UNIVERSIDAD TÉCNICA DE COTOPAXI

UNIDAD ACADÉMICA DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

INGENIERÍA EN INFORMÁTICA Y SISTEMAS COMPUTACIONALES

TESIS DE GRADO

TEMA:

"DISEÑO DE UNA HERRAMIENTA PARA EL MONITOREO DE SERVICIOS Y DE LA RED DEL BLOQUE B DE LA UNIVERSIDAD TÉCNICA DE COTOPAXI CAMPUS SAN FELIPE"

TESIS PRESENTADA PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN INFORMÁTICA Y SISTEMAS COMPUTACIONALES.

POSTULANTES:

QUINTANILLA BERMUDEZ GABRIELA DEL CARMEN

REINOSO REINOSO DANIELA LUCIA

DIRECTOR:

ING. JUAN CARLOS RODRIGUEZ.

LATACUNGA – ECUADOR

FEBRERO 2011

AVAL DEL TRIBUNAL DE DEFENSA

DE TESIS

En nuestra calidad de Miembros del Tribunal de la Defensa de Tesis titulada **"DISEÑO DE UNA HERRAMIENTA PARA EL MONITOREO DE SERVICIOS Y DE LA RED DEL BLOQUE B DE LA UNIVERSIDAD TÉCNICA DE COTOPAXI CAMPUS SAN FELIPE"**, de autoría de las postulantes, Gabriela del Carmen Quintanilla Bermúdez y Daniela Lucía Reinoso Reinoso; Ingenieras la Carrera de Ingeniería en Informática y Sistemas Computacionales CIYA – UTC. Certificamos que se han realizado las correcciones sugeridas al mismo; por lo que se encuentran aptas para empastar la Tesis.

Es todo cuanto podemos certificar en honor a la verdad.

Atentamente,

Presidente

Miembro Interno

Miembro Externo

Opositor

AUTORÍA DE TESIS

Las abajo firmantes, en calidad de egresados de la Universidad Técnica de Cotopaxi, Unidad Académica de Ciencias de la Ingeniería y Aplicadas, Especialización Ingeniería en Informática y Sistemas Computacionales, declaramos que los contenidos de esta Tesis de grado, requisito previo a la obtención del Título de Ingeniero en Informática y Sistemas Computacionales, son absolutamente originales, auténticos, personales y de exclusiva responsabilidad legal y académica de las autoras.

Latacunga, febrero del 2011.

Quintanilla B. Gabriela
C.I. 050251713-9

Reinoso R. Daniela
C.I. 050296145-1

CERTIFICACIÓN

Debo certificar que cumpliendo con lo estipulado en el capítulo V, artículo 12, literal f del reglamento de la Universidad Técnica de Cotopaxi, que el tema de tesis titulado "**DISEÑO DE UNA HERRAMIENTA PARA EL MONITOREO DE LOS SERVICIO Y DE LA RED DEL BLOQUE B DE LA UNIVERSIDAD TÉCNICA DE COTOPAXI, CAMPUS SAN FELIPE**", propuesto por las egresadas Gabriela del Carmen Quintanilla Bermúdez y Daniela Lucía Reinoso Reinoso han concluido el presente trabajo de investigación de acuerdo a los planteamientos formulados en el plan de tesis, siendo ejecutado y revisado meticulosamente.

Atentamente

Ing. Juan Carlos Rodríguez
Director de Tesis

Latacunga, febrero del 2011

CERTIFICACIÓN

Yo, Lcda. Mónica Chuchico, con cédula de identidad 050315269-6, en mi calidad de docente del Idioma Inglés del Centro Infantil Gotita de Gente y propietaria de Green Apple Courses certifico haber revisado el resumen de la tesis de las señoritas; Gabriela del Carmen Quintanilla Bermúdez y Daniela Lucía Reinoso Reinoso, egresadas de la Carrera de Ingeniería en Informática i Sistemas Computacionales de la Universidad Técnica de Cotopaxi, pudiendo atestiguar que el contenido está correctamente estructurado y libre de errores.

Es todo cuento puedo afirmar en honor a la verdad, las interesadas pueden hacer uso del presente documento como lo estipulen conveniente

Lo certifico

Lcda. Mónica Chuchico

UNIVERSIDAD TÉCNICA DE COTOPAXI

DIRECCIÓN DE SERVCIOS INFORMÁTICOS

Latacunga – Ecuador

CERTIFICACIÓN

Mediante el presente me permito en certificar que las Señoritas Gabriela Quintanilla y Daniela Reinoso estudiantes de la Universidad Técnica de Cotopaxi, realizaron las pruebas en las salas de Cómputo del Bloque Académico B para el monitoreo de la red.

Es todo cuanto puedo certificar de conformidad con los archivos que reposan en esta dependencia.

Latacunga febrero 07,2011

Atentamente

"POR LA VINCULACIÓN DE LA UNIVERSIDAD CON EL PUEBLO"

Ing. Adrián Mena Rojas

DIRECTOR DE SERVICIOS INFORMÁTICOS (E)

DEDICATORIA

Todo el tiempo y el esfuerzo empleados en este trabajo de investigación lo dedico a las personas que nunca dudaron que culminaría con éxito esta etapa de mi vida estudiantil.

A Dios que por su infinita gracia y bondad me dio la vida.

A mis padres, Mario y Martha, por su apoyo incondicional y confianza infinita en mi.

A mi querida hermana, Tania, quien depositó toda su fe y esperanza en mi capacidad y por su cariño y compañía.

A mi sobrinito, Javi, que con su sola existencia me da fuerza para seguir sin dudar.

A mi tío, José, que cada día me demuestra que el que persevera alcanza.

A ellos ofrezco todo mi trabajo por su ilimitado amor.

Gabriela

DEDICATORIA

Dedico este material investigativo, a Dios por ser mi creador y el que actúa en la trayectoria de nuestras vidas.

A mis padres, por brindarme la vida y en el transcurso de ella cobijarme con su amor, comprensión, e infinito apoyo.

A mi hermana por aportar en mi vida una idea, un consejo.

Esperamos al cumplir esta meta, recompensar en alguna forma todo el sacrificio y apoyo que nos han brindado.

Daniela

AGRADECIMIENTO

El mundo está lleno de personas importantes pero algunas de ellas se destacan porque son tu motivo, tu motor, por ello a esas personas indispensables en mi vida quiero darles el reconocimiento que merecen por su aporte en mi vida y a la culminación exitosa de mi tesis.

A Dios, gracias, por invisible compañía en cada paso que di durante la elaboración de mi tesis.

A mis padres, gracias, porque con su ejemplo de perseverancia y amor ante los desafíos del mundo siempre nos apoyaron a mi hermana y a mi.

A mi hermana, gracias, porque con su ejemplo de lucha y valor al enfrentar la vida me inspiró a seguir sin rendirme y porque día a día me acompaña su amor.

A mi sobrinito, gracias, porque su inocencia y alegría contagian mi existencia y me permiten disfrutar de ella.

A mi tío, gracias, por todos el apoyo brindado durante toda mi vida.

Gabriela

AGRADECIMIENTO

En la historia está comprobado que no ha existido ni existirá ningún ser humano que pueda sobrevivir individualmente; mucho menos sobresalir y desarrollarse íntegramente, por lo que agradezco el éxito de este proyecto a todos aquellos que aportaron de una u otra manera con la finalización del mismo.

A la Universidad Técnica de Cotopaxi por ser la cuna de mi desarrollo profesional.

A mis padres quienes con su ejemplo de sacrificio y amor, me infundieron la responsabilidad, ética y moral que tutelan mi transitar por la vida.

A mi hermanas, que siempre han tenido para mi una palabra de aliento y que con su apoyo me ha dado la fortaleza para terminar este trabajo de investigación.

Daniela

RESUMEN

LA HERRAMIENTA DISEÑADA PARA EL MONITOREO DE SERVICIOS Y DE LA RED DEL BLOQUE B DE LA UNIVERSIDAD TÉCNICA DE COTOPAXI, tema de nuestra tesis, recolecta información sobre los aspectos claves de la red de esta Institución Educativa para así detectar y resolver posibles conflictos en la misma registrando sistemáticamente las variables y procesos claves que ocurren en un periodo de tiempo y espacio.

Para cumplir con esta tarea se consideraron los siguientes aspectos: utilización de enlaces, servicios de red, porcentaje de transmisión y recepción de información, disponibilidad de elementos de la red, entre otros; una vez recogida la información mediante el monitoreo se interpreta la misma para poder determinar el comportamiento de la red y tomar las decisiones específicas que ayuden a mejorar su desempeño. La administración de la herramienta de monitoreo estará a cargo del Departamento de Desarrollo de Software de la Universidad Técnica de Cotopaxi.

ABSTRACT

The tool designed for **MONITORING THE SERVICES AND NETWORKING OF BLOCK B OF THE TECHNICAL UNIVERSITY OF COTOPAXI**, our thesis topic, collects the information about relevant aspects in the network of this Institution to detect and to resolve posible conflicts in it, recording systematically the variables and process that occur in a period of time and space.

The prform this task we considered the following aspects: use of links, network services, percentage of transmission and reception of information, availability of network elements, among others, once the information is collected it is interpreted by monitoring to determine the behavior of the network and take specific decisions which improve its performance. The administration of the Tool of Monitoring will be in charge of Software Developmente Department of Technical University of Cotopaxi

INDICE

CAPÌTULO I

FUNDAMENTACIÓN TEÓRICA DEL DISEÑO DE UNA HERRAMIENTA PARA EL MONITOREO DE SERVICIOS Y DE LA RED

CAPÌTULO II

DESCRIPCION, ANALISIS E INTERPRETACION DE RESULTADOS

CAPÍTULO III

"DISEÑO DE UNA HERRAMIENTA PARA EL MONITOREO DE SERVICIOS Y DE LA RED DEL BLOQUE B DE LA UNIVERSIDAD TECNICA DE COTOPAXI CAMPUS SAN FELIPE"

CAPITULO I

"FUNDAMENTACIÓN TEÓRICA DEL DISEÑO DE UNA HERRAMIENTA PARA EL MONITOREO DE SERVICIOS Y DE LA RED"

1. Generalidades

Llevar el control de una organización requiere poner atención en diferentes horizontes, en la factibilidad, en la eficiencia, en la rapidez, etc. y también efectuar mediciones sobre algunos parámetros como la calidad de la señal, el alcance, el nivel del servicio, etc. todo esto es posible gracias al monitoreo.

Los ataques a la red y las fallas del sistema provenientes del interior o exterior de las organizaciones o instituciones son comunes por lo tanto resulta de trascendental importancia el monitoreo de las redes para no disminuir el desempeño de la red.

1.1 Red

Por red se entiende a un sistema de ordenadores interconectados, a través de los cuales se podrá compartir recursos e intercambiar información entre las diferentes máquinas.

En cuanto a los elementos que la conforman, la red está integrada por un nodo o terminal y un medio de transmisión.

El nodo o terminal es el que inicia o termina la comunicación, como la computadora, aunque también hay otros dispositivos, como por ejemplo una impresora. Mientras que los medios de transmisión son los cables o las ondas electromagnéticas (tecnología inalámbrica, enlaces vía satélite, etc.).

1.1.1 Tipos de Redes

Entre los principales tipos de redes tenemos:

Redes de Área Local (LAN).- Una LAN (Local Area Network) es un sistema de interconexión de equipos de equipos informáticos basado en líneas de alta velocidad (decenas o cientos de megabits por segundo) y que suele abarcar, como mucho, un edificio.

Las principales tecnologías usadas en una LAN son: Ethernet, Token ring, ARCNET y FDDI.

Redes de Área Metropolitana (MAN).- Una MAN (Metropolitan Area Network) es un sistema de interconexión de equipos informáticos distribuidos en una zona que abarca diversos edificios, por medios pertenecientes a la misma organización propietaria de los equipos. Este tipo de redes se utiliza normalmente para interconectar redes de área local.

Redes de Área Extensa (WAN).- Una WAN (Wide Area Network) es un sistema de interconexión de equipos informáticos geográficamente dispersos, que pueden estar incluso en continentes distintos. El sistema de conexión para estas redes normalmente involucra a redes públicas de transmisión de datos.

Redes de Área Local Inalámbricas (WLAN).- WLAN (Wireless Local Área Network), que se basa en la transmisión de datos mediante ondas de radio, microondas, satélites o infrarrojos.

La velocidad de transmisión de las redes WLAN, surgidas experimentalmente a principios de los noventa, va de los 10 a los 100 Mbps, y son el complemento ideal para las redes fijas, por tener capacidad de enlazarse con las redes cableadas.

1.1.2 Topologías de las redes

Las topologías de red se clasifican en físicas y en lógicas:

Topologías físicas

- Una **topología de bus** usa un solo cable backbone que debe terminarse en ambos extremos, todos los hosts se conectan directamente a este backbone.

- La **topología de anillo** conecta un host con el siguiente y al último host con el primero, esto crea un anillo físico de cable.

- La **topología en estrella** conecta todos los cables con un punto central de concentración.

- Una **topología en estrella extendida** conecta estrellas individuales entre sí mediante la conexión de HUBs o switches, esta topología puede extender el alcance y la cobertura de la red.

- Una **topología jerárquica** es similar a una estrella extendida, pero en lugar de conectar los HUBs o switches entre sí, el sistema se conecta con un computador que controla el tráfico de la topología.

- La **topología de malla** se implementa para proporcionar la mayor protección posible para evitar una interrupción del servicio, el uso de una topología de malla en los sistemas de control en red de una planta nuclear sería un ejemplo excelente.

- La **topología de árbol** combina características de la topología de estrella con la BUS, consiste en un conjunto de subredes estrella conectadas a un BUS, esta topología facilita el crecimiento de la red.

Topologías lógicas

La topología lógica de una red es la forma en que los hosts se comunican a través del medio, los dos tipos más comunes de topologías lógicas son broadcast y transmisión de tokens.

- **La topología broadcast** simplemente significa que cada host envía sus datos hacia todos los demás hosts del medio de red, no existe una orden que las estaciones deban seguir para utilizar la red, es por orden de llegada, es como funciona Ethernet.

- **La topología de transmisión de tokens** controla el acceso a la red mediante la transmisión de un token electrónico a cada host de forma secuencial, cuando un host recibe el token, ese host puede enviar datos a través de la red, si el host no tiene ningún dato para enviar transmite el token al siguiente host y el proceso se vuelve a repetir. Dos ejemplos de redes que utilizan la transmisión de tokens son Token Ring y la Interfaz de Datos Distribuida por Fibra (FDDI). Arcnet es una variación de Token Ring y FDDI. Arcnet es la transmisión de tokens en una topología de bus.

1.2 Servicios

Se definen como servicios a los bienes que no pueden verse, probarse, sentirse, oírse ni olerse y sin embargo cubren necesidades.

Las redes comenzaron a ser utilizadas de muchas formas y con muchos propósitos, cada una de estas formas de uso es lo que se conoce como un servicio, una forma estandarizada de utilización, lo que a su vez requiere el uso de protocolos (estándares) universalmente aceptados tanto por los "clientes" como por los "proveedores" del servicio.

1.2.1 Servicios de red

Una red básicamente provee servicios tales como:

Internet.- La llamada "autopista de la información" es, realmente, un conjunto de miles de redes informáticas unidas entre sí.

Al hablar de Internet se hace referencia a una red que no pertenece a nadie, sino que esta conformada por la información que le brindan los millones de usuarios que se conectan a ella.

Compartir recursos e información.- Los equipos que están conectados en red les dan la posibilidad a los usuarios de la misma de compartir recursos tales como impresoras e información entre componentes de la red facilitando el trabajo y ahorrando tiempo.

La finalidad de una red es que los usuarios de los sistemas informáticos de una organización puedan hacer un mejor uso de los mismos optimizando de este modo el rendimiento global de la organización, así las organizaciones obtienen una serie de ventajas del uso de las redes en sus entornos de trabajo, como pueden ser:

- Mayor facilidad de comunicación.
- Mejora de la competitividad.
- Mejora de la dinámica de grupo.
- Reducción del presupuesto para proceso de datos.
- Reducción de los costos de proceso por usuario.
- Mejoras en la administración de los programas.
- Mejoras en la integridad de los datos.
- Mejora en los tiempos de respuesta.
- Flexibilidad en el proceso de datos.
- Mayor variedad de programas.
- Mayor facilidad de uso.

1.3 Monitorear

Es el proceso de recoger información sobre varios aspectos de una red, es seguir sistemáticamente las variables y procesos claves en un periodo de tiempo y espacio y ver cómo cambian por el resultado de las actividades en el medio, hacer esto requiere analizar aspectos tales como:

- La implementación de un plan de monitoreo.
- La evaluación de los resultados del plan de monitoreo.
- La elaboración de informes y diseminación de los hallazgos de estas actividades.
- La disponibilidad de elementos de la red como:
  - Routers, Firewalls, servidores.
  - Tiempo de respuesta.
  - Utilización de los enlaces.
  - Otros: carga cpu, espacio de disco, uso de la memoria, etc.

1.3.1 Plan de monitoreo

Es la estrategia empleada para el proceso de recolección de la información rutinariamente sobre aspectos determinados y usarla en la administración y toma de decisiones de la red; por lo que se puede afirmar que un plan de monitoreo es el como se realizará la monitorización de la red, lo que es vital ya que provee a los miembros de la red, a los administradores de la misma y a otros interesados información que es esencial en el diseño, implementación, administración y evaluación de las actividades de monitoreo.

1.3.2 Monitoreo de redes

La palabra monitoreo no tiene una definición exacta, pero en el contexto computacional a adquirido una gran importancia, de manera más especifica en el área de redes, es una función que busca conocer cómo se están realizando las tareas especificadas en el plan de monitoreo. En el área tecnológica el concepto de monitoreo de redes debe de ser más práctico, por ejemplo, todas las acciones que se realizan son activas y se necesita tener constante control del equipo involucrado, es decir, supervisar funciones específicas que se dan en la red, osea los servicios que oferte la misma.

El monitoreo se encuentra muy ligado con el concepto de inteligencia competitiva la cual se define como conocimiento generado a partir del análisis resultante de la integración de información sobre el entorno de los datos disponibles lícitamente, en base a este conocimiento generado el administrador de la red puede realizar acciones concretas en el caso de que una red o un servicio que presta la misma presente algún tipo de problema.

1.3.3 Administración del rendimiento del monitoreo

Tiene como objetivo recoger y analizar información sobre el funcionamiento de la red y sus servicios para determinar 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 de acuerdo al comportamiento encontrado.

La administración del rendimiento se divide en 2 etapas:

- Monitoreo
- Análisis

Monitoreo

El monitoreo consiste en observar y recolectar la información referente al comportamiento de la red en aspectos como los siguientes:

a) **Utilización de enlaces.-** Se refiere básicamente al ancho de banda utilizada por cada uno de los enlaces de área local (Ethernet, Fastethernet, GigabitEthernet, etc.), ya sea por elemento o de la red en su conjunto.

b) **Caracterización de tráfico.-** Es la tarea de detectar los diferentes tipos de tráfico que circulan por la red, con el fin de obtener datos sobre los servicios de red, como http, ftp, además esto también permite establecer un patrón en cuanto al uso de la red.

c) **Porcentaje de transmisión y recepción de información.-** Encontrar los elementos de la red que más solicitudes hacen y atienden, como servidores, estaciones de trabajo, dispositivos de interconexión, puertos y servicios.

Análisis.

Una vez recolectada la información mediante la actividad de monitoreo, es necesario interpretarla para determinar el comportamiento de la red y tomar decisiones adecuadas que ayuden a mejorar su desempeño.

a) **Utilización elevada.-** Si se detecta que la utilización de un enlace es muy alta, se puede tomar la decisión de incrementar su ancho de banda o de agregar otro enlace para balancear las cargas de tráfico. También el incremento en la utilización puede ser el resultado de la saturación por tráfico generado maliciosamente, en este caso se debe contar con un plan de respuesta a incidentes de seguridad.

b) **Tráfico inusual.-** El haber encontrado mediante el monitoreo el patrón de aplicaciones que circulan por la red ayudará a poder detectar tráfico inusual o fuera del patrón, aportando elementos importantes en la resolución de problemas que afecten el rendimiento de la red.

c) **Elementos principales de la red.-** Un aspecto importante de conocer cuáles son los elementos que más reciben y transmiten, es el hecho de poder identificar los elementos a los cuales establecer un monitoreo más constante, debido a que seguramente son de importancia. Además, si se detecta un elemento que generalmente no se encuentra dentro del patrón de los equipos con más actividad, puede ayudar a la detección de posibles ataques a la seguridad de dicho equipo.

d) **Calidad de servicio.-** Otro aspecto, es la Calidad de servicio o QoS, es decir, garantizar, mediante ciertos mecanismos, las condiciones necesarias, como ancho de banda, retardo, aplicaciones que requieren de un trato especial como lo son la voz sobre IP.

e) **Control de tráfico.-** El tráfico puede ser reenviado o ruteado por otro lado cuando se detecte saturación por un enlace o al detectar que se encuentra fuera de servicio, esto se puede hacer de manera automática si se cuenta con enlaces redundantes.

1.4 Protocolos

1.4.1 IP: Protocolo de Internet

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.

1.4.2 TCP: Protocolo de Control de Transmisión

Protocolo de la capa de Transporte, que permite dividir y ordenar la información a transportar en paquetes de menor tamaño para su transporte y recepción.

El principal propósito de TCP es proporcionar una conexión lógica fiable entre parejas procesos, no asume la fiabilidad de los protocolos de niveles inferiores (como IP) por lo que debe ocuparse de garantizarla.

TCP se puede caracterizar por los siguientes servicios que suministra a las aplicaciones que lo usan:

- **Transferencia de datos a través de un canal.-** Desde el punto de vista de la aplicación, TCP transfiere un flujo continuo de bytes a través de Internet. La aplicación no ha de preocuparse de trocear los datos en bloques o en datagramas. TCP se encarga de esto al agrupar los bytes en segmentos TCP, que se pasan a IP para ser retransmitidos al destino.

- **Control de flujo.-**El TCP receptor, al enviar un ACK, cuyo significado es "ha llegado el paquete y además ha llegado correctamente" al emisor, indica también el número de bytes que puede recibir aún, sin que se produzca sobrecarga y desbordamiento de sus buffers internos, este mecanismo se conoce también como mecanismo de ventanas.

- **Multiplexación.-**Se consigue usando puertos.

- **Conexiones lógicas.-**La fiabilidad y el control de flujo descritos más arriba requieren que TCP inicialice y mantenga cierta información de estado para cada canal, la combinación de este estado, incluyendo zócalos, números de secuencia y tamaños de ventanas, se denomina conexión lógica. Cada conexión se identifica unívocamente por el par de zócalos del emisor y el receptor.

- **Full Duplex.-**TCP garantiza la concurrencia de los flujos de datos en ambos sentidos de la conexión.

1.4.3 UDP: Protocolo de Datagrama de Usuario

Este protocolo proporciona una comunicación muy sencilla entre las aplicaciones de dos ordenadores, al igual que el protocolo IP, UDP es:

- No orientado a conexión.- No se establece una conexión previa con el otro extremo para transmitir un mensaje UDP, los mensajes se envían sin más y éstos pueden duplicarse o llegar desordenados al destino.

- No fiable.- Los mensajes UDP se pueden perder o llegar dañados.

UDP utiliza el protocolo IP para transportar sus mensajes, no añade ninguna mejora en la calidad de la transferencia; aunque sí incorpora los puertos origen y destino en su formato de mensaje.

1.5 Herramientas para monitoreo y administración de red

1.5.1 Nagios.- Es un sistema de monitorización de redes y servidores que chequea de forma periódica los nodos y los servicios que se especifiquen a través de la red, alertando cuando se superan los indicadores definidos y cuando se vuelve de nuevo a una situación estable. Su gran versatilidad permite a Nagios monitorizar prácticamente cualquier cosa que está en la red. Originalmente fue diseñado para ejecutarse bajo sistemas Linux, aunque en la actualidad es posible su instalación en otros sistemas.

1.5.2 Ntop.- Network top (Ntop) es un software que nos permite monitorizar el tráfico de la red, con una gran potencia debido al gran número de protocolos que soporta. La utilidad Ntop nos permite realizar un análisis de todo el tráfico que pasa a través de nuestro interfaz, Ntop proporciona determinadas características como por ejemplo: modelo de los equipos y

S.O., trafico acumulado entre dos equipos, equipos que generan mayor cantidad de tráfico, entre otros. Es fácil de instalar y de usar, ya que es vía web y se le pueden incorporar otras herramientas como el rrdtools para la generación de gráficas que facilitan la obtención de resultados.

1.5.3 Nessus.- Auditor de Seguridad Remoto. El cliente "The Nessus Security Scanner" es una herramienta de auditoría de seguridad que hace posible evaluar módulos de seguridad intentando encontrar puntos vulnerables que deberían ser reparados.

Está compuesto por dos partes: un servidor y un cliente. El servidor/daemon, "nessus" se encarga de los ataques, mientras que el cliente, "nessus", se ocupa del usuario por medio de una linda interfaz para X11/GTK+; este paquete contiene el cliente para GTK+1.2, que además existe en otras formas y para otras plataformas.

1.5.4 Netcat.- Es una navaja multiuso para TCP/IP. Una utilidad simple para Unix que lee y escribe datos a través de conexiones de red usando los protocolos TCP o UDP. Está diseñada para ser una utilidad del tipo "back-end" confiable que pueda ser usada directamente o fácilmente manejada por otros programas y scripts. Al mismo tiempo es una herramienta rica en características útil para depurar (debug) y explorar, ya que puede crear casi cualquier tipo de conexión que puedas necesitar y tiene muchas características incluidas.

1.5.5 Tcpdump.- Es una poderosa herramienta para el monitoreo y la adquisición de datos en redes; este programa te permite volcar (a un archivo, la pantalla, etc.) el tráfico que presenta una red. Puede ser usado para imprimir los encabezados de los paquetes en una interfaz de red ("network interface") que concuerden con una cierta expresión. Se puede usar esta herramienta para seguir problemas en la red, para detectar "ping attacks" o para monitorear las actividades de una red.

1.5.6 Snort.- Es un Sniffer/logger de paquetes flexible que detecta ataques. Snort está basado en la biblioteca `libpcap' y puede ser usado como un "sistema de detección de intrusiones" (IDS) de poco peso. Posee un registro basado en reglas y puede buscar/identificar contenido además de poder ser usado para detectar una gran variedad de otros ataques e investigaciones (probes), como buffer overflows, barridos de puertos indetectables (stealth port scans), ataques CGI, pruebas de SMB (SMB probes), y mucho más. Otra característica importante de Snort es la capacidad de alertar en tiempo real, siendo estas alertas enviadas a syslog, un archivo de alerta separado o incluso a una computadora con Windows a través de Samba.

1.5.7 Saint.- SAINT (Security Administrator's Integrated Network Tool / Herramienta De Red Integrada del Administrador de Seguridad) es una herramienta de evaluación de seguridad basada en SATAN. Incluye escaneos a través de un firewall, chequeos de seguridad actualizados de los boletines de CERT Y CIAC, 4 niveles de severidad (rojo, amarillo, marrón y verde) y una interfaz HTML rica en características.

1.5.8 Ethereal.- Ethereal es un analizador de tráfico de redes, o "sniffer" para Unix y Sistemas operativos del tipo Unix, usa GTK+, una biblioteca de interfaz gráfica para el usuario (GUI), y libcap, una biblioteca que facilita la captura y el filtrado de paquetes.

1.5.9 Sara.- El Asistente de Investigación para el Auditor de Seguridad (Security Auditor's Research Assistant) es una herramienta de análisis de seguridad de tercera generación que está basada en el modelo de SATAN y distribuida bajo una licencia del estilo de la GNU GPL. Promueve un ambiente colaborativo y es actualizada periódicamente para tener en cuenta las últimas amenazas.

1.5.10 Satan.- Herramienta de Auditoría de Seguridad para Analizar Redes (Security Auditing Tool for Analysing Networks). Ésta es una poderosa herramienta para analizar redes en búsqueda de vulnerabilidades, creada para administradores de sistemas que no pueden estar constantemente.

1.6 Sockets

Socket designa un concepto abstracto por el cual dos programas (posiblemente situados en computadoras distintas) pueden intercambiar cualquier flujo de datos, generalmente de manera fiable y ordenada.Un socket queda definido por una dirección IP, un protocolo de transporte y un número de puerto.

Para que dos programas puedan comunicarse entre sí es necesario que se cumplan ciertos requisitos:

- Que un programa sea capaz de localizar al otro.
- Que ambos programas sean capaces de intercambiarse cualquier secuencia de octetos, es decir, datos relevantes a su finalidad.

Para ello son necesarios los tres recursos que originan el concepto de socket:

- Un protocolo de comunicaciones, que permite el intercambio de octetos.
- Una dirección del Protocolo de Red (Dirección IP, si se utiliza el Protocolo TCP/IP), que identifica una computadora.
- Un número de puerto, que identifica a un programa dentro de una computadora.

Los sockets permiten implementar una arquitectura cliente-servidor, la comunicación ha de ser iniciada por uno de los programas que se denomina

programa cliente, el segundo programa espera a que otro inicie la comunicación, por este motivo se denomina programa servidor.

Un socket es un fichero existente en la máquina cliente y en la máquina servidora, que sirve en última instancia para que el programa servidor y el cliente lean y escriban la información. Esta información será la transmitida por las diferentes capas de red.

1.6.1 Propiedades inherentes a los Sockets

Las propiedades de un socket dependen de las características del protocolo en el que se implementan. El protocolo más utilizado es Transmission Control Protocol, aunque también es posible utilizar UDP o IPX. Cuando se implementan con el protocolo TCP, los sockets tienen las siguientes propiedades:

- Orientado a conexión.
- Garantía de transmisión de todos los octetos sin errores ni omisiones.
- Garantía que todo octeto llegará a su destino en el mismo orden en que se ha transmitido.

Estas propiedades son muy importantes para garantizar la corrección de los programas que tratan la información.

1.6.2 Mecanismo de comunicación vía Socket

El mecanismo de comunicación vía sockets tiene los siguientes pasos:

1º) El proceso servidor crea un socket con nombre y espera la conexión.
2º) El proceso cliente crea un socket sin nombre.

3º) El proceso cliente realiza una petición de conexión al socket servidor.

4º) El cliente realiza la conexión a través de su socket mientras el proceso servidor mantiene el socket servidor original con nombre.

Es muy común en este tipo de comunicación lanzar un proceso hijo, una vez realizada la conexión, que se ocupe del intercambio de información con el proceso cliente mientras el proceso padre servidor sigue aceptando conexiones. Para eliminar esta característica se cerrará el descriptor del socket servidor con nombre en cuanto realice una conexión con un proceso socket cliente.

CAPITULO II

"DESCRIPCION, ANALISIS E INTERPRETACION DE RESULTADOS"

2.1 Breve caracterización de la Universidad Técnica de Cotopaxi

En los primeros meses de 1989, en el salón de la Unión Nacional de Educadores de Cotopaxi (UNEC), los maestros, estudiantes, padres de familia y los sectores preocupados de nuestra Provincia conforman un Comité Provisional de Gestión, con el firme propósito de alcanzar su objetivo: "La creación de una Universidad para Cotopaxi", este comité estuvo conformado por las siguientes personalidades:

- Lic. César Tinajero, Diputado por Cotopaxi.
- Lic. Sócrates Hernández, Coordinador Técnico.
- Prof. José Huertas, Coordinador Administrativo.
- Lic. Edgar Cárdenas, Coordinador.
- Prof. Francisco Quishpe, Coordinador.
- Antonio Posso Salgado, Rector de la Universidad Técnica del Norte.

Se pretendía con esto formar una EXTENSION UNIVERSITARIA en nuestra ciudad bajo el aval de la Universidad Técnica del Norte, con este propósito, se conforma un Comité Pro-Extensión Universitaria para Cotopaxi designando al Sr. Edgar Cárdenas Vicepresidente de dicho comité; Antonio Posso acoge el clamor popular y manifestó en una visita que efectuó al Cantón Saquisilí la aceptación de

crear una Extensión Universitaria en nuestra Provincia que funcionaria desde Octubre de 1990.

Entre los requisitos legales para aprobar la creación de la Extensión Universitaria, era necesario que previamente se realicen estudios estadísticos de la realidad socioeconómica de la Provincia, y es así que bajo la dirección del Sr. Arq. Cesar Tinajero, Diputado por Cotopaxi, mediante telegrama del 18 de abril de 1990, indica que ha logrado incluir en el presupuesto de 1990 la cantidad de TREINTA MILLONES DE SUCRES para la extensión. Edgar Cárdenas, Vicepresidente del Comité Pro-Extensión Universitaria, comunica que "EL DOCUMENTO DEFINITIVO DE CREACION DE LA EXTENSION DE COTOPAXI SE ENCUENTRA CONCLUIDO", el mismo que fue enviado al CONUEP para su aprobación reglamentaria.

El 22 de febrero de 1991, las Autoridades Provinciales convocan a un PARO y en uno de los puntos principales de la plataforma de lucha se citaba la Creación de la Extensión Universitaria, en varias sesiones se ratificó el Comité Pro-Extensión Universitaria y se mantiene al Sr. Sócrates Hernández como su Presidente. Antonio Posso Rector de la Universidad Técnica del Norte, solicita al Sr. Teodoro Coello, presidente del CONUEP, "se inicie el trámite legal para la aprobación de la Extensión".

En el mes de junio de 1991, personeros de la Universidad Técnica del Norte, del CONUEP y del Comité Pro-Extensión, se dan a la tarea de inspeccionar los centros educativos que fueron ofrecidos en préstamo para que funcione la extensión. El 18 de julio de 1991, el CONUEP sesiona en la ciudad de Ibarra para tratar la Creación de la Extensión Universitaria, a esta reunión acudieron un gran número de ciudadanos cotopaxenses con el efecto de solicitar el apoyo y la aceptación para la creación de la Extensión Universitaria.

Sócrates Hernández, así como la valiosa participación de los señores militantes del Movimiento Popular Democrático (MPD), que trabajaron y ayudaron para conseguir la Extensión Universitaria, personeros como el Lic. César Tinajero, Sr.

Lic. Rodrigo Bustillos, Sr. Lic. Sociólogo Cristóbal Tinajero, Sr. Lic. Edgar Cárdenas, Sr. Lic. Francisco Quishpe, Sr. Lic. Rómulo Álvarez, entre otros trabajaron decididamente por la creación de la Extensión.

Inmediatamente se convoca a los aspirantes a alumnos, profesores y empleados, para la Extensión Universitaria. Dumy Naranjo de Lanas, Gobernadora de la Provincia de Cotopaxi anuncia según resolución No. 1619 la fijación de una partida en el Presupuesto del Estado, mediante la cual se asigna CIENTO VEINTE MILLONES DE SUCRES, para la Extensión Universitaria.

El Comité Pro- Extensión Universitaria convoca a una gran sesión, a la que acuden alumnos, profesores y la comunidad en general para nombrar las ternas para Directivos de la Extensión, el 14 de febrero de 1.992, se inaugura el Año Académico en la Extensión Universitaria, en el local del Colegio Técnico "Luís Fernando Ruiz", previa la firma de un Comodato para ocupar sus instalaciones.

La población universitaria estaba conformada por 398 alumnos, distribuidos así:

- Ingeniería Agroindustrial 151 alumnos 2 paralelos.
- Artesanía Artística 126 alumnos 2 paralelos.
- Contabilidad Pedagógica 121 alumnos 2 paralelos.

El Comité del Barrio Eloy Alfaro, motivado por el señor cura de la parroquia, le propone a la Ilustre Municipalidad de Latacunga donar a la Extensión Universitaria el edificio construido en el sector El Ejido. El mes de junio de 1.994 se toma la decisión por parte de las Autoridades de la Extensión de trasladarse a trabajar en las instalaciones del Colegio "Simón Rodríguez".

CREACION DE LA UNIVERSIDAD TECNICA DE COTOPAXI MEDIANTE ACUERDO DE LA CAMARA NACIONAL DE REPRESENTANTES

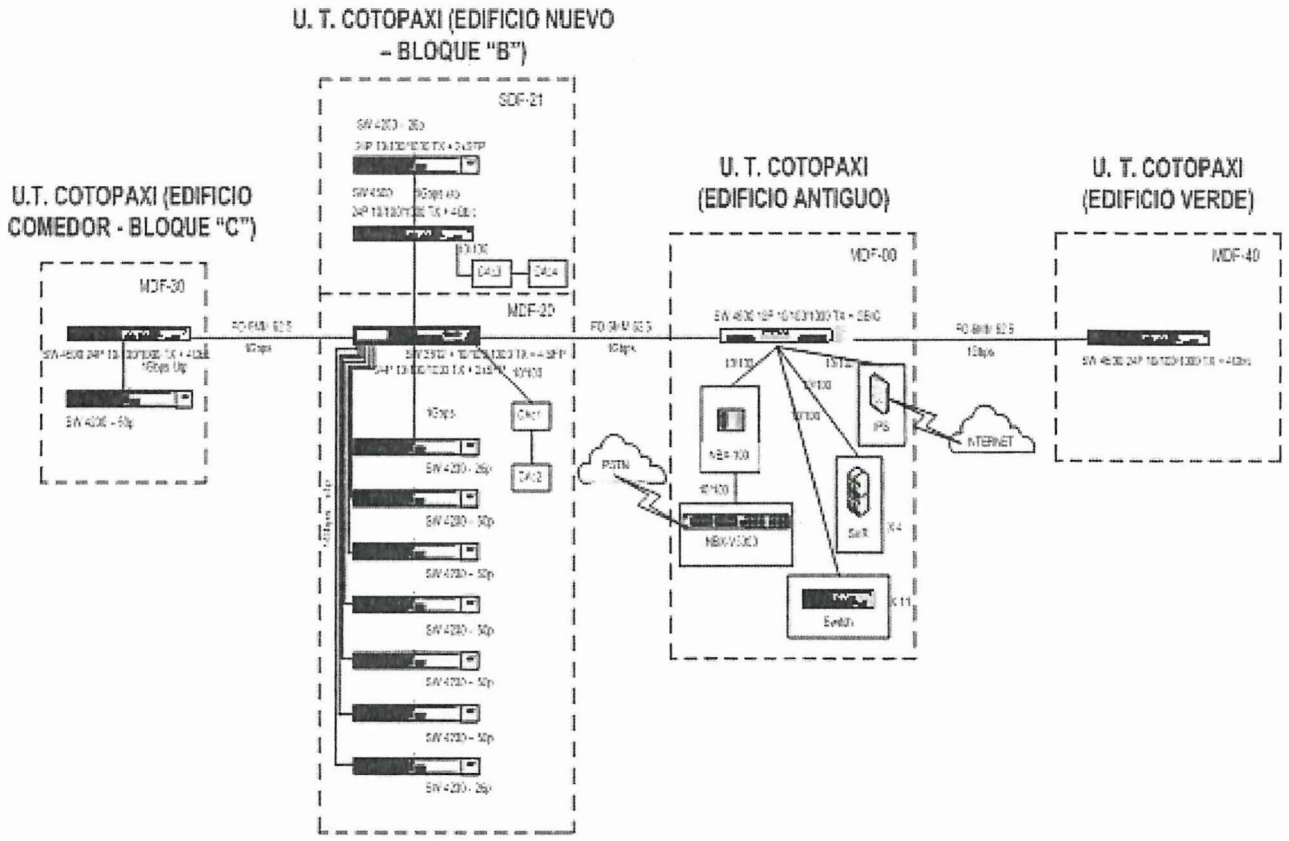
Siendo Director Titular de la Extensión el Sr. Fabián Fabara, Diputado por la Provincia se propone la elaboración de un Proyecto de Ley para la Creación de la Universidad Técnica de Cotopaxi, tomando como base la Extensión Universitaria, sus autoridades, su cuerpo docente y trabajadores, sus alumnos y patrimonio.

Con el apoyo del Lic. César Tinajero y por iniciativa de las Autoridades Universitarias, se realiza una campaña recogiendo firmas de respaldo de la comunidad cotopaxense, llegando autoridades y estudiantes universitarios al Ministerio de Gobierno para depositar en ese despacho, **TREINTA Y SEIS MIL FIRMAS** que apoyaban la creación de la Universidad Técnica de Cotopaxi.

Finalmente el Congreso Nacional, acogiéndose al veto ejecutivo, APRUEBA EN SEGUNDA INSTANCIA EL PROYECTO DE CREACION DE LA UNIVERSIDAD TECNICA DE COTOPAXI, Y QUE SE PUBLIQUE EN EL REGISTRO OFICIAL No. 618 DEL 24 DE ENERO DE 1995. Para dar fiel cumplimiento al mandato de creación el Sr. Lic. Rómulo Álvarez, Director de la Extensión pasa en forma legal a ser Rector Encargado de la Universidad, así también se nombran las restantes autoridades.

El mismo día, en magna sesión convocada por la Universidad se posesionan los SEÑORES RECTOR Y VICERRECTOR DE LA UNIVERSIDAD TECNICA DE COTOPAXI.

