UNIVERSIDAD DE QUINTANA ROO

División de Ciencias e Ingeniería

Diseño de una arquitectura para la estimación de ancho de banda en ambientes controlados

Tesis para obtener el grado de

Ingeniero en Redes

Presenta

Luis Jesús Oliva Canché

Director de tesis

M.T.I. Vladimir Veniamin Cabