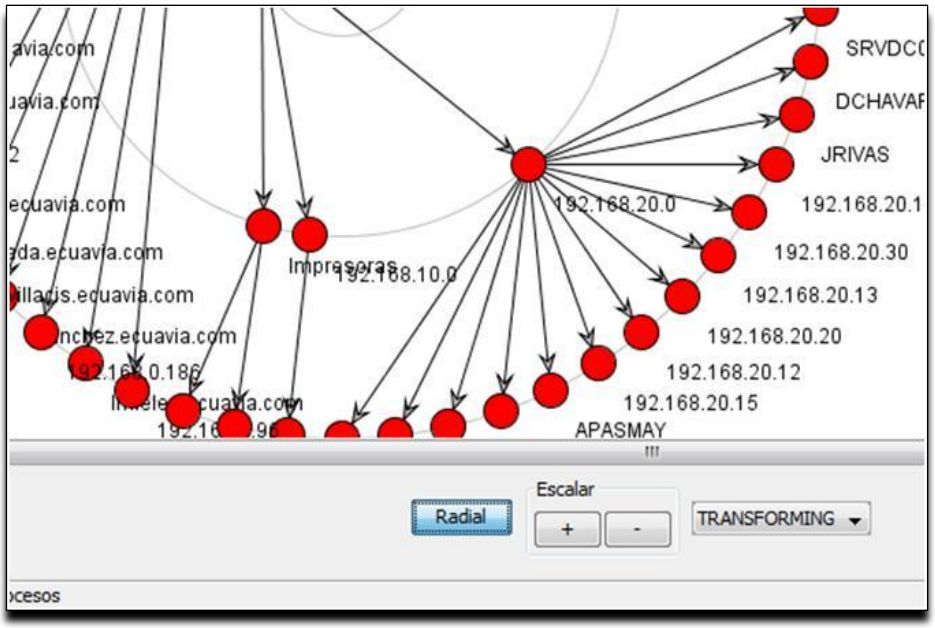


Figura H.5 Topología de la red mediante gráfico

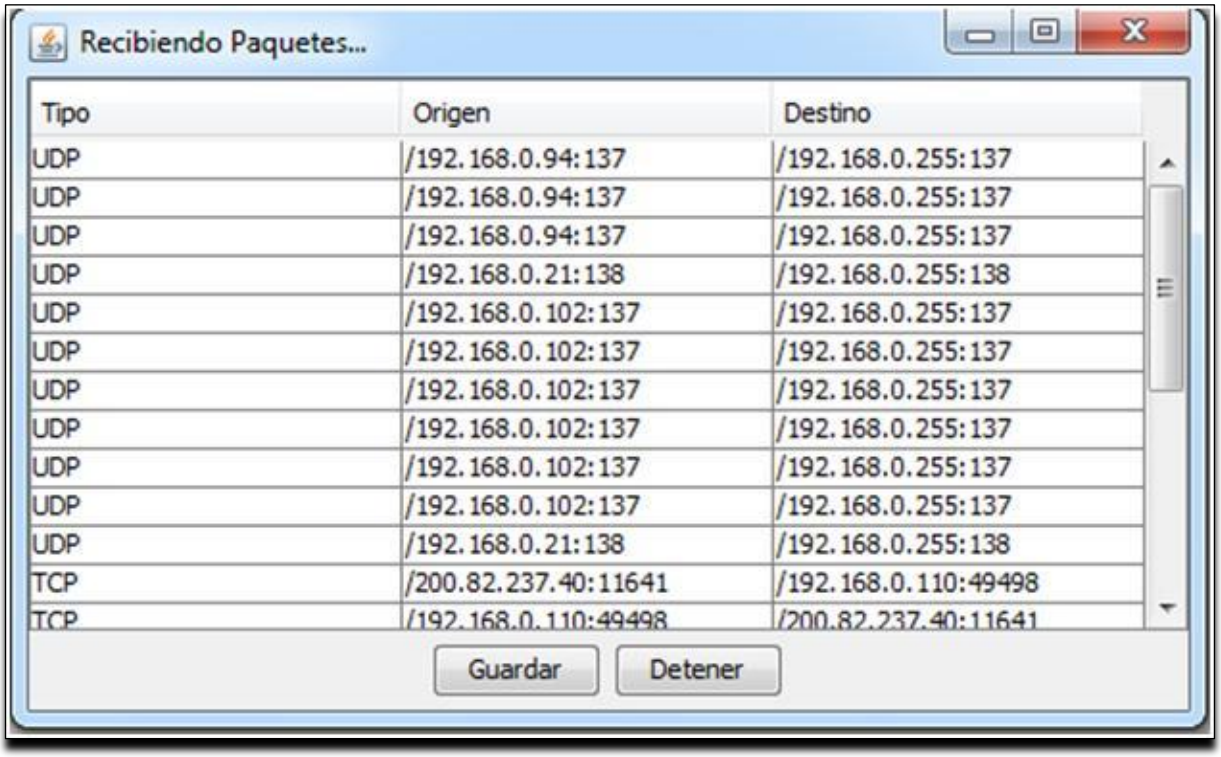


Figura H.6 Escaneo de tráfico en la Interfaz Local

BIBLIOGRAFÍA

[1] "STALLINGS, William"; Comunicaciones y Redes de Computadores. Sexta Edición. Prentice Hall.2000.

[2] Herramientas Web para la enseñanza de protocolos de comunicación - EL PROTOCOLO IP < http://neo.lcc.uma.es/evirtual/cdd/tutorial/red/ip.html>, 16 de Marzo del 2011

[3] El modelo OSI y los protocolos de red
< http://blyx.com/public/docs/pila_OSI.pdf> ,16 de Marzo del 2011

[4] "CISCO" - Network Management Fundamentals -
<http://www.scribd.com/doc/13089205/Network-Management Fundamentals-Alexander-Clemm>, 23 de Marzo del 2011

[5] "Aiko Pras" - Arquitectura de Administración de Redes –
< http://doc.utwente.nl/17897/1/t0000011.pdf, > 29 de Marzo del 2011

[6] Modelos de gestión de red < http://tvdi.det.uvigo.es/~mramos/gprsi/gprsi3.pdf >, 2 de Febrero del 2011

[7] Administración de Redes – Capitulo 4 < https://docs.google.com/document/d/1c1tW1GfBC7chP-SpgtgtowjoHb3XVknIOf3YVfSgJmg/edit?hl=en&pli=1#>, 21 de Febrero del 2011

[8] "R. y Kevin J. Schmidt" - Essentials SNMP – Douglas <http://docstore.mik.ua/orelly/networking_2ndEd/snmp/ >, 4 de Febrero del 2011

[9] "M. Rose" - RFC 1155 – Estructura e identificación de información para la administración de TCP/IP – <http://www.faqs.orgl >, 3 de Febrero del 2011

[10] "Mark A. Miller" - Gestión de Internetworks con SNMP – <http://ebookee.org/Managing-Internetworks-With-Snmp_362608.html >,6 de Abril del 2011

[11] "K. McCloghrie" - RFC 2578 - Estructura de administración información versión 2 – <http://www.faqs.orgl >, 4 de Abril del 2011

[15] "Martin Fowler" - La nueva metodología – <http://www.martinfowler.com/articles/newMethodology.html >, 29 de Marzo del 2011

[16] "Risk Technology"" Metodología de desarrollo de sistemas – < http://www.risktechnology.net/metodologia-rup.aspx>, 29 de Marzo del 2011

[17] PCharles – Network Packet Capture Facility for Java - < http://www.sf.net >, 3 de Febrero del 2011

[18] “Fantom Drives” – User´s Guide External Hard Drive – 27 de Marzo del 2011