2.2 Organigrama funcional



LATECH		
Proyecto: UNIVERSIDAD TECNICA DE COTOPAXI		
Contenido: DIAGRAMA EQUIPOS DE CONECTIVIDAD	Dibujo: Federico Ortega	Hoja N°
Fecha: 5 de Octubre del 2006	Escala: S/E	2 de 7

2.2.1 Cuadros de distribución de la red

CUADRO DE DISTRIBUCIÓN DE RED 1

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:		Distribuidor Secundario SDF-21- BLOQUE B	
Ubicación del distribuidor:		Piso - 2, Cuarto de equipos RACK A	
Id de Pach Panel: 21A-3		Tipo de patch panel: **PP-24P C5e - PID-00174**	
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	Bvi 4	21A-301	BACKBONE VIDEO P2	P2
2	Bvi 5	21A-302	BACKBONE VIDEO P2	P2
3	Bvi 6	21A-303	BACKBONE VIDEO P2	P2
4	Bvi 7	21A-304	BACKBONE VIDEO P2	P2
5	Bvi 8	21A-305	BACKBONE VIDEO P2	P2
6	Bvi 9	21A-306	BACKBONE VIDEO P2	P2
7	Bvi 10	21A-307	BACKBONE VIDEO P2	P2
8	Bvi 11	21A-308	BACKBONE VIDEO P2	P2
9	Bvi 12	21A-309	BACKBONE VIDEO P2	P2
10	Bvi 13	21A-310	BACKBONE VIDEO P2	P2
11	Bvi 14	21A-311	BACKBONE VIDEO P2	P2
12	Bvi 15	21A-312	BACKBONE VIDEO P2	P2
13	Bvi 16	21A-313	BACKBONE VIDEO P2	P2
14	Bvi 17	21A-314	BACKBONE VIDEO P2	P2
15	Bvi 18	21A-315	BACKBONE VIDEO P2	P2
16	Bvi 19	21A-316	BACKBONE VIDEO P2	P2
17	Bvi 20	21A-317	BACKBONE VIDEO P2	P2
18	Bvi 21	21A-318	BACKBONE VIDEO P2	P2
19	Bvi 22	21A-319	BACKBONE VIDEO P2	P2
20	Bvi 23	21A-320	BACKBONE VIDEO P2	P2
21	Bvi 24	21A-321	BACKBONE VIDEO P2	P2
22	Bvi 25	21A-322	BACKBONE VIDEO P2	P2
23	Bvi 26	21A-323	BACKBONE VIDEO P2	P2
24	Bvi 27	21A-324	BACKBONE VIDEO P2	P2

CUADRO DE DISTRIBUCIÓN DE RED 2

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:		Distribuidor Secundario SDF-21- BLOQUE B		
Ubicación del distribuidor:		Piso - 2, Cuarto de equipos RACK B		
Id de Pach Panel: 21B-1		**Tipo de patch panel:** PP-24P C5e - PID-00174		
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	1	21B-101	AULA 1	P2
2	2	21B-102	AULA 1	P2
3	3	21B-103	EXPOSICION TRABAJO S0	P2
4	4	21B-104	EXPOSICION TRABAJO SE	P2
5	5	21B-105	AULA 4	P2
6	6	21B-106	AULA 3	P2
7	7	21B-107	AULA 5	P2
8	8	21B-108	AULA 11	P2
9	9	21B-109	AULA 10	P2
10	10	21B-110	AULA 9	P2
11	11	21B-111	AULA 8	P2
12	12	21B-112	EXPOSICION TRABAJO NE	P2
13	13	21B-113	EXPOSICION TRABAJO NO	P2
14	14	21B-114	AULA 7	P2
15	15	21B-115	AULA 6	P2
16	16	21B-116	CUBICULO PROFESORES	P2
17	17	21B-117	CUBICULO PROFESORES	P2
18	18	21B-118	ACCESS POINT HALL N0	P2
19	19	21B-119	ACCESS POINT HALL SE	P2
20		21B-120	LIBRE	P2
21	Bd1	21B-121	BACKBONE DATOS P2 - PB	P2
22	Bd2	21B-122	BACKBONE DATOS P2	P2
23	Bd3	21B-123	BACKBONE DATOS P2	P2
24	Bd4	21B-124	BACKBONE DATOS P2	P2

CUADRO DE DISTRIBUCIÓN DE RED 3

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:		Distribuidor Secundario SDF-21- BLOQUE B		
Ubicación del distribuidor:		Piso - 2, Cuarto de equipos RACK B		
Id de Pach Panel: 21B-2		**Tipo de patch panel: PP-24P C5e - PID-00174**		
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	1	21B-201	AULA 1	P3
2	2	21B-102	AULA 1	P3
3	3	21B-103	EXPOSICION TRABAJO S0	P3
4	4	21B-104	EXPOSICION TRABAJO SE	P3
5	5	21B-105	AULA 3	P3
6	6	21B-106	AULA 4	P3
7	7	21B-107	AULA 5	P3
8	8	21B-108	AULA 9	P3
9	9	21B-109	AULA8	P3
10	10	21B-110	AULA 7	P3
11	11	21B-111	AULA 6	P3
12	12	21B-112	EXPOSICION TRABAJO NE	P3
13	13	21B-113	ACCESS POINT HALL N0	P3
14	14	21B-114	ACCESS POINT HALL 0	P3
15	15	21B-115	ACCESS POINT HALL SE	P3
16		21B-116	LIBRE	P3
17		21B-117	LIBRE	P3
18		21B-118	LIBRE	P3
19		21B-119	LIBRE	P3
20		21B-120	LIBRE	P3
21	Bd1	21B-121	LIBRE	P3
22	Bd2	21B-122	LIBRE	P3
23	Bd3	21B-123	LIBRE	P3
24	Bd4	21B-124	LIBRE	P3

CUADRO DE DISTRIBUCIÓN DE RED 4

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:		Distribuidor Principal MDF-30 - Bloque C		
Ubicación del distribuidor:		Piso - PB, Cuarto de equipos RACK A		
Id de Pach Panel: 21B-2		**Tipo de patch panel: PF-6 - AFR-00112**		
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	FO2	30A-101	BLOQUE B, MDF-20	PB
2	FO2	30A-102	BLOQUE B, MDF	PB
3	FO2	30A-103	BLOQUE B, MDF	PB
4	FO2	30A-104	BLOQUE B, MDF	PB
5	FO2	30A-105	BLOQUE B, MDF	PB
6	FO2	30A-106	BLOQUE B, MDF	
7				
8				
9				
10				
11				
12				

CUADRO DE DISTRIBUCIÓN DE RED 5

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:		Distribuidor Principal MDF-30 - Bloque C		
Ubicación del distribuidor:		Piso - PB, Cuarto de equipos RACK A		
Id de Pach Panel: 21B-2		**Tipo de patch panel: PP-24P C5e - PID-00174**		
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	25	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
2	26	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
3	26	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
4	28	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
5	29	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
6	30	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
7	31	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
8	32	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
9	33	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
10	34	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
11	35	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
12	36	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
13	37	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
14	38	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
15	39	30A-301	SALA ACADEMICA - INVESTIGACION	MZ
16	40	30A-301	PROYECTOR - SALA ACADEMICA - INVESTIGACION	MZ
17		30A-301	LIBRE	
18		30A-301	LIBRE	
19		30A-301	LIBRE	
20		30A-301	LIBRE	
21		30A-301	LIBRE	
22		30A-301	LIBRE	
23		30A-301	LIBRE	
24		30A-301	LIBRE	

CUADRO DE DISTRIBUCIÓN DE RED 6

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:		Distribuidor Principal MDF-30 - Bloque C		
Ubicación del distribuidor:		Piso - PB, Cuarto de equipos RACK A		
Id de Pach Panel: 30A-4		**Tipo de patch panel: PP-24P C5e - PID-00174**		
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	V1	30A-401	TALLER ELECTRONICA CB5	PB
2	V2	30A-402	TALLER CERAMICA CB3	PB
3	V3	30A-403	TALLER METALES CB1	PB
4	V4	30A-404	TALLER CERAMICA CB3	PB
5	V5	30A-405	TALLER METALES CB1	PB
6	V6	30A-406	ATENCION BIBLIOTECA	P1
7	V7	30A-407	ADMINISTRACION BIBLIOTECA	MZ
8	V8	30A-408	ADMINISTRACION BIBLIOTECA	MZ
9	V9	30A-409	ADMINISTRACION BIBLIOTECA	MZ
10	V10	30A-410	MONEDERO - COMEDOR	P1
11		30A-411	LIBRE	
12		30A-412	LIBRE	
13		30A-413	LIBRE	
14		30A-414	LIBRE	
15		30A-415	LIBRE	
16		30A-416	LIBRE	
17		30A-417	LIBRE	
18		30A-418	LIBRE	
19		30A-419	LIBRE	
20		30A-420	LIBRE	
21		30A-421	LIBRE	
22	Vi 1	30A-422	TALLER ELECTRONICA CB5	PB
23	Vi 2	30A-423	TALLER CERAMICA CB3	PB
24	Vi 3	30A-424	TALLER METALES CB1	PB

CUADRO DE DISTRIBUCIÓN DE RED 7

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:		Distribuidor Principal MDF-40 - Edificio Verde		
Ubicación del distribuidor:		**Piso - Terraza, Cuarto de equipos RACK A**		
Id de Pach Panel: 40A-1		**Tipo de patch panel: PF-6 - AFR-00112**		
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	FO3	40A-101	Edificio Antiguo - MDF-0	TERRAZA
2	FO3	40A-102	Edificio Antiguo - MDF-0	TERRAZA
3	FO3	40A-103	Edificio Antiguo - MDF-0	TERRAZA
4	FO3	40A-104	Edificio Antiguo - MDF-0	TERRAZA
5	FO3	40A-105	Edificio Antiguo - MDF-0	TERRAZA
6	FO3	40A-106	Edificio Antiguo - MDF-0	TERRAZA
7				
8				
9				
10				
11				
12				

CUADRO DE DISTRIBUCIÓN DE RED 8

Usuario: UNIVERSIDAD TÉCNICA DE COTOPAXI
Dirección (d/c): LATACUNGA

Distribuidor:			Distribuidor Principal MDF-40 - Edificio Verde	
Ubicación del distribuidor:			Piso - TERRAZA, Relaciones Publicas RACK A	
Id de Pach Panel: 30A-4			**Tipo de patch panel: PP-24P C5e - PID-00174**	
Puert de Panel	**Id Cable**	**Id Canal**	**Ubicación área de trabajo**	**Planta**
1	1	40A-201	PLANIFICACION, MANTENIMIENTO Y CONSTRUCCION	P2
2	2	40A-202	PLANIFICACION, MANTENIMIENTO Y CONSTRUCCION	P2
3	3	40A-203	PLANIFICACION, MANTENIMIENTO Y CONSTRUCCION	P2
4	4	40A-204	PLANIFICACION, MANTENIMIENTO Y CONSTRUCCION	P2
5	5	40A-205	SECRETARIA GENERAL	P2
6	6	40A-206	PLANIFICACION FISICA	P2
7	7	40A-207	PLANIFICACION FISICA	P2
8	8	40A-208	PROCURADURIA	P2
9	9	40A-209	PROCURADURIA	P2
10	10	40A-210	CONSULTORIO MEDICO	P1
11	11	40A-211	SALA DE ESPERA	P1
12	12	40A-212	DIRECCION	P1
13	13	40A-213	TRABAJO SOCIAL	P1
14	14	40A-214	ORIENTACION VOCACIONAL	P1
15	15	40A-215	DIRECCION DE VINCULACION SOCIAL	PB
16	16	40A-216	DIRECCION DE VINCULACION SOCIAL	PB
17	17	40A-217	PRACTICA DOCENTE	TERRAZA
18	18	40A-218	RELACIONES PUBLICAS	TERRAZA
19	19	40A-219	RELACIONES PUBLICAS	TERRAZA
20	20	40A-220	RELACIONES PUBLICAS	TERRAZA
21	21	40A-221	RELACIONES PUBLICAS	TERRAZA
22		40A-222	LIBRE	
23		40A-223	LIBRE	
24		40A-224	LIBRE	

2.3 Análisis e interpretación de los resultados de la investigación de campo

POBLACIÓN

SUJETO	Nº
Alumnos del bloque B de la Universidad Técnica de Cotopaxi. (Especialidad Ingeniería en Informática y Sistemas Computacionales e Ingeniería en Diseño Gráfico)	1656
Docentes de la Carrera de Ciencias de la Ingeniería y Aplicadas de la Universidad Técnica de Cotopaxi.	74
Personal del Área de Redes y Servidores de la Universidad Técnica de Cotopaxi.	1
Personal administrativo de la Secretaría de la Carrera de Ciencias de la Ingeniería y aplicadas.	3
Total	1734

FUENTE: Encuesta.

REALIZADO POR: Grupo de Tesis.

MUESTRA

En honor a que tenemos una población extensa se procede a calcular la respectiva muestra para la aplicación de las encuestas, que exige la elaboración del Proyecto.

FÓRMULA Y APLICACIÓN

$$n = \frac{PQxN}{(N-1)\frac{E^2}{K^2} + PQ} \qquad n = \frac{0.25x1734}{(1734-1)\frac{(0.08)^2}{(2)^2} + 0.25} \qquad \boldsymbol{n = 143.41}$$

Donde: n = Tamaño de la muestra

PQ = Constante de varianza (0.25)

N = Tamaño de la población

E = Error máximo admisible (8%)

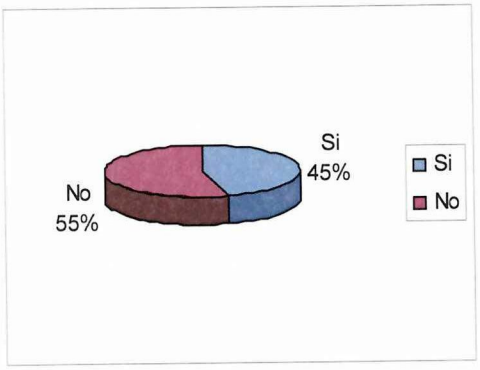
K = Constante de corrección (2)

Pregunta N° 1.- ¿Sabe lo que es monitorear?

Tabla 2.1.- Respuesta pregunta 1

Contestaciones	N. Encuestados	Porcentaje
Si	65	45,5
No	78	54,5
Total	**143**	**100,00**

Figura 2.1.- Respuesta pregunta 1



Fuente: Grupo Investigador

Análisis

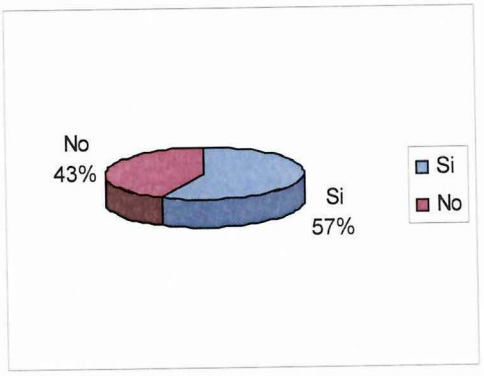
Se puede afirmar que el 54.55% de los encuestados en la Universidad Técnica de Cotopaxi con respecto a si saben lo que es monitorear contestaron que **NO**, mientras que el 45,45% dijeron que **SÍ**, pudiendo concluir entonces que lamentablemente hay una mayoría que no tiene conocimiento sobre el monitoreo.

Pregunta Nº 2.- ¿Conoce los beneficios que brinda una red?

Tabla 2.2.- Respuesta pregunta 2

Contestaciones	N. Encuestados	Porcentaje
Si	82	57,3
No	61	42,7
Total	**143**	**100,00**

Figura 2.2.- Respuesta pregunta 2



Fuente: Grupo Investigador

Análisis

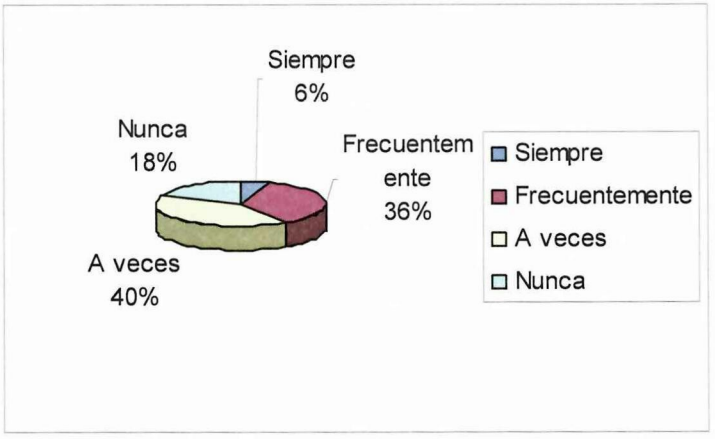
En relación a si los encuestados conocen los beneficios que brinda un red obtuvimos que el 57,3% si saben cuales son los beneficios de una red y el 42,73% los desconocen, pudiendo afirmar que la mayoría de encuestados están al tanto sobre cuales son los beneficios de las redes.

Pregunta Nº 3.- ¿Ha utilizado alguna vez una red monitoreada?

Tabla 2.3.- Respuesta pregunta 3

Contestaciones	N. Encuestados	Porcentaje
Siempre	8	5,6
Frecuentemente	51	35,7
A veces	58	40,6
Nunca	26	18,2
Total	**143**	**100**

Figura 2.3.- Respuesta pregunta 3



Fuente: Grupo Investigador

Análisis

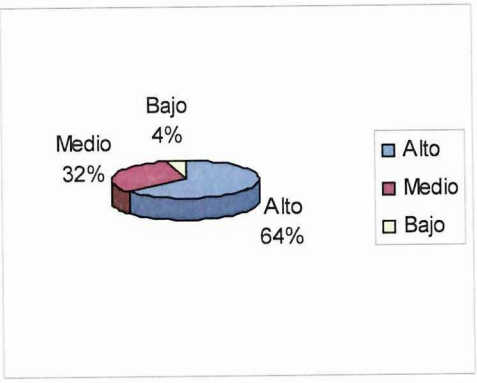
Respecto a si los encuestados han utilizado alguna vez una red monitoreada obtuvimos que un 5,6% las emplean siempre, un 35,7% las utilizan frecuentemente, un 40,6% las usan a veces y un 18,2% nunca han usado una red monitoreada, por lo que se concluye que hay un gran numero de encuestados que usan redes monitoreadas solo a veces.

Pregunta N° 4.- Piensa que una herramienta de monitoreo incrementa el rendimiento de una red en un porcentaje: alto, medio o bajo.

Tabla 2.4.- Respuesta pregunta 4

Contestaciones	N. Estudiantes	Porcentaje
Alto	91	63,64
Medio	46	32,17
Bajo	6	4,20
Total	**143**	**100,00**

Figura 2.4.- Respuesta pregunta 4



Fuente: Grupo Investigador

Análisis

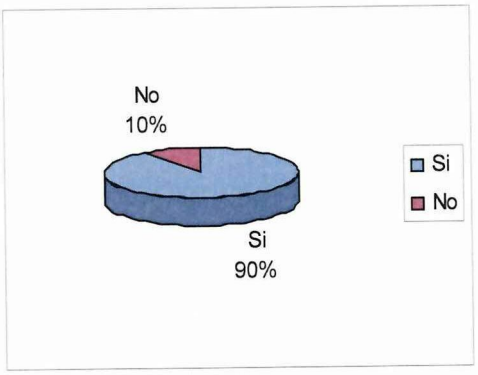
Referente a si una herramienta de monitoreo incrementa el rendimiento de una red los encuestados opinaron en un 63,64% que el rendimiento incrementaría en un alto porcentaje con dicha herramienta, el 32,17% opinó que habría un mediano rendimiento de la red y el 4,20% expresó que el rendimiento sería bajo; por lo que podemos aseverar que la mayoría de encuestados concluye en que el rendimiento de una red sería mayor con una herramienta para el monitoreo de la misma.

Pregunta Nº 5.- ¿Cree Ud. que una herramienta de monitoreo brindará información confiable?

Tabla 2.5.- Respuesta pregunta 5

Contestaciones	N. Estudiantes	Porcentaje
Si	129	90,21
No	14	9,79
Total	**143**	**100,00**

Figura 2.5.- Respuesta pregunta 5



Fuente: Grupo Investigador

Análisis

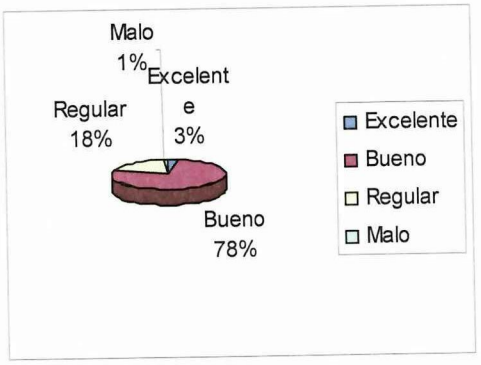
Respecto a si una herramienta de monitoreo brindará información confiable los encuestados opinaron en un 90,21% que si y tan solo el 9,79% dijo que no, por lo que se puede afirmar que la gran mayoría de encuestados confía en que las herramientas de monitoreo sí brindan información confiable.

Pregunta Nº 6.- Considera que el actual rendimiento de los equipos que conforman la red del bloque B de la Universidad Técnica de Cotopaxi es: excelente, bueno, regular o malo.

Tabla 2.6.- Respuesta pregunta 6

Contestaciones	N. Estudiantes	Porcentaje
Excelente	4	2,80
Bueno	111	77,62
Regular	26	18,18
Malo	2	1,40
Total	**143**	**100,0**

Figura 2.6.- Respuesta pregunta 6



Fuente: Grupo Investigador

Análisis

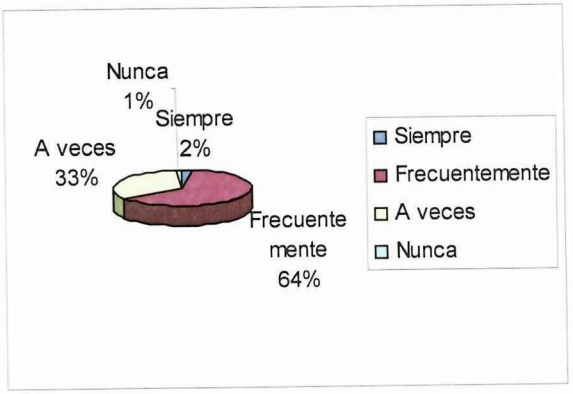
El 2,80% de los encuestados consideran que el actual rendimiento de los equipos de la red del bloque B de la Universidad Técnica de Cotopaxi es excelente, el 77,62% cree que su rendimiento es bueno, el 18,18% piensa que es regular y el 1,40% lo considera malo, con relación a estos resultados podemos concluir que la mayor parte de encuestados considera que la red de la Universidad y su rendimiento no es excelente tan solo bueno, lo que no es óptimo.

Pregunta N° 7.- ¿Piensa que existe dificultad en la red del bloque B de la Universidad Técnica de Cotopaxi?

Tabla 2.7.- Respuesta pregunta 7

Contestaciones	N. Estudiantes	Porcentaje
Siempre	3	2,1
Frecuentemente	91	63,6
A veces	47	32,9
Nunca	2	1,4
Total	**143**	**100,0**

Figura 2.7.- Respuesta pregunta 7



Fuente: Grupo Investigador

Análisis

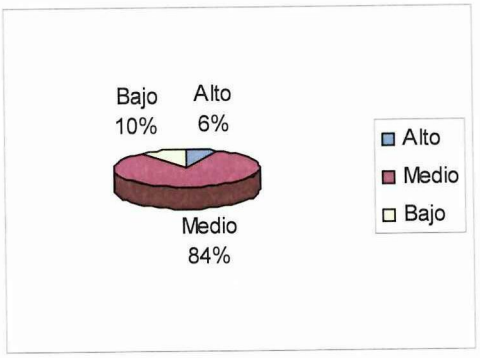
El 2,10% de encuestados opinó que siempre existe dificultad en la red del bloque B de la Universidad Técnica de Cotopaxi, el 63,64% dijo que hay dificultad frecuentemente, el 32,87% cree que solo a veces existen problemas y el 1,40% expresó que nunca hay dificultades, por estas razones se puede testificar que la gran mayoría de encuestados coinciden en que las dificultades en la red se presentan frecuentemente.

Pregunta N° 8.- Cree que el nivel de seguridad en la red del bloque B de la Universidad Técnica de Cotopaxi es: alto, medio o bajo.

Tabla 2.8.- Respuesta pregunta 8

Contestaciones	N. Estudiantes	Porcentaje
Alto	9	6,29
Medio	119	83,22
Bajo	15	10,49
Total	**143**	**100**

Figura 2.8.- Respuesta pregunta 8



Fuente: Grupo Investigador

Análisis

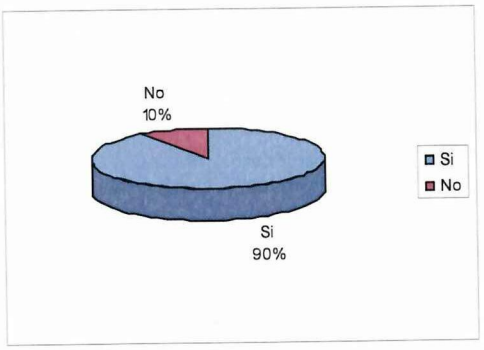
Respecto al nivel de seguridad de la red del bloque B de la Universidad Técnica de Cotopaxi los encuestados en un 6,9% piensan que existe un nivel alto, el 83,22% opina que el nivel es medio y el 10,49 creen que el nivel de seguridad es bajo, con base en los resultados obtenidos se puede afirmar que la seguridad de la red se encuentra en un nivel medio, mismo que no satisface a los encuestados.

Pregunta Nº 9.- ¿Considera necesario instalar una herramienta para monitorear la red en la Universidad Técnica de Cotopaxi?

Tabla 2.9.- Respuesta pregunta 9

Contestaciones	N. Estudiantes	Porcentaje
Si	129	90,21
No	14	9,79
Total	**143**	**100,00**

Figura 2.9.- Respuesta pregunta 9



Fuente: Grupo Investigador

Análisis

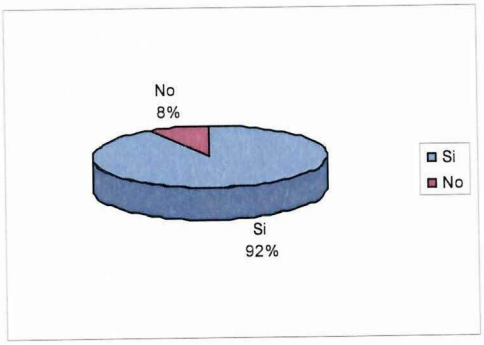
Según la encuesta realizada el 90,21% considera que es necesario instalar una herramienta para monitorear la red y solo el 9,79% piensa que no lo es, por lo que se concluye que la instalación de una herramienta para monitorear la red es necesaria en el bloque B de la Universidad Técnica de Cotopaxi.

Pregunta N° 10.- ¿Cree Ud. que con una herramienta de monitoreo mejoren los servicios prestados por la red en el bloque B de la Universidad Técnica de Cotopaxi?

Tabla 2.10.- Respuesta pregunta 10

Contestaciones	N. Estudiantes	Porcentaje
Si	131	91,61
No	12	8,39
Total	**143**	**100,00**

Figura 2.10.- Respuesta pregunta 10



Fuente: Grupo Investigador

Análisis

El 91,61% de encuestados cree que con una herramienta de monitoreo mejoraran los servicios prestados por la red en el bloque B de la Universidad Técnica de Cotopaxi mientras que el 8,39% opina que no, con estas razones afirmamos que la herramienta de monitoreo es imperiosa para mejorar los servicios suministrados por la red.

2.4 Verificación de la hipótesis

Luego de haber aplicado las encuestas, tabulado y analizado los resultados y cumpliendo con lo previsto se logró comprobar la hipótesis planteada:

"El diseño de una herramienta para el monitoreo mejorará el control de los servicios y de la red del bloque B de la UTC campus San Felipe", comprobación lograda gracias a los resultados obtenidos en las encuestas realizadas y basándonos en los porcentajes logrados se concluye que el tema de tesis "Diseño de una herramienta para el monitoreo de servicios y de la red del bloque B de la Universidad Técnica de Cotopaxi campus san Felipe" será un instrumento de suma utilidad para la administración de la red.

CAPÍTULO III

"DISEÑO DE UNA HERRAMIENTA PARA EL MONITOREO DE SERVICIOS Y DE LA RED DEL BLOQUE B DE LA UNIVERSIDAD TECNICA DE COTOPAXI CAMPUS SAN FELIPE"

3.1 Presentación

El diseño de una Herramienta de Monitoreo permitirá una mejor administración y un mayor control de la red del Bloque B de la Universidad Técnica de Cotopaxi, herramienta que pasará a formar parte de la tecnología de punta con la que ya cuenta el Departamento de Servicios Informáticos de dicha Institución, este Departamento provee de varios servicios a los diferentes Campus y Bloques Académicos de la Universidad, tales como internet en las diferentes áreas administrativas y en las salas de cómputo disponibles para el uso de estudiantes y maestros para sus tareas o clases, correo institucional en la red interna, biblioteca web, inscripciones, entro otros.

La Herramienta de Monitoreo lo que hace básicamente es captar información relevante sobre la red y la transmisión de datos en la misma, por ejemplo capta el IP de los equipos en red, los protocolos empleados en la transferencia de datos, detecta la cabecera de los paquetes y el id de los mismos, muestra el puerto de origen y el de destino y despliega el tamaño del paquete.

Resultan de particular importancia así mismo las gráficas que se crean a partir de la información receptada, se expone un gráfico de barras que indica las páginas más visitadas y un gráfico de líneas que publica el ancho de banda usado por cada equipo.

3.2 Justificación de la propuesta

EL Departamento de Servicios Informáticos de la Universidad Técnica de Cotopaxi con el pasar del tiempo ha venido realizando progresos significativos en el área tecnológica siendo estos cada vez mas notorios, así pues se puede mencionar la implementación de las Redes de Área Local Virtuales las mismas que han permitido incrementar en buena medida el número de usuarios de maquinas o puntos de red así como también podemos decir que es una realidad la implementación de las Jireles, que ayudan a que más usuarios puedan utilizar servicios tales como el Internet o el intercambio de información entre usuarios de red.

Se concibió la idea de Diseñar una Herramienta de Monitoreo con el fin de colaborar con el avance tecnológico de la Universidad a través de su Departamento de Servicios Informáticos para que exista una mejor administración de la red y esto solo es posible si se puede monitorear la misma.

Para el monitoreo de los servicios y de la red del bloque B de la Universidad Técnica de Cotopaxi se emplea un Tipping Point X 505 3 COM en lo referente a hardware que se encarga del:

- Filtrado de Página Web
- Filtrado para direccionamiento
- Control del acceso a protocolos

Además se utiliza el SQUID 6.02 en Centos 5.5 como software de monitoreo para lo que se refiere primordialmente al control de páginas web, como complemento estará la Herramienta de Monitoreo que se explotará fundamentalmente en los laboratorios pero teniendo en cuenta que se puede utilizar en cualquier dependencia de la Institución, obedeciendo al criterio del administrador de la herramienta y al fin que se quiera obtener con ella.

Se considera factible el Diseño de una Herramienta de Monitoreo gracias a los conocimientos adquiridos durante toda una formación académica con la que se obtuvo la capacitación necesaria para el uso del lenguaje de la herramienta de programación, que en este caso es VisualStudio 2010, el lenguaje de codificación C#.

3.3 Objetivos

Objetivo General

- Diseñar una herramienta para el monitoreo de servicios y de la red del bloque B de la Universidad Técnica de Cotopaxi Campus San Felipe.

Objetivos Específicos

- Identificar los principales aspectos a monitorear en la red del bloque B de la Universidad Técnica de Cotopaxi Campus San Felipe.
- Desarrollar la interfaz amigable de la herramienta de monitoreo para que registre los equipos conectados en red.
- Desplegar en pantalla la información recogida y generar gráficos a partir de esos datos para facilitar la toma de decisiones por parte del administrador.

.

3.4 Factibilidad de la propuesta

Una vez planteada la propuesta de diseñar una herramienta de monitoreo que ayude en el control de la red del Bloque B de la Universidad Técnica de Cotopaxi, campus San Felipe, se procedió a la recolección de información y el diálogo con los administradores de las salas y con los miembros del Departamento de Desarrollo de Software.

Luego de un análisis se define como realizable el diseñar la herramienta de monitoreo, pues los requisitos expuestos y los instrumentos disponibles permiten que este proyecto se pueda desarrollar en un tiempo prudencial y con el apoyo de quienes utilizaran el sistema y el grupo investigador.

El sistema a desarrollarse por parte del grupo investigador se basará en una fusión de varios lenguajes de programación web como son: Visual Studio 2010 y C#.

3.5 Impacto de la propuesta

Esta Herramienta de Monitoreo constituye un instrumento para incrementar la eficiencia de la red a través de la medición de parámetros tales como la cantidad de páginas visitadas y el ancho de banda asignado a cada equipo conectado en red, con lo cual se verán beneficiados todos los usuarios de la misma porque si por a o b circunstancia un equipo hace uso de un mayor ancho de banda que el resto el administrador de la herramienta de monitoreo inmediatamente tomará cartas en el asunto y solucionará el problema para que todos los que usen los servicios de la red tengan iguales beneficios de la misma.

Además tiene una interfaz amigable, muy sencilla de usar, el software posee un manual de usuario que permitirá al Administrador del mismo llevar un mejor control de la red.

3.6 Desarrollo de la propuesta

3.6.1 Descripción de las herramientas de programación

Visual Studio 2010

La herramienta manejada en el desarrollo del sistema de monitoreo es Microsoft Visual Studio ya que presenta un entorno de desarrollo integrado para sistemas operativos Windows, que es la plataforma empleada en el Bloque B de la Universidad Técnica de Cotopaxi, siendo más específicas se trabaja con Windows XP Profesional.

Visual Studio tiene varias versiones pero específicamente es empleada la versión 2010 de esta herramienta porque soporta mayor número de lenguajes de programación que versiones anteriores, entre ellos C#, que es el lenguaje a manejar en la codificación.

Otro motivo de la decisión de explotar Visual Studio 2010 es que cuenta con un gran logro a su haber, como es el hecho de que puede ser ejecutado sin problema alguno con el sistema Operativo Windows 7, que es la posible plataforma a utilizarse.

C#

Es un lenguaje de programación orientado a objetos, desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA (European Computer Manufacturers Association) e ISO (International Standarization Organization).

C# forma parte de la plataforma .NET, ésta es una interfaz de programación de aplicaciones (API) y escogimos este lenguaje de programación por su versatilidad.

Infragistics

Es un complemento de VisualStudio 2010, ésta herramienta adicional es aprovechada para ahorrar tiempo ya que permite una rápida creación de interfaces de usuario, mismos que son amigables y de fácil uso para el usuario.

3.6.2 Descripción de la metodología

3.6.2.1 Tipo de Investigación.- El tipo de investigación a utilizar en el desarrollo del presente proyecto es de tipo descriptiva, ya que este tipo de investigación nos permitirá detallar todas las características del problema, así como también sus causas y efectos.

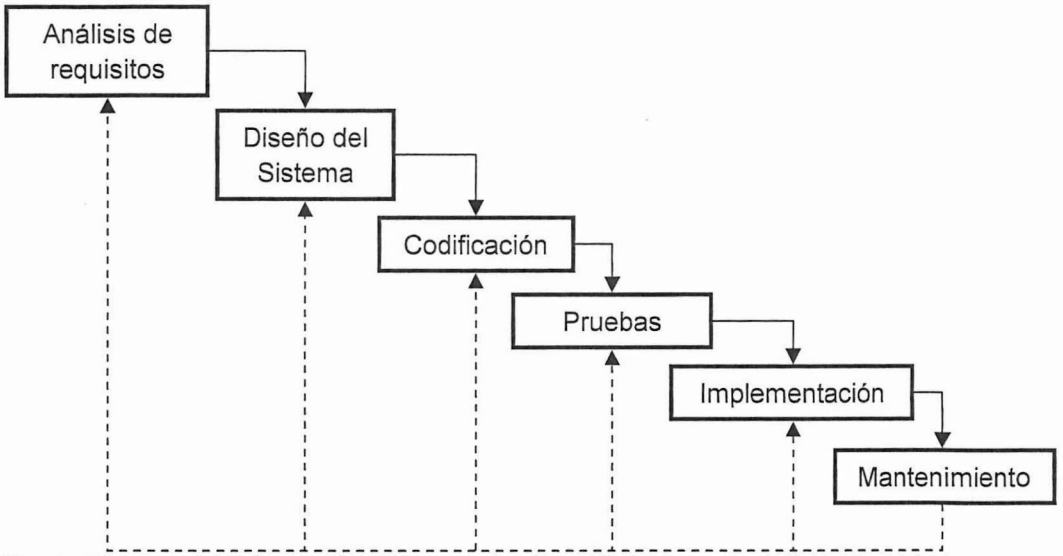
3.6.2.2 Metodología.- Para llegar a cumplir con los objetivos fijados en este trabajo, es necesario realizar una serie de experimentos o pruebas en la red del bloque B de la Universidad Técnica de Cotopaxi, por lo tanto el grupo de investigación ha decidido desarrollar este proyecto de acuerdo a la Metodología Experimental.

3.6.3 Ciclo de vida del Software / Modelo Cascada

El desarrollo en cascada, también llamado modelo en cascada, es el enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, de forma tal que el inicio de cada etapa debe esperar a la finalización de la inmediatamente anterior.

La Metodología de Desarrollo en Cascada consta de las siguientes etapas:

FIG. 1 FASES DEL MODELO EN CASCADA



Fuente: Ingeniería del Software: Un enfoque práctico, Roger S. Presuman, 3ra Edición, Pag. 26-30

3.6.3.1 Análisis de requisitos

El Análisis de Requisitos en la presente investigación busca conocer y describir de una forma general el ámbito dónde se pondrá en ejecución el sistema de monitoreo y los necesidades que va a cubrir, para ello es necesario especificar que en el Bloque B de la Universidad Técnica de Cotopaxi existen cinco laboratorios, de los cuales los Laboratorios 1 y 2 con 22 y 15 máquinas respectivamente cuentan con internet que se lo habilita cuando es necesario y están diseñados para el uso exclusivo de los estudiantes de la Carrera de Diseño Gráfico; los Laboratorios 3 y 4 con 36 y 31 computadoras respectivamente son empleados para los estudiantes de la Carrera de Ingeniería en Informática y Sistemas Computacionales y si la situación lo amerita se habilita el servicio de internet en ellos; y por último el Laboratorio 5 que cuenta con 45 PCs, de las cuales 25 son destinadas para internet y las 20 restantes para uso de los estudiantes y docentes que los requieran sin importar su especialidad.

Necesidades

El Bloque B, al igual que el resto de instalaciones que conforman la Universidad Técnica de Cotopaxi están conectadas en red y poseen el servicio de internet; servicio que es muy utilizado sobre todo en las salas de cómputo e internet, mismas que son empleadas por docentes y estudiantes como:

- Salas de clase.
- Se usan para impartir seminarios, cursos, conferencias con audiovisuales y/o virtuales.
- Se aprovechan para hacer consultas, tareas e investigaciones a través de internet.
- Adicionalmente para trabajos que no requieren de internet, lo que se conoce al momento de registrar al usuario como: para uso.

Todos estos beneficios se ven opacados por inconvenientes como virus que se descargan junto con las páginas de internet que son visitadas y se quedan alojados en las máquinas, para ello se han tomado medidas como congelar a las máquinas para evitar que los virus permanezcan.

Otro inconveniente es la descarga de programas de música o música en sí, esto disminuye mucho el ancho banda del resto de máquinas lo que provoca que el internet se vuelva deficiente en lo que respecta al tiempo empleado en la respuesta a la solicitud de los usuarios del las otras PCs.

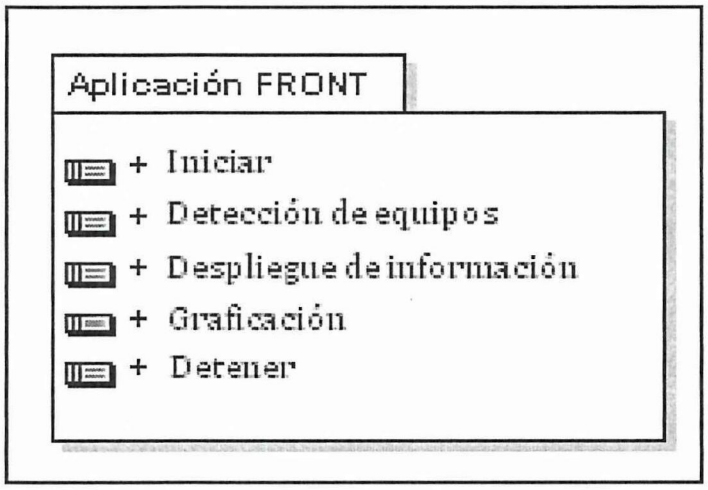
También están restringidas páginas sociales, de juegos y de pornografía, estas restricciones vienen de un tipping point y también del servidor, pese a ello se puede acceder a las mismas a través de páginas de búsqueda como el google.

Estos problemas afectan negativamente a los usuarios, por ejemplo en el tiempo que se tardan en realizar las consultas por el ancho de banda disminuido o por los

virus que hacen defectuoso el funcionamiento de la máquina, todo esto se puede enmendar empleando una Herramienta de Monitoreo.

Se debe aclarar en este punto que la Herramienta de Monitoreo, tema de esta investigación, no emplea una base de datos en vista que la misma solo funciona como un monitor de red y por lo tanto su ejecución solo permite capturar datos del flujo de información en tiempo real.

Requerimientos Funcionales



La Aplicación Front es la que utilizará el administrador de la herramienta que será el encargado de evaluar el comportamiento de la red de acuerdo a los resultados obtenidos del instrumento de monitoreo.

Iniciar

La herramienta debe empezar a ejecutarse en el momento en que el administrador pulse el botón "Iniciar".

Detección de equipos

Es necesario que la herramienta de monitoreo registre los IPs de los equipos conectados en red una vez iniciado el monitoreo.

Despliegue de información

La herramienta de monitoreo debe desplegar información adicional al ip de los equipos de la red, información como el url de las páginas web visitadas, el tamaño del paquete, los puertos de origen y destino.

Graficación

La información derivada del monitoreo debe presentarse gráficamente de manera forma clara y sencilla para ser interpretada fácilmente.

Detener

La herramienta monitoreadora de la red debe finalizar inmediatamente las acciones que se estén ejecutando cuando el administrador pulse el botón "Detener".

Requerimientos No Funcionales

Usabilidad

La interfaz para el administrador de la herramienta de monitoreo debe ser familiar y sencilla de utilizar, pudiendo en el último de los casos recurrir a la intuición basada en los íconos o botones empleados para dar inicio a cada diferente acción de la herramienta.

Requerimientos de Rendimiento

La Herramienta de Monitoreo, mantendrá un óptimo rendimiento, ya que será desarrollado con un conjunto de herramientas de última tecnología como son C# y VisualStudio 2010 que soportan grandes flujos de información.

Requerimientos Tecnológicos

Hardware

El computador con la Herramienta de Monitoreo debe contemplar las siguientes características:

Procesador:	Pentium IV de 800 Mhz o superior
Memoria RAM:	256 MB o superior
Disco Duro:	80 Gb o superior
Tarjeta de red:	10/100 o superior

Los computadores a monitorear deben contemplar las siguientes características:

Procesador:	Pentium III de 866 Mhz o superior
Memoria RAM:	128 MB o superior
Disco Duro:	30 Gb o superior
Tarjeta de red:	10/100 o superior

Software

Computador con la Herramienta de Monitoreo debe contemplar las siguientes características:

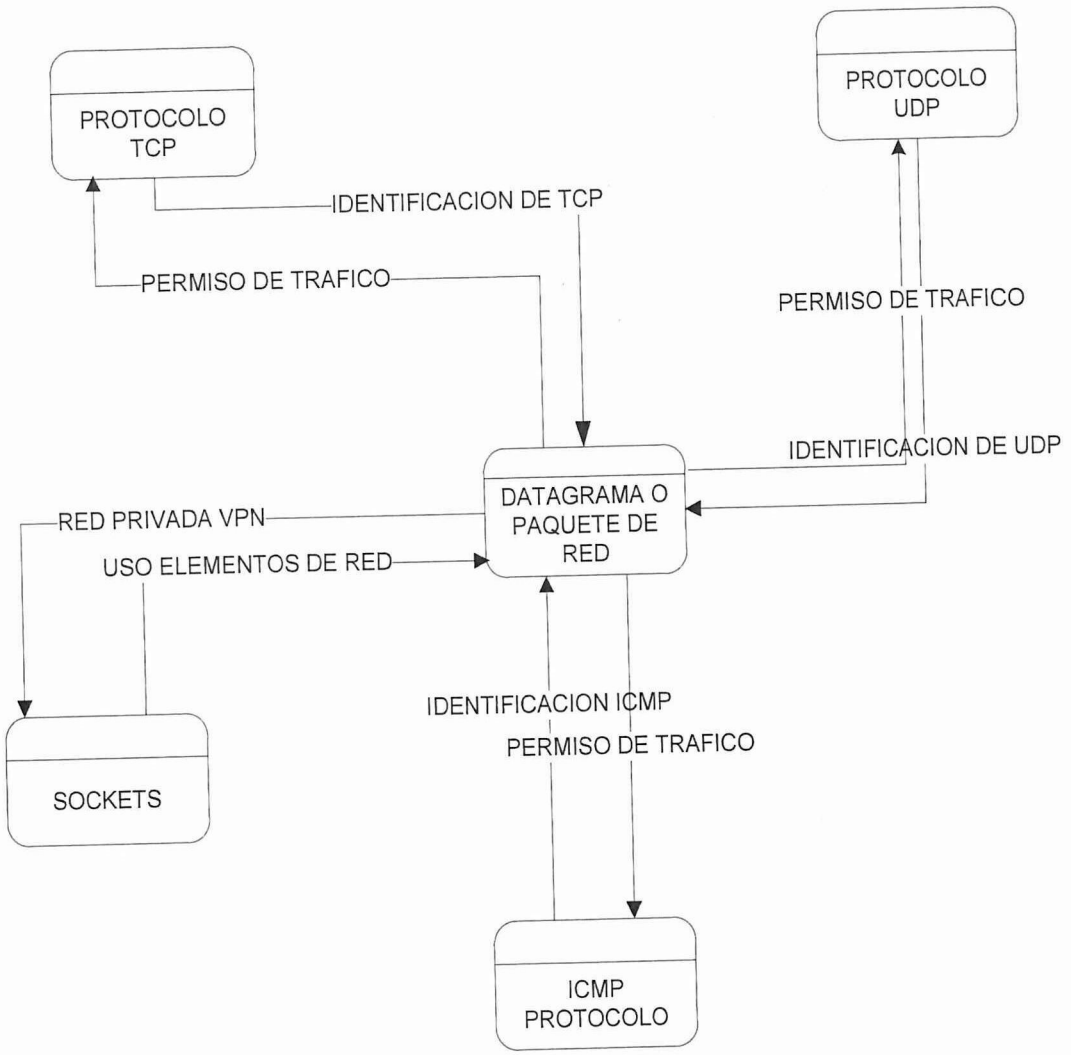
Sistema Operativo Windows XP Profesional, Windows 2000 Server, Windows 2003 Server o Windows 7.
Visual Studio 2010
Infragistics.
Internet Explorer 6.0

Computadores a monitorear debe contemplar las siguientes características:

Sistema Operativo Windows 95 o superior.
Internet Explorer

3.6.3.2 Diseño del Sistema

Para cumplir con lo estipulado en esta fase se analiza y determina lo siguiente:



El programa debe capturar los paquetes de los siguientes tipos: TCP, IP y UDP de dichos paquetes la información que mostrará en pantalla se especifica en la siguiente tabla:

Protocolo	Información a mostrar
TCP	• Ip • Puerto de origen • Puerto de destino • Cabecera • Tamaño • Id
UDP	
IP	

Estos datos son capturados por la herramienta de monitoreo a través de sockets en tiempo real; a partir de estas referencias se obtiene una primera gráfica de las páginas más visitadas mediante la contabilización general de las URL de los equipo conectados en red, y una segunda gráfica del ancho de banda con base al tamaño del paquete recibido por cada máquina de la red monitoreada.

Una vez especificado lo que debe hacer la herramienta de monitoreo se emplean las técnicas de Usabilidad expuestas en la asignatura de Interacción Humana con los Computadores (IHO); para dejar claro que es IHO, KUO, Benjamín C, manifiesta lo siguiente: "Es el estudio de la gente, la tecnología, las formas y como se influencian", en su libro Sistemas Automáticos de Control, pag. 128.

Aplicación de las Técnicas de IHO al Interfaz

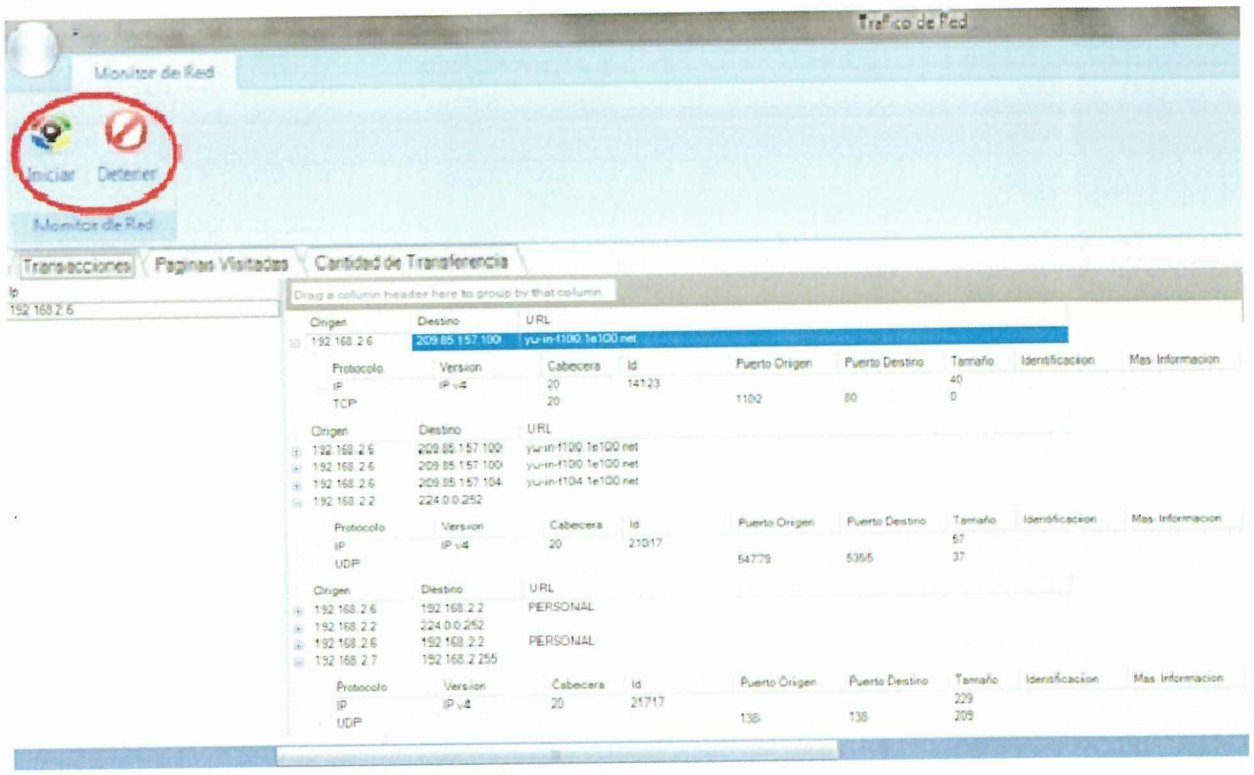
Para realizar el análisis del Interfaz, debemos estudiar opción a opción, en este caso resultó muy sencillo dada la simplicidad. La idea es que para cada tarea se debe realizar unas preguntas:

- **P1: ¿Está la acción correcta disponible en el interfaz?**
- **P2: ¿Cuan fácil se puede relacionar la descripción de la acción (icono, etiqueta, nombre del comando,...) con nuestro objetivo?**

- **P3: ¿La respuesta del sistema a la acción seleccionada muestra progreso hacia el objetivo final del usuario?**

Con estas respuestas se obtienen los elementos de evaluación y con ellos se ve donde se debe mejorar y es lo que se aplicó para la generación de las tres pantallas que dispone la herramienta de monitoreo.

FIG. 3 GUI FINAL DEL PROGRAMA



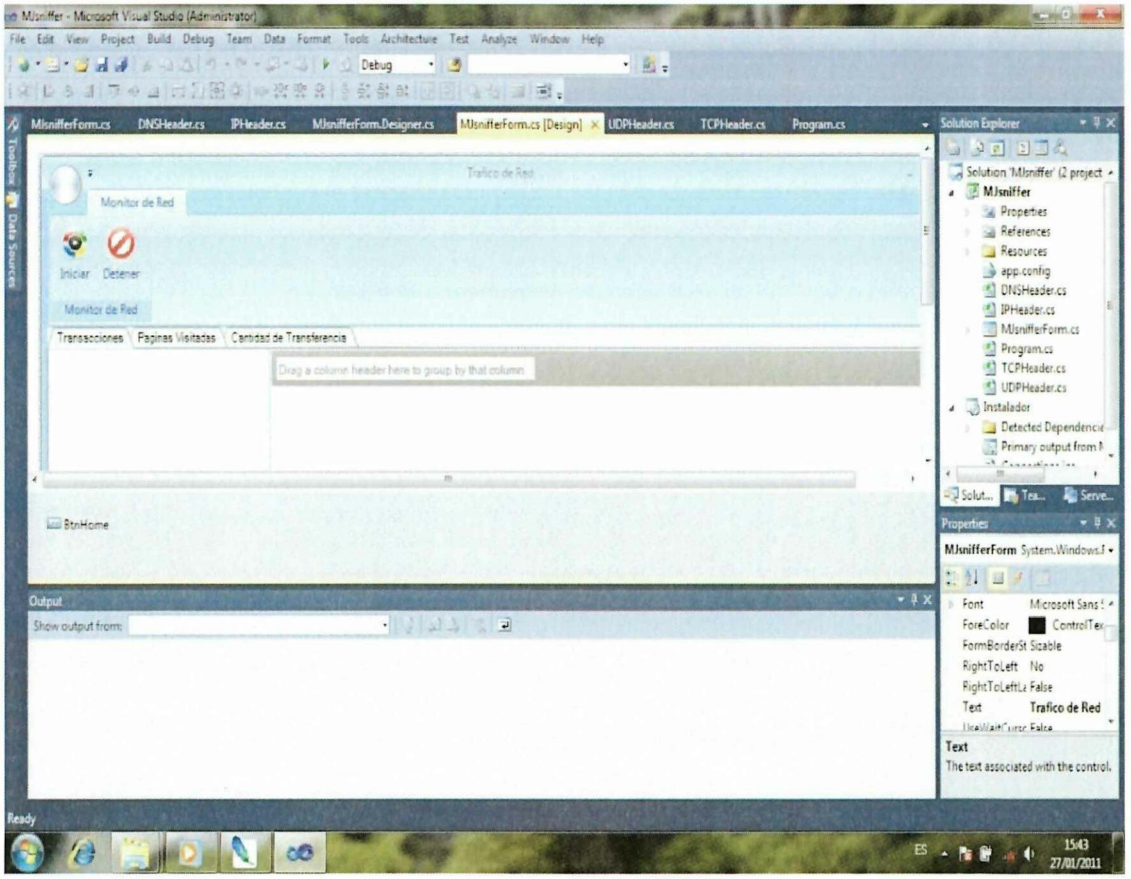
Fuente: El grupo investigador

En esta primera pantalla se observa de forma general el estilo que ha tomado el programa tras la aplicacaión de las técnicas de IHO en cuanto a la evaluación del GUI, y la eliminación de los elementos innecesarios. Finalmente ha resultado un programa muy sencillo de usar, que da la información básica de lo capturado en red y lo muestra de forma clara.

3.6.3.3 Codificación

Es la fase en donde se implementa el código fuente, haciendo uso de prototipos así como de pruebas y ensayos para corregir errores. Dependiendo del lenguaje de programación y su versión se crean las bibliotecas y componentes reutilizables dentro del mismo proyecto para hacer que la programación sea un proceso mucho más rápido.

FIG. 5 FORMULARIO DE VISUALSTUDIO 2010



Fuente: El grupo investigador

Ver Anexo 1. Codificación

Se crearon los siguientes formularios para realizar las acciones requeridas en las etapas anteriores y éstas se describen específicamente en cada uno de ellos.

MJSNIFFERFORM.DESIGNER.CS

Es el formulario principal de la aplicación, la misma que esta compuesta de todos los componentes de los formularios.

APP.CONFIG

Clase de la versión de las variables de ambiente, este utiliza XML 1.0

DNSHEADER.CS

Clase de uso de parámetros y de clases principales del servidor de dominios

IPHEADER.CS

Clases de defunción de protocolos de la aplicación

TCPHEADER.CS

Clase de especificación del TCP

UDPHEADER.CS

Clase de especificación del UDP

3.6.3.4 Pruebas

Los elementos, ya programados, se ensamblan para componer el sistema y se comprueba que funciona correctamente y que cumple con los requisitos, antes de ser entregado al usuario final.

Se efectuaron diferentes pruebas con la herramienta de monitoreo; los tres primeros ensayos se efectuaron en tres distintos centros de cómputo que para efectos del presente documento se conocerán como C.Cómputo1, en dónde se monitorearon cuatro máquinas, C.Cómputo2, en dónde también se monitorearon 4 ordenadores y C.Cómputo3, que cuenta con 3 Pc's, registrando la herramienta de monitoreo los ítems detallados en la tabla siguiente:

	C.Cómputo 1				C.Cómputo 2				C.Cómputo 3		
	Pc1	Pc2	Pc3	Pc4	Pc1	Pc2	Pc3	Pc4	Pc1	Pc2	Pc3
Dirección IP (Equipos en red)	√	√	√	√	√	√	√	√	√	√	√
Versión (Internet)	√	√	√	√	√	√	√	√	√	√	√
Protocolos	√	√	√	√	√	√	√	√	√	√	√
Puerto de Origen y Destino	√	√	√	√	√	√	√	√	√	√	√
Tamaño	√	√	√	√	√	√	√	√	√	√	√
Gráfico1(*Barras*) *Páginas Web*	√	√	√	√	√	√	√	√	√	√	√
Gráfico2(*Líneas*) *Ancho de Banda*	√	√	√	√	√	√	√	√	√	√	√

Ver Anexo 2. Pruebas externas

La prueba final fue en el laboratorio 5 del Bloque B de la Universidad Técnica de Cotopaxi, el que cuenta con 45 máquinas de las cuales 25 tienen acceso a Internet y esos ordenadores precisamente fueron los monitoreados y con los que la herramienta de monitoreo mantuvo un correcto comportamiento, registrando los siguientes ítems:

	UTC - Bloque B / Laboratorio 5				
	Pc1 - Pc5	Pc6 - Pc10	Pc11 - Pc15	Pc16 - Pc20	Pc21 - Pc25
Dirección IP (Equipos en red)	√	√	√	√	√
Versión	√	√	√	√	√
Protocolos	√	√	√	√	√
Puerto de Origen y Destino	√	√	√	√	√
Tamaño	√	√	√	√	√
Gráfico1(*Barras*) *Páginas Web*	√	√	√	√	√
Gráfico2(*Líneas*) *Ancho de Banda*	√	√	√	√	√

Ver Anexo3. Prueba Lab. 5 UTC

Las respectivas redes del C.Cómputo1, C.Cómputo2 y C.Cómputo3 están diseñadas con una topología de bus en la que la Herramienta de Monitoreo funcionó sin presentar problema alguno; por su parte el laboratorio 5 del Bloque B de la Universidad Técnica de Cotopaxi, campus ubicado en San Felipe posee una topología en estrella, topología en la que también la herramienta de monitoreo se desempeñó adecuadamente.

3.6.3.5 Implementación

Es la fase en donde el usuario final ejecuta el sistema, para ello el o los programadores ya realizaron exhaustivas pruebas para comprobar que el sistema no falle.

Una vez que el grupo de investigación se aseguró de la funcionalidad de la herramienta se la implanta teniendo en cuenta que se requiere para ello cumplir de forma estricta con los requerimientos mínimos ya detallados anteriormente, tanto de Hardware para un buen rendimiento del equipo en su uso, como de Software especialmente en el computador que haga las veces de monitor de la red.

3.6.3.6 Mantenimiento

Es la mantención del Software ya que al utilizarlo como usuario final puede ser que no cumpla con todas nuestras expectativas y puede surgir la necesidad de cambios, bien para corregir errores o bien para introducir mejoras.

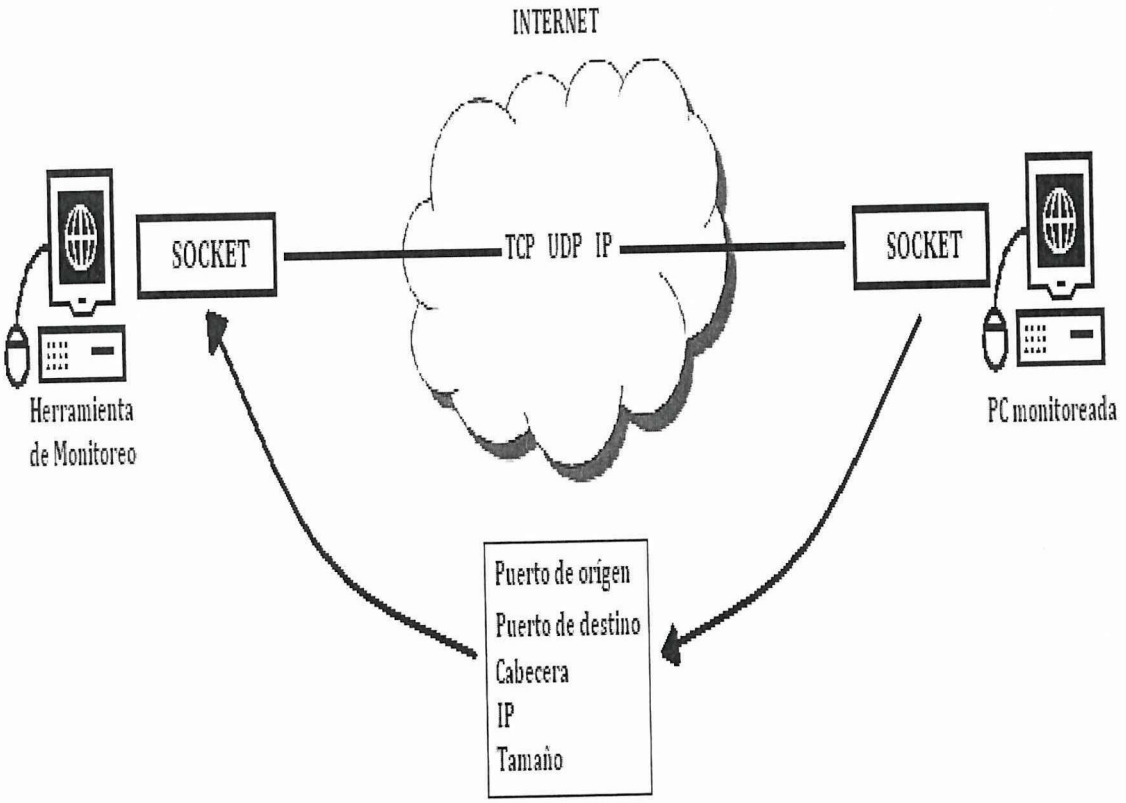
El mantenimiento de la herramienta de monitoreo se lo efectuará de acuerdo a las necesidades que surjan durante su uso y operación; dentro del mantenimiento del sistema es importante destacar la necesidad de respaldar los datos e información creando copias de seguridad de la herramienta a fin de evitar posibles pérdidas de

datos e información y si los hubiere tener a mano respaldo actualizados frecuentemente para poder enfrentar una posible emergencia.

Por último se debe indicar que el mantenimiento de la herramienta de monitoreo estará a cargo del Área de Desarrollo de Software de la Dirección de Servicios Informáticos, ya que cuentan con las herramientas de desarrollo y serán quienes decidan dónde y cuando emplearla.

3.6.4 Mapa de navegación de la aplicación

FIG. 6 FUNCIONAMIENTO DE LA HERRAMIENTA DE MONITOREO



Fuente: Grupo de investigación

3.6.5 Elaboración del manual de usuario del sistema

El manual del usuario contiene datos referentes al correcto y adecuado uso de la herramienta de monitoreo, a éste se debe recurrir al producirse algún error o al existir alguna inquietud en cuanto a la interacción con este software. El manual del usuario en detalle se encuentra en la sección anexos.

Ver Anexo4. Manual del Usuario

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- ✓ Mediante el desarrollo de la investigación planteada se puede afirmar que el diseño de una herramienta para el monitoreo de servicios y de la red del bloque B de la Universidad Técnica de Cotopaxi, campus San Felipe ayuda de manera significativa a la adecuada administración, control y distribución de los servicios de la red del ya mencionado bloque.

- ✓ Con el presente proyecto se pudo aplicar todos los conceptos adquiridos durante nuestros estudios realizados, así como también se pudo obtener nuevos conocimientos gracias a las investigaciones realizadas.

- ✓ El software diseñado se realizó acorde a las necesidades y requerimientos del bloque en mención, el mismo que se encuentra satisfecho con los logros alcanzados según sus necesidades.

- ✓ El diseño de una herramienta para el monitoreo de red será de gran ayuda a los administradores de la red de la Universidad Técnica de Cotopaxi, que son los encargados de resolver los problemas que se presentan.

.

RECOMENDACIONES

- ✓ Es importante tomar en cuenta los requerimientos tanto de software como de hardware, para un correcto funcionamiento de la herramienta de monitoreo.

- ✓ Para lograr un software con éxito es necesario realizar pruebas constantes del mismo con los usuarios que lo manejarán, para poder satisfacer requerimientos y necesidades en el camino.

- ✓ Se debe trabajar con herramientas que estén acorde a los avances tecnológicos de tal manera que el sistema desarrollado sea lo más confiable posible.

- ✓ Dar mayor apertura a los estudiantes de Ingeniería en Informática y Sistemas Computacionales para sistematizar los procesos de las diferentes dependencias universitarias mediante la implementación de sistemas informáticos y así optimizar su atención.

BIBLIOGRAFÍA

BIBLIOGRAFÍA CITADA

- KUO, Benjamín C., Sistemas Automáticos de Control, México, 1996.
- MUNICH, Lourdes, Métodos y Técnicas de Investigación, México, 1990.
- PRESSMAN, Roger S.: "Ingeniería de Software. Un enfoque práctico." Quinta edición. McGraw-Hill. Madrid. 2002.

BIBLIOGRAFÍA CONSULTADA

- CHARTE O. Francisco, Programación con Visual C# .NET, Madrid, 2002.
- LEIVA ZEA, Francisco, Nociones de Metodología de Investigación Científica, Cuarta Edición, Quito – 1996.
- SAMPIERI ROBERTO & COAUTORES, Metodología de la Investigación. Segunda Edición., México, 1998.
- TAMAYO Y TAMAYO, Mario, El Proceso de la Investigación Científica, Tercera Edición, México 1997.
- ULLOA, Francisco, Investigación 2000, Latacunga, 2004.
- LOOMIS, Mary E. S., Estructura de Datos y Organización de Archivos, México, 1991. xxi.

BIBLIOGRAFÍA VIRTUAL

- http://es.wikipedia.org/wiki/Sistema_informatico
- http://galeon.hispavista.com/zaboot/analisiscc.html
- http://www.monografias.com/trabajos11/admicomp/admicomp.shtml
- http://www.monografias.com/trabajos11/cenco/cenco.shtml
- http://www.monografias.com/trabajos16/sistemas-distribuidos/sistemas-distribuidos.shtml
- http://www.desarrollaconmsdn.com/msdn/Cursos/Curso_Introduccion_a_.NET_con_CSharp/index.html

GLOSARIO DE TÉRMINOS

- **Administración:** Manera como se dirige o administra determinados recursos o bienes.

- **Administrador**: La persona que supervisa y controla el correcto funcionamiento de un sistema informático.

- **C#:** (leído en inglés "C Sharp" y en español "C Almohadilla") es el nuevo lenguaje de propósito general diseñado por Microsoft para su plataforma .NET. Sus principales creadores son Scott Wiltamuth y Anders Hejlsberg, éste último también conocido por haber sido el diseñador del lenguaje Turbo Pascal y la herramienta RAD Delphi.

- **Control:** El proceso para determinar lo que se está llevando a cabo, valorización y, si es necesario, aplicando medidas correctivas, de manera que la ejecución se desarrolle de acuerdo con lo planeado.

- **Infragistics:** Es un complemento de VisualStudio 2010, este complemento sirve para una rápida creación de interfaces de usuario, mismos que son amigables y de fácil uso para el usuario.

- **Internet:** Red de ordenadores a nivel mundial.

- **Monitoreo:** Es el proceso de recoger información sobre todos los aspectos de una red, es seguir sistemáticamente las variables y procesos claves en un periodo de tiempo y espacio y ver cómo cambian.

- **Monitoreo de Redes:** Es el análisis detallado que surge a partir del estudio sobre la red supervisada y que nos da un conocimiento de su funcionamiento y en el caso de tener algún error dar acción inmediata a su restablecimiento.

- **Programa:** Un conjunto de órdenes para un ordenador. Cuando se trata de un programa ya terminado que se compra, se suele hablar de una Aplicación Informática. Los programas se deben escribir en un cierto lenguaje de programación. Los lenguajes de programación que se acercan más al lenguaje humano que al del ordenador reciben el nombre de "lenguajes de alto nivel" (como Pascal); los que se acercan más al ordenador son los de "bajo nivel" (como el ensamblador).

- **Red.-** Una red de computadoras es una interconexión de computadoras para compartir información, recursos y servicios. Esta interconexión puede ser a través de un enlace físico (alambrado) o inalámbrico.

- **Sockets:** Los sockets no son más que puntos o mecanismos de comunicación entre procesos que permiten que un proceso hable (emita o reciba información) con otro proceso incluso estando estos procesos en distintas máquinas.

- **Software:** Es un programa informático creado para ser implantado en un computador, con la finalidad de ayudar a efectuar alguna operación o proceso determinado que se realice en un campo determinado.

- **Visual Studio 2010:** Es un entorno de desarrollo integrado para sistemas operativos Windows, soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET.

ANEXO 1. CODIFICACIÓN

DNSHEADER.CS

```
using System.Net;
using System.Text;
using System;
using System.IO;
using System.Windows.Forms;
using System.Collections.Specialized;
using System.Collections;
using System.Collections.Generic;

namespace MJsniffer
{
  public class DNSHeader
  {
    //DNS header fields
    private ushort usIdentification;       //Sixteen bits for identification
    private ushort usFlags;                //Sixteen bits for DNS flags
    private ushort usTotalQuestions;       //Sixteen bits indicating the number of entries
                                           //in the questions list
    private ushort usTotalAnswerRRs;       //Sixteen bits indicating the number of entries
                                           //entries in the answer resource record list
    private ushort usTotalAuthorityRRs;    //Sixteen bits indicating the number of entries
                                           //entries in the authority resource record list
    private ushort usTotalAdditionalRRs;   //Sixteen bits indicating the number of entries
                                           //entries in the additional resource record list
    //End DNS header fields

    public DNSHeader(byte []byBuffer, int nReceived)
    {
      MemoryStream memoryStream = new MemoryStream(byBuffer, 0, nReceived);
      BinaryReader binaryReader = new BinaryReader(memoryStream);

      //First sixteen bits are for identification
      usIdentification = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

      //Next sixteen contain the flags
      usFlags = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

      //Read the total numbers of questions in the quesion list
      usTotalQuestions = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

      //Read the total number of answers in the answer list
      usTotalAnswerRRs = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

      //Read the total number of entries in the authority list
      usTotalAuthorityRRs = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

      //Total number of entries in the additional resource record list
      usTotalAdditionalRRs = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());
    }

    public string Identification
    {
      get
      {
        return string.Format("0x{0:x2}", usIdentification);
```

```
        }
    }

    public string Flags
    {
        get
        {
            return string.Format("0x{0:x2}", usFlags);
        }
    }

    public string TotalQuestions
    {
        get
        {
            return usTotalQuestions.ToString();
        }
    }

    public string TotalAnswerRRs
    {
        get
        {
            return usTotalAnswerRRs.ToString();
        }
    }

    public string TotalAuthorityRRs
    {
        get
        {
            return usTotalAuthorityRRs.ToString();
        }
    }

    public string TotalAdditionalRRs
    {
        get
        {
            return usTotalAdditionalRRs.ToString();
        }
    }
        }
}
```

IPHEADER.CS

```
using System.Net;
using System.Text;
using System;
using System.IO;
using System.Windows.Forms;

namespace MJsniffer
{
    public class IPHeader
    {
        //IP Header fields
        private byte    byVersionAndHeaderLength;   //Eight bits for version and header length
        private byte    byDifferentiatedServices;   //Eight bits for differentiated services (TOS)
```

```
    private ushort   usTotalLength;          //Sixteen bits for total length of the datagram (header +
message)
    private ushort   usIdentification;       //Sixteen bits for identification
    private ushort   usFlagsAndOffset;       //Eight bits for flags and fragmentation offset
    private byte     byTTL;                  //Eight bits for TTL (Time To Live)
    private byte     byProtocol;             //Eight bits for the underlying protocol
    private short    sChecksum;              //Sixteen bits containing the checksum of the header
                                             //(checksum can be negative so taken as short)
    private uint     uiSourceIPAddress;      //Thirty two bit source IP Address
    private uint     uiDestinationIPAddress; //Thirty two bit destination IP Address
    //End IP Header fields

    private byte     byHeaderLength;         //Header length
    private byte[]   byIPData = new byte[4096]; //Data carried by the datagram


    public IPHeader(byte[] byBuffer, int nReceived)
    {

      try
      {
        //Create MemoryStream out of the received bytes
        MemoryStream memoryStream = new MemoryStream(byBuffer, 0, nReceived);
        //Next we create a BinaryReader out of the MemoryStream
        BinaryReader binaryReader = new BinaryReader(memoryStream);

        //The first eight bits of the IP header contain the version and
        //header length so we read them
        byVersionAndHeaderLength = binaryReader.ReadByte();

        //The next eight bits contain the Differentiated services
        byDifferentiatedServices = binaryReader.ReadByte();

        //Next eight bits hold the total length of the datagram
        usTotalLength = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

        //Next sixteen have the identification bytes
        usIdentification = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

        //Next sixteen bits contain the flags and fragmentation offset
        usFlagsAndOffset = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

        //Next eight bits have the TTL value
        byTTL = binaryReader.ReadByte();

        //Next eight represnts the protocol encapsulated in the datagram
        byProtocol = binaryReader.ReadByte();

        //Next sixteen bits contain the checksum of the header
        sChecksum = IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

        //Next thirty two bits have the source IP address
        uiSourceIPAddress = (uint)(binaryReader.ReadInt32());

        //Next thirty two hold the destination IP address
        uiDestinationIPAddress = (uint)(binaryReader.ReadInt32());

        //Now we calculate the header length

        byHeaderLength = byVersionAndHeaderLength;
        //The last four bits of the version and header length field contain the
```

```
    //header length, we perform some simple binary airthmatic operations to
    //extract them
    byHeaderLength <<= 4;
    byHeaderLength >>= 4;
    //Multiply by four to get the exact header length
    byHeaderLength *= 4;

    //Copy the data carried by the data gram into another array so that
    //according to the protocol being carried in the IP datagram
    Array.Copy(byBuffer,
            byHeaderLength,  //start copying from the end of the header
            byIPData, 0,
            usTotalLength - byHeaderLength);
  }
catch (Exception ex)
{
  MessageBox.Show(ex.Message, "MJsniffer", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
}
}

public string Version
{
  get
  {
    //Calculate the IP version

    //The four bits of the IP header contain the IP version
    if ((byVersionAndHeaderLength >> 4) == 4)
    {
      return "IP v4";
    }
    else if ((byVersionAndHeaderLength >> 4) == 6)
    {
      return "IP v6";
    }
    else
    {
      return "Unknown";
    }
  }
}

public string HeaderLength
{
  get
  {
    return byHeaderLength.ToString();
  }
}

public ushort MessageLength
{
  get
  {
    //MessageLength = Total length of the datagram - Header length
    return (ushort)(usTotalLength - byHeaderLength);
  }
}

public string DifferentiatedServices
```

```
{
  get
  {
    //Returns the differentiated services in hexadecimal format
    return string.Format ("0x{0:x2} ({1})", byDifferentiatedServices,
      byDifferentiatedServices);
  }
}

public string Flags
{
  get
  {
    //The first three bits of the flags and fragmentation field
    //represent the flags (which indicate whether the data is
    //fragmented or not)
    int nFlags = usFlagsAndOffset >> 13;
    if (nFlags == 2)
    {
      return "Don't fragment";
    }
    else if (nFlags == 1)
    {
      return "More fragments to come";
    }
    else
    {
      return nFlags.ToString();
    }
  }
}

public string FragmentationOffset
{
  get
  {
    //The last thirteen bits of the flags and fragmentation field
    //contain the fragmentation offset
    int nOffset = usFlagsAndOffset << 3;
    nOffset >>= 3;

    return nOffset.ToString();
  }
}

public string TTL
{
  get
  {
    return byTTL.ToString();
  }
}

public Protocol ProtocolType
{
  get
  {
    //The protocol field represents the protocol in the data portion
    //of the datagram
    if (byProtocol == 6)      //A value of six represents the TCP protocol
    {
```

```
            return Protocol.TCP;
        }
        else if (byProtocol == 17)  //Seventeen for UDP
        {
            return Protocol.UDP;
        }
        else
        {
            return Protocol.Unknown;
        }
    }
}

public string Checksum
{
    get
    {
        //Returns the checksum in hexadecimal format
        return string.Format ("0x{0:x2}", sChecksum);
    }
}

public IPAddress SourceAddress
{
    get
    {
        return new IPAddress(uiSourceIPAddress);
    }
}

public IPAddress DestinationAddress
{
    get
    {
        return new IPAddress(uiDestinationIPAddress);
    }
}

public string TotalLength
{
    get
    {
        return usTotalLength.ToString();
    }
}

public string Identification
{
    get
    {
        return usIdentification.ToString();
    }
}

public byte[] Data
{
    get
    {
        return byIPData;
    }
}
```

```
    }
}
```

TCPHEADER.CS

```
using System.Net;
using System.Text;
using System;
using System.IO;
using System.Windows.Forms;

namespace MJsniffer
{
   public class TCPHeader
   {
      //TCP header fields
      private ushort usSourcePort;            //Sixteen bits for the source port number
      private ushort usDestinationPort;       //Sixteen bits for the destination port number
      private uint   uiSequenceNumber=555;        //Thirty two bits for the sequence number
      private uint   uiAcknowledgementNumber=555;  //Thirty two bits for the acknowledgement
number
      private ushort usDataOffsetAndFlags=555;     //Sixteen bits for flags and data offset
      private ushort usWindow=555;                //Sixteen bits for the window size
      private short  sChecksum=555;               //Sixteen bits for the checksum
                                    //(checksum can be negative so taken as short)
      private ushort usUrgentPointer;         //Sixteen bits for the urgent pointer
      //End TCP header fields

      private byte   byHeaderLength;          //Header length
      private ushort usMessageLength;         //Length of the data being carried
      private byte[] byTCPData = new byte[4096];//Data carried by the TCP packet

      public TCPHeader(byte [] byBuffer, int nReceived)
      {
         try
         {
            MemoryStream memoryStream = new MemoryStream(byBuffer, 0, nReceived);
            BinaryReader binaryReader = new BinaryReader(memoryStream);

            //The first sixteen bits contain the source port
            usSourcePort = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16 ());

            //The next sixteen contain the destiination port
            usDestinationPort = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16 ());

            //Next thirty two have the sequence number
            uiSequenceNumber = (uint)IPAddress.NetworkToHostOrder(binaryReader.ReadInt32());

            //Next thirty two have the acknowledgement number
            uiAcknowledgementNumber =
(uint)IPAddress.NetworkToHostOrder(binaryReader.ReadInt32());

            //The next sixteen bits hold the flags and the data offset
            usDataOffsetAndFlags =
(ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

            //The next sixteen contain the window size
            usWindow = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

            //In the next sixteen we have the checksum
            sChecksum = (short)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());
```

```
            //The following sixteen contain the urgent pointer
            usUrgentPointer = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

            //The data offset indicates where the data begins, so using it we
            //calculate the header length
            byHeaderLength = (byte)(usDataOffsetAndFlags >> 12);
            byHeaderLength *= 4;

            //Message length = Total length of the TCP packet - Header length
            usMessageLength = (ushort)(nReceived - byHeaderLength);

            //Copy the TCP data into the data buffer
            Array.Copy(byBuffer, byHeaderLength, byTCPData, 0, nReceived - byHeaderLength);
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message, "Sniffer" + (nReceived), MessageBoxButtons.OK,
MessageBoxIcon.Error);
        }
    }

    public string SourcePort
    {
        get
        {
            return usSourcePort.ToString();
        }
    }

    public string DestinationPort
    {
        get
        {
            return usDestinationPort.ToString ();
        }
    }

    public string SequenceNumber
    {
        get
        {
            return uiSequenceNumber.ToString();
        }
    }

    public string AcknowledgementNumber
    {
        get
        {
            //If the ACK flag is set then only we have a valid value in
            //the acknowlegement field, so check for it beore returning
            //anything
            if ((usDataOffsetAndFlags & 0x10) != 0)
            {
                return uiAcknowledgementNumber.ToString();
            }
            else
                return "";
        }
    }
```

```
public string HeaderLength
{
   get
   {
      return byHeaderLength.ToString();
   }
}

public string WindowSize
{
   get
   {
      return usWindow.ToString();
   }
}

public string UrgentPointer
{
   get
   {
      //If the URG flag is set then only we have a valid value in
      //the urgent pointer field, so check for it beore returning
      //anything
      if ((usDataOffsetAndFlags & 0x20) != 0)
      {
         return usUrgentPointer.ToString();
      }
      else
         return "";
   }
}

public string Flags
{
   get
   {
      //The last six bits of the data offset and flags contain the
      //control bits

      //First we extract the flags
      int nFlags = usDataOffsetAndFlags & 0x3F;

      string strFlags = string.Format ("0x{0:x2} (", nFlags);

      //Now we start looking whether individual bits are set or not
      if ((nFlags & 0x01) != 0)
      {
         strFlags += "FIN, ";
      }
      if ((nFlags & 0x02) != 0)
      {
         strFlags += "SYN, ";
      }
      if ((nFlags & 0x04) != 0)
      {
         strFlags += "RST, ";
      }
      if ((nFlags & 0x08) != 0)
      {
         strFlags += "PSH, ";
```

```
                }
                if ((nFlags & 0x10) != 0)
                {
                    strFlags += "ACK, ";
                }
                if ((nFlags & 0x20) != 0)
                {
                    strFlags += "URG";
                }
                strFlags += ")";

                if (strFlags.Contains("()"))
                {
                    strFlags = strFlags.Remove(strFlags.Length - 3);
                }
                else if (strFlags.Contains(", )"))
                {
                    strFlags = strFlags.Remove(strFlags.Length - 3, 2);
                }

                return strFlags;
            }
        }

        public string Checksum
        {
            get
            {
                //Return the checksum in hexadecimal format
                return string.Format("0x{0:x2}", sChecksum);
            }
        }

        public byte[] Data
        {
            get
            {
                return byTCPData;
            }
        }

        public ushort MessageLength
        {
            get
            {
                return usMessageLength;
            }
        }
    }
}
```

UDPHEADER.CS

```
using System.Net;
using System.Text;
using System;
using System.IO;
using System.Windows.Forms;

namespace MJsniffer
{
  public class UDPHeader
```

```
{
  //UDP header fields
  private ushort usSourcePort;          //Sixteen bits for the source port number
  private ushort usDestinationPort;     //Sixteen bits for the destination port number
  private ushort usLength;              //Length of the UDP header
  private short sChecksum;              //Sixteen bits for the checksum
                                //(checksum can be negative so taken as short)
  //End UDP header fields

  private byte[] byUDPData = new byte[4096];  //Data carried by the UDP packet

  public UDPHeader(byte [] byBuffer, int nReceived)
  {
    MemoryStream memoryStream = new MemoryStream(byBuffer, 0, nReceived);
    BinaryReader binaryReader = new BinaryReader(memoryStream);

    //The first sixteen bits contain the source port
    usSourcePort = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

    //The next sixteen bits contain the destination port
    usDestinationPort = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

    //The next sixteen bits contain the length of the UDP packet
    usLength = (ushort)IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

    //The next sixteen bits contain the checksum
    sChecksum = IPAddress.NetworkToHostOrder(binaryReader.ReadInt16());

    //Copy the data carried by the UDP packet into the data buffer
    Array.Copy(byBuffer,
          8,          //The UDP header is of 8 bytes so we start copying after it
          byUDPData,
          0,
          nReceived - 8);
  }

  public string SourcePort
  {
    get
    {
      return usSourcePort.ToString();
    }
  }

  public string DestinationPort
  {
    get
    {
      return usDestinationPort.ToString();
    }
  }

  public string Length
  {
    get
    {
      return usLength.ToString ();
    }
  }

  public string Checksum
```

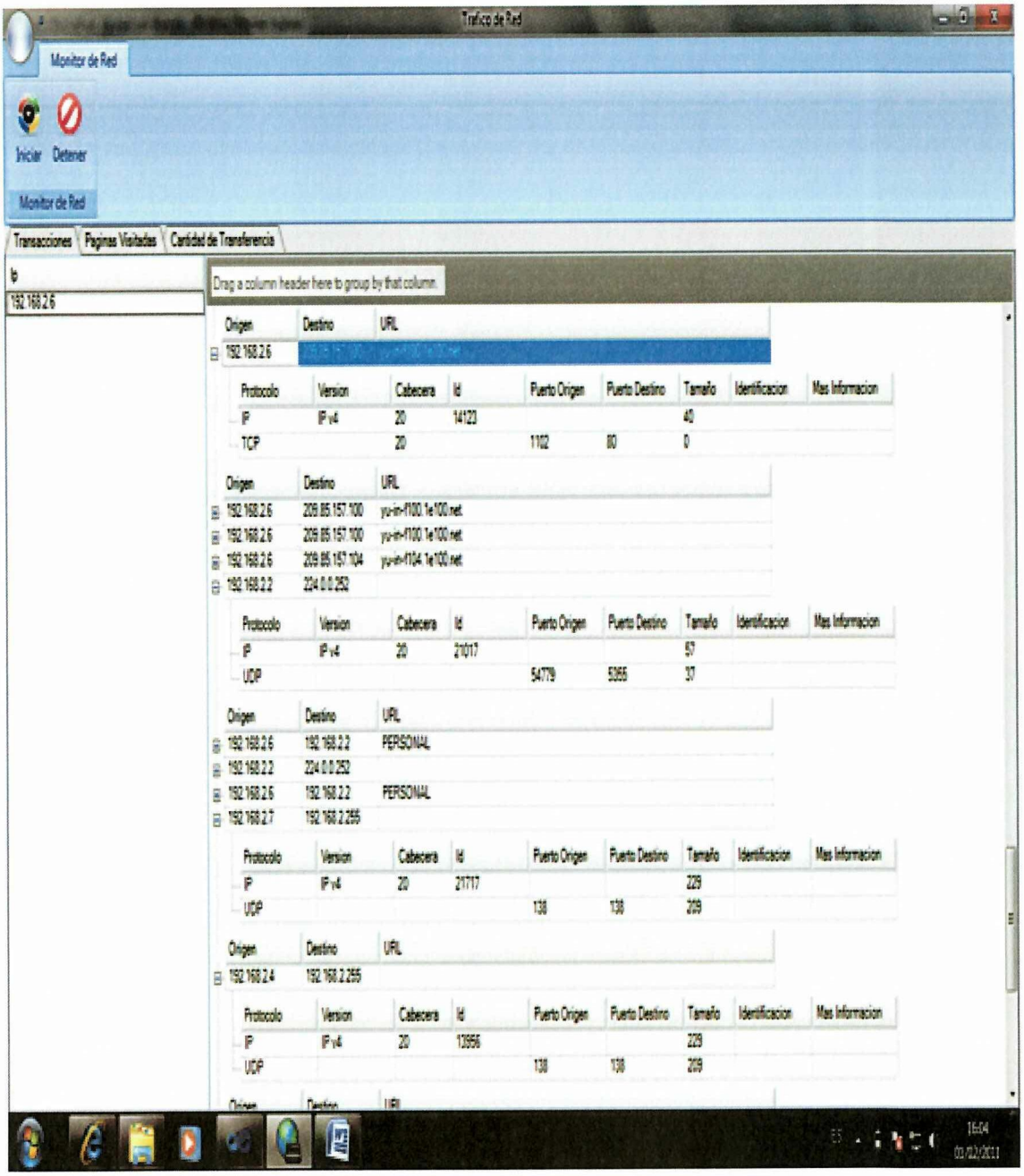
```
        {
            get
            {
                //Return the checksum in hexadecimal format
                return string.Format("0x{0:x2}", sChecksum);
            }
        }

        public byte[] Data
        {
            get
            {
                return byUDPData;
            }
        }
    }
}
```

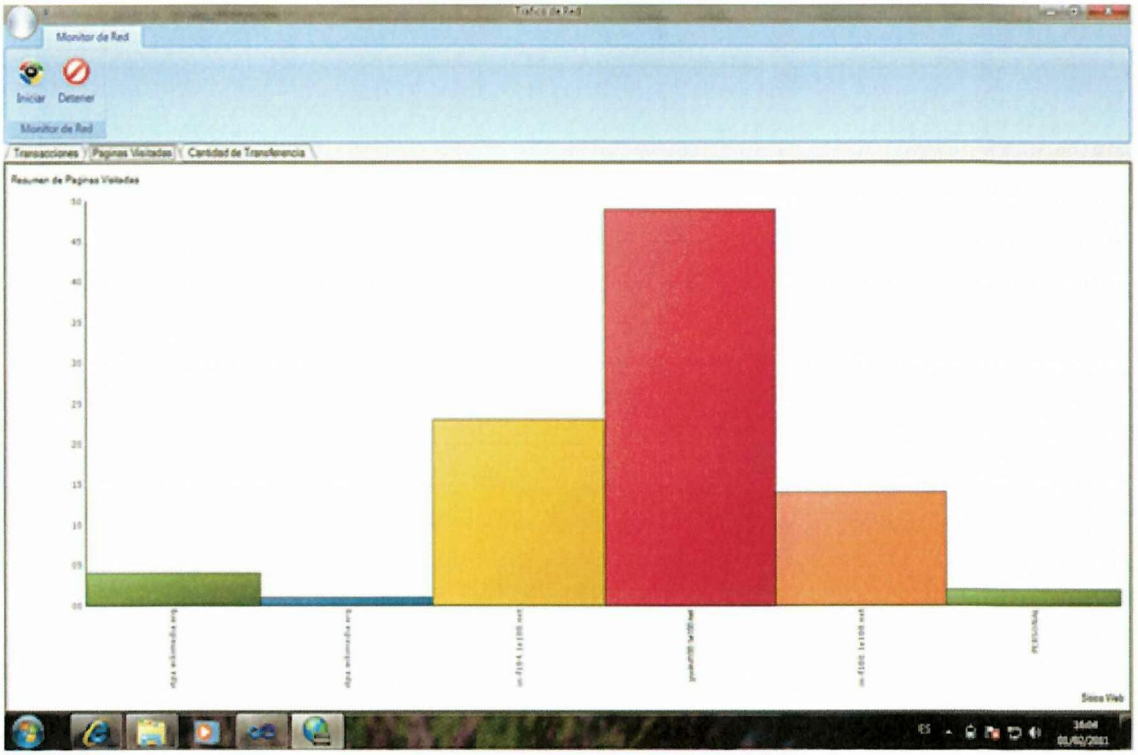
ANEXO 2. PRUEBAS EXTERNAS

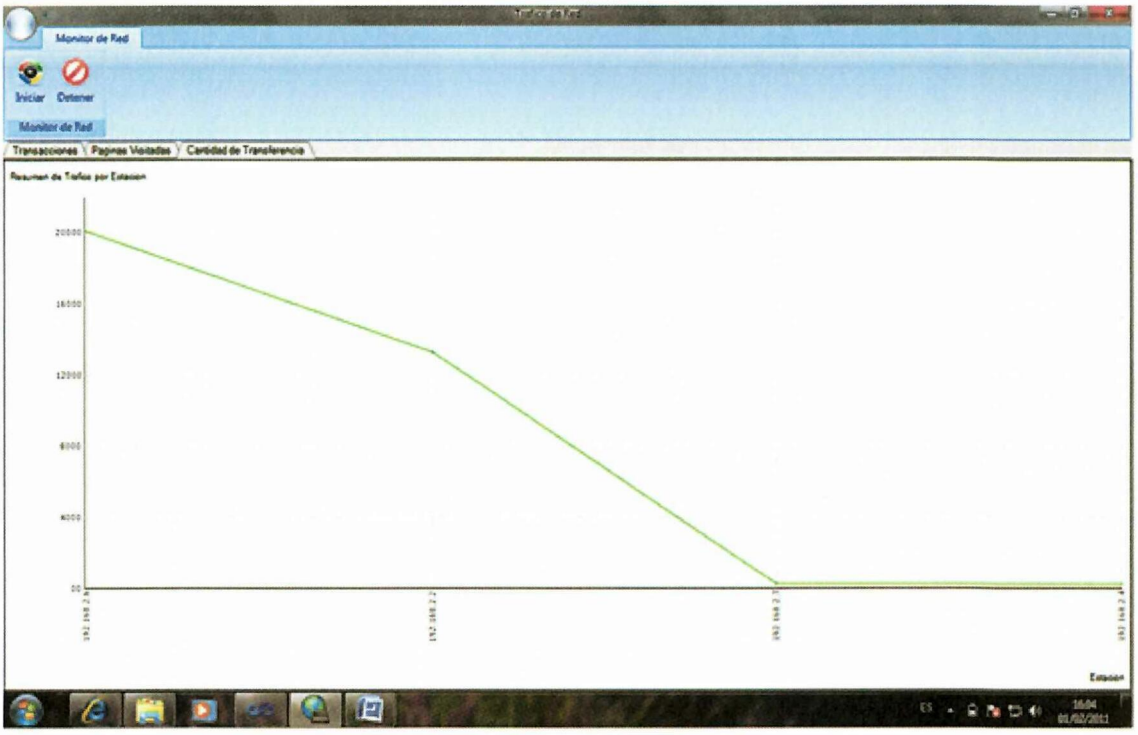
C.Cómputo 1

Propietario : Sr. Paredes

Dirección : Calle Juan Abel Echeverría e Isla Isabela



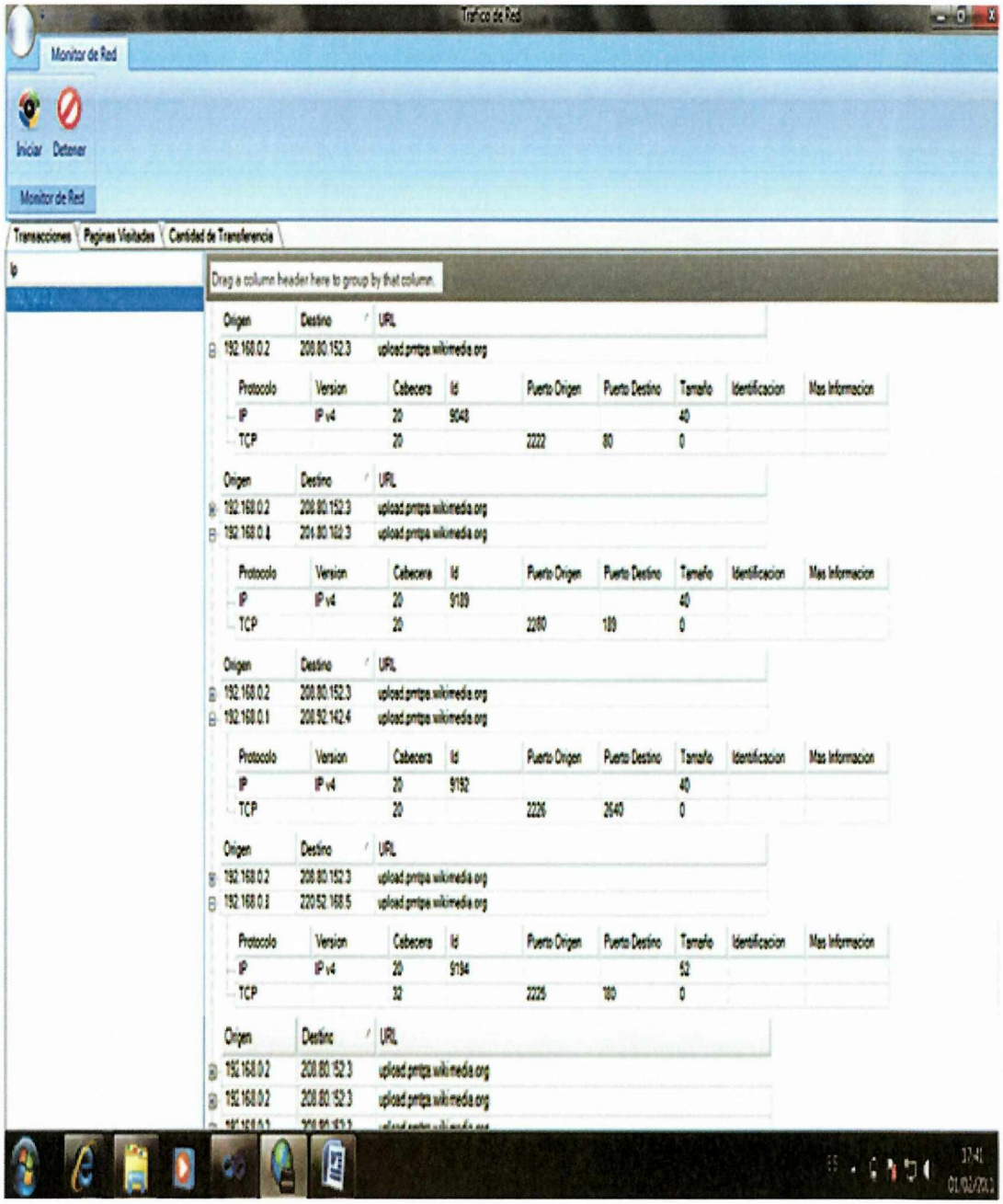

Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia
Resumen de Paginas Visitadas

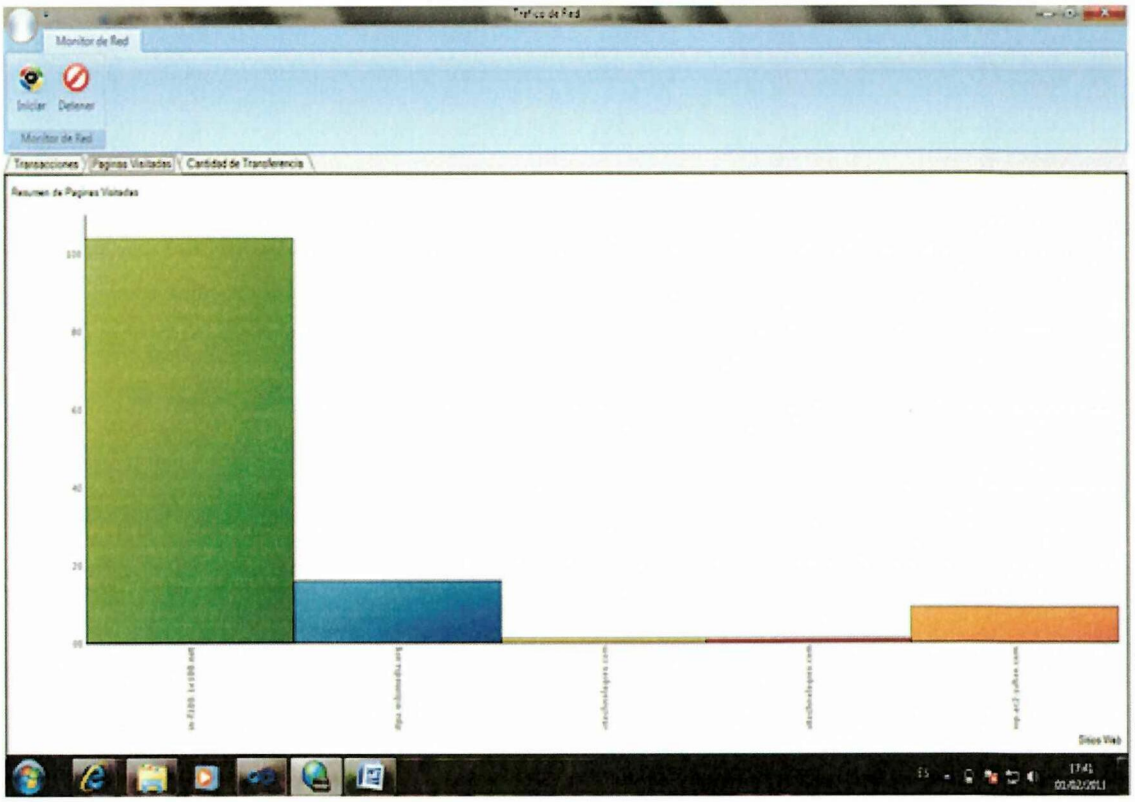

Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia
Resumen de Trafico por Estacion
20000
16000
12000
8000
4000
0

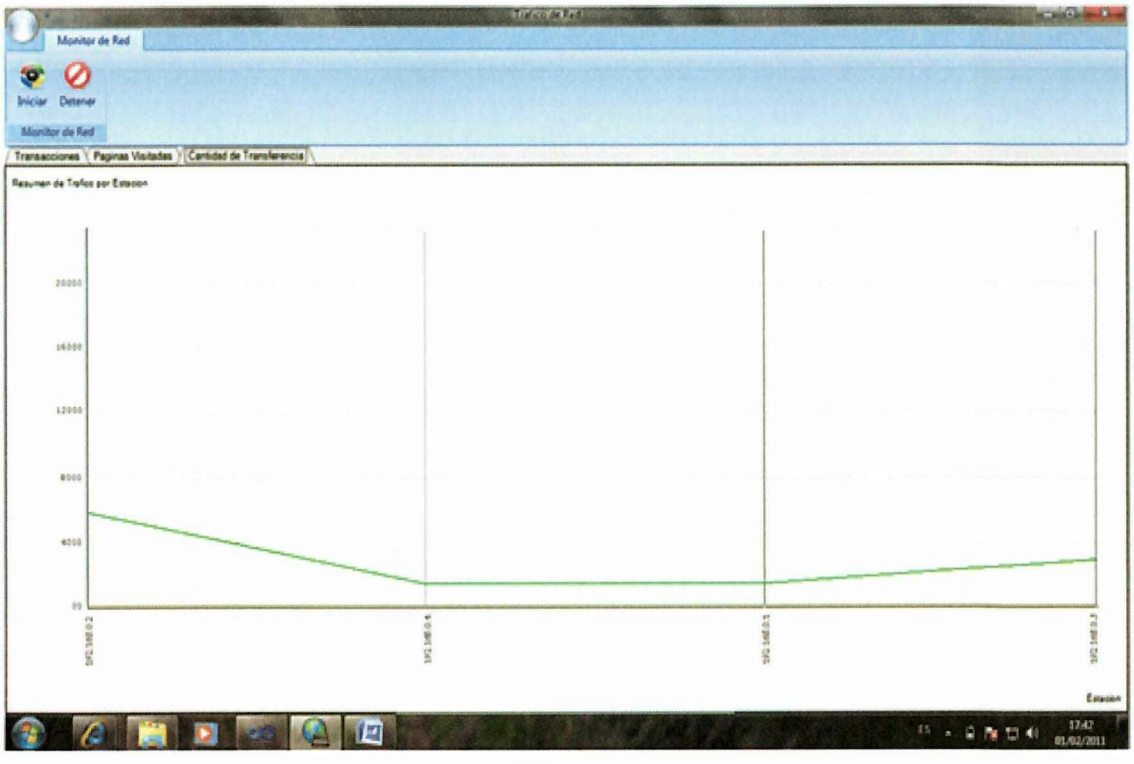
C. Cómputo 2

Propietario : Sr. Santiago Terán

Dirección : Calle Pangua



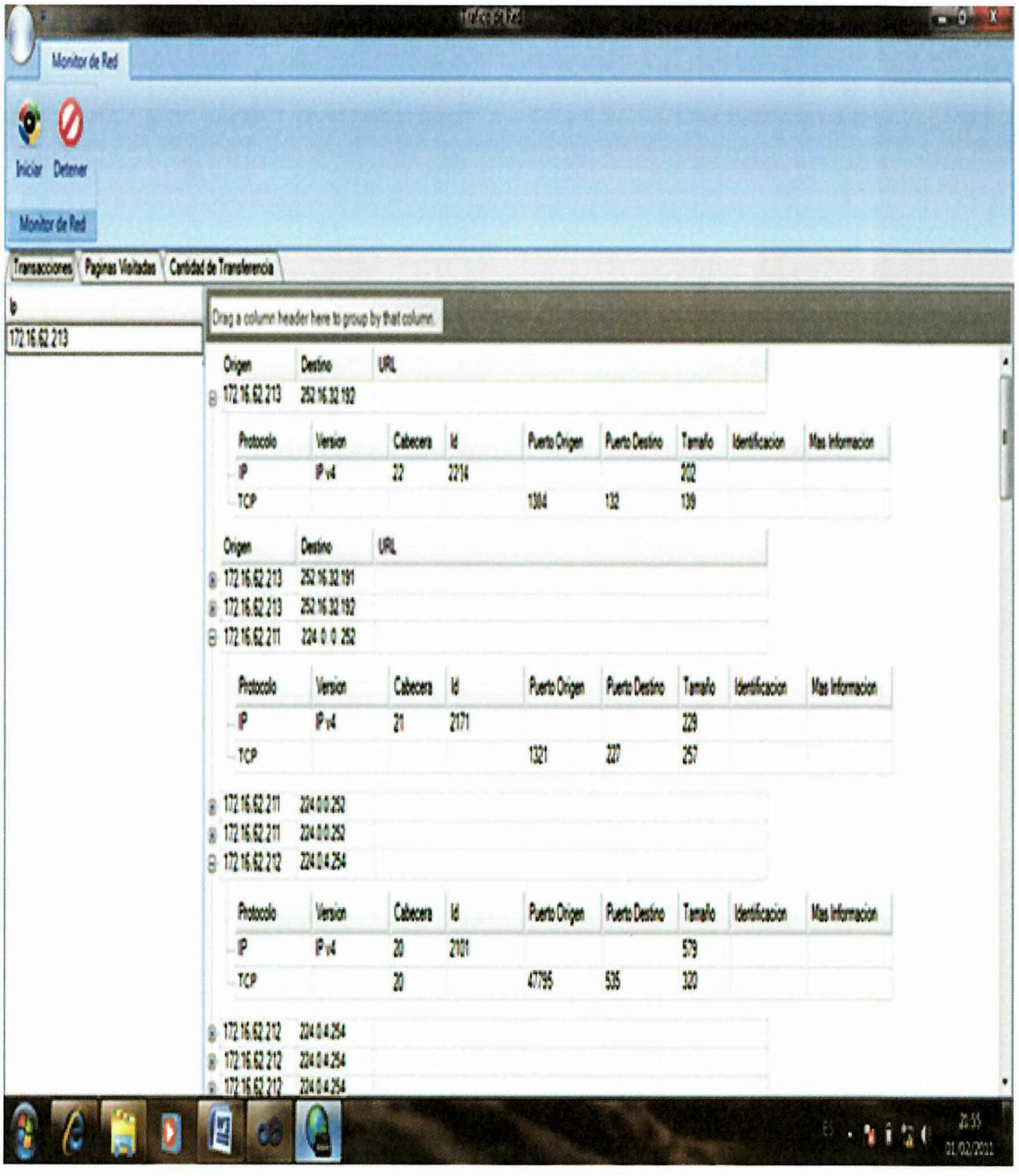

Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia
Resumen de Paginas Visitadas

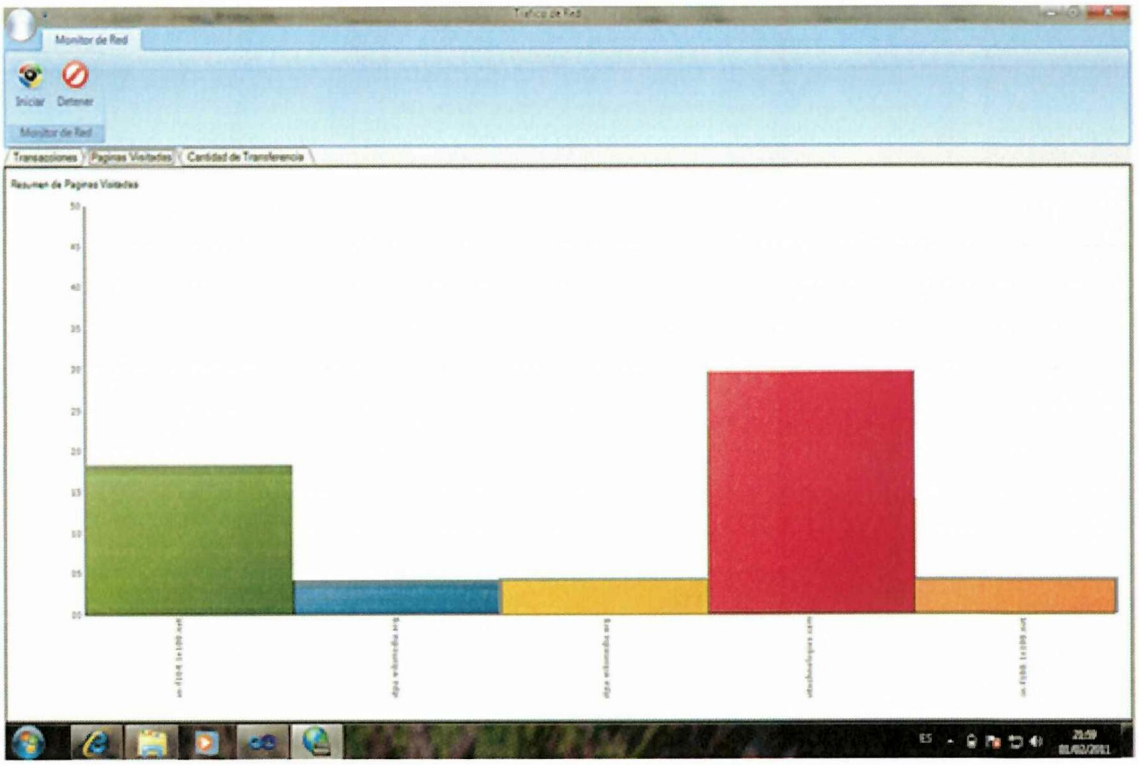

Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia
Resumen de Trafico por Estacion

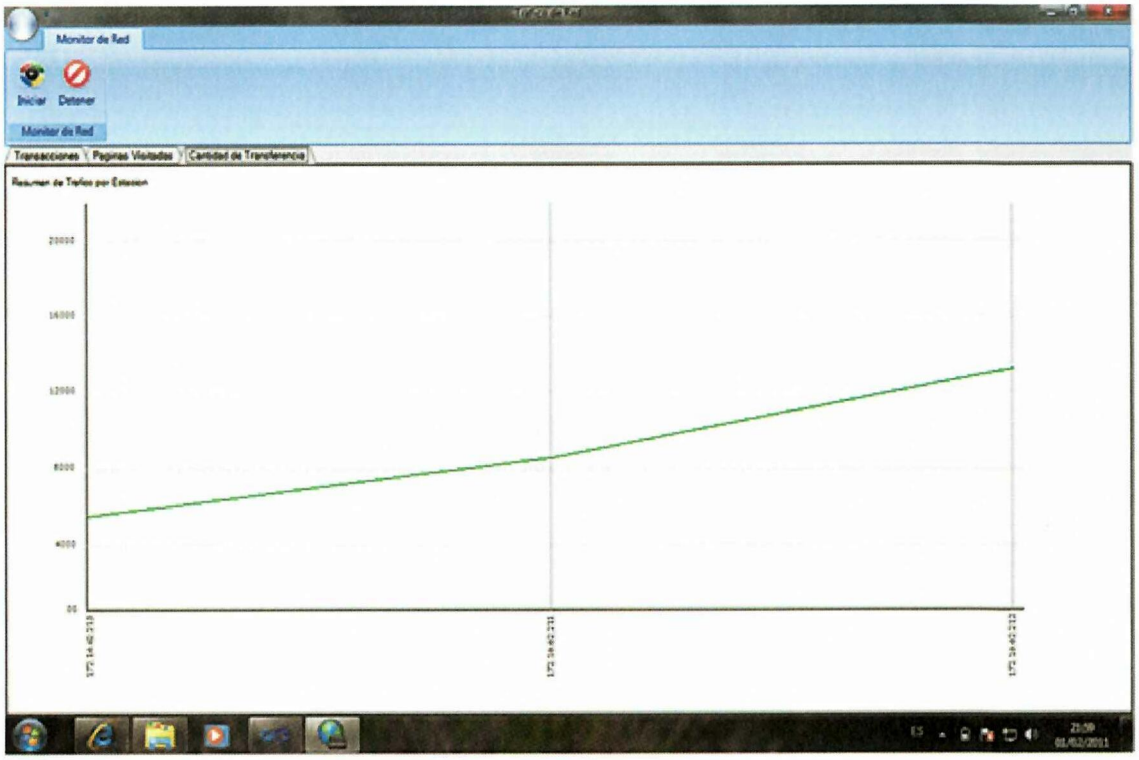
C.Cómputo 3

Propietario : Sr. Daniel Chicaiza

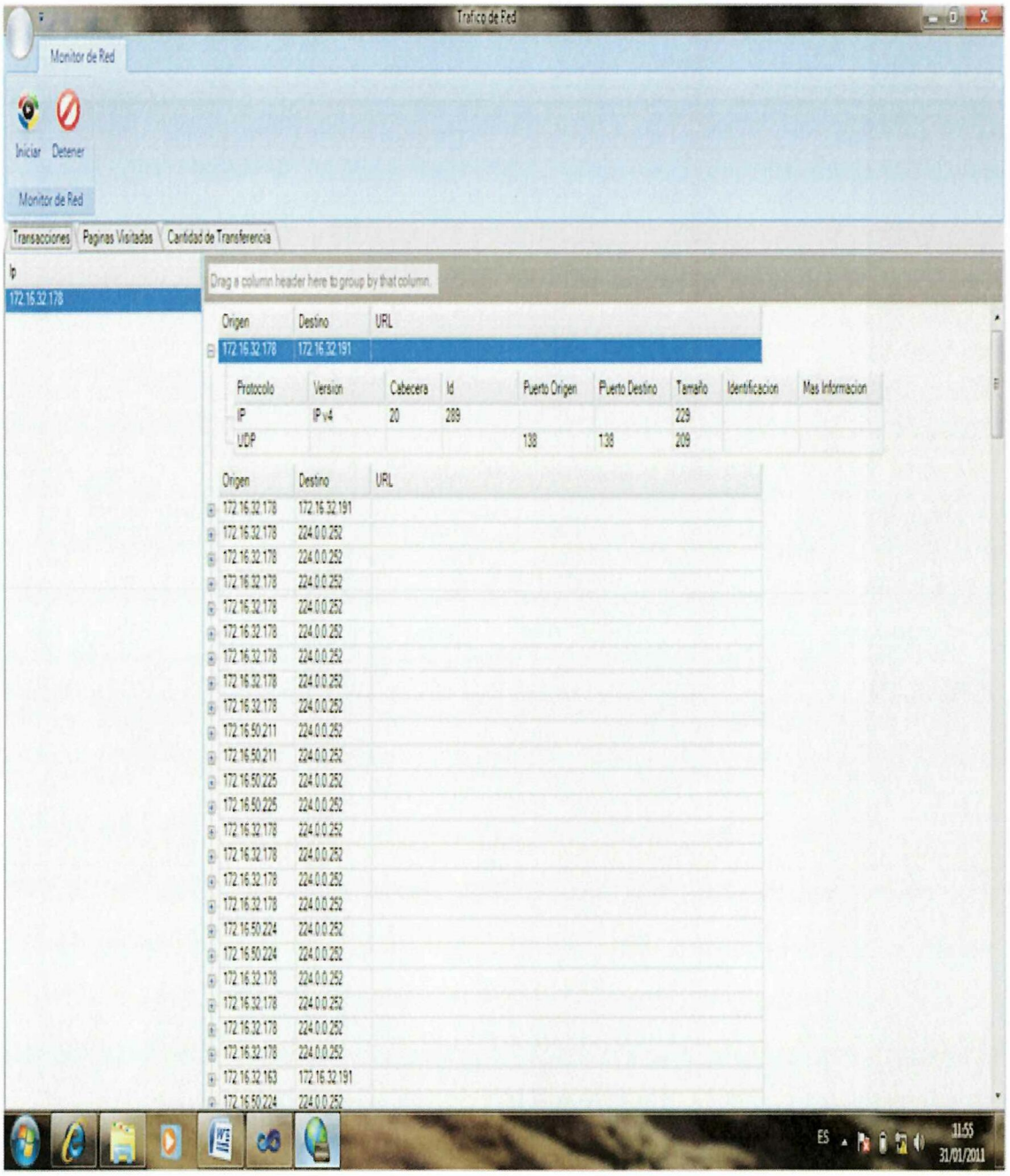
Dirección : La Cocha

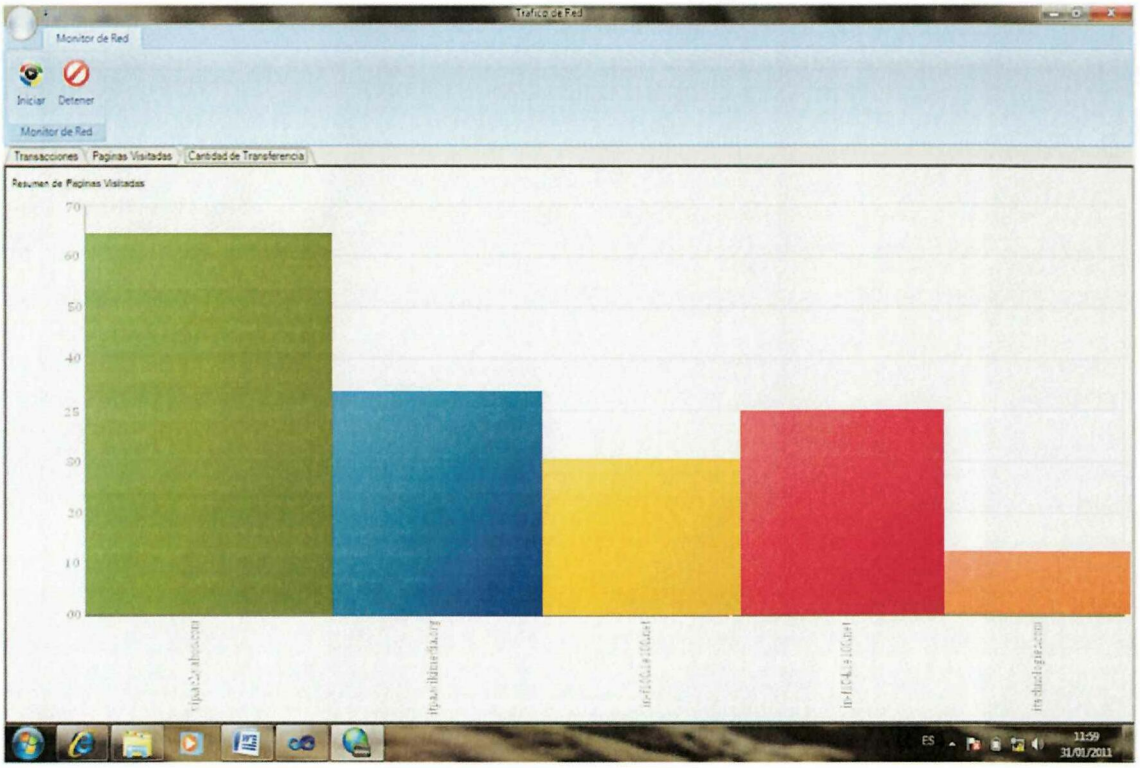


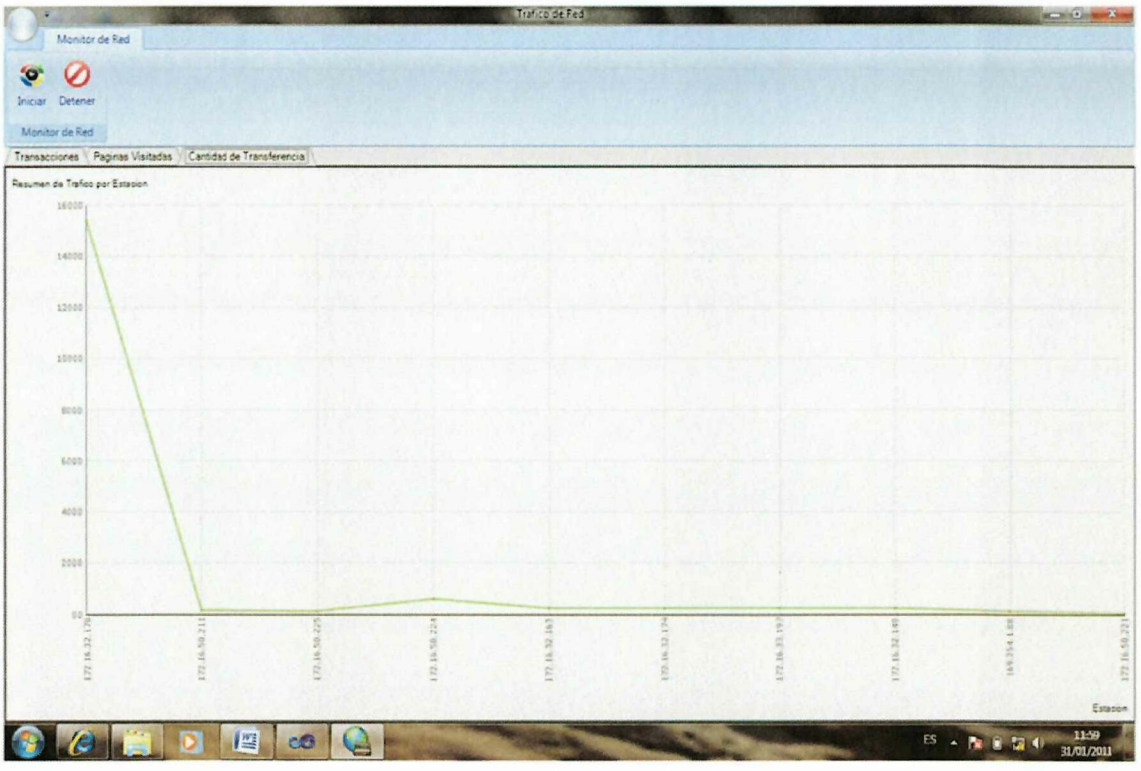

Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia
Resumen de Paginas Visitadas


Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia

ANEXO3. PRUEBAS LAB 5 / UTC




Trafico de Red
Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia
Resumen de Paginas Visitadas
70
60
50
40
30
20
10
00
ES
11:59
31/01/2011


Trafico de Red
Monitor de Red
Iniciar
Detener
Monitor de Red
Transacciones
Paginas Visitadas
Cantidad de Transferencia
Resumen de Trafico por Estacion
16000
14000
12000
10000
8000
6000
4000
2000
00
Estacion
ES
11:59
31/01/2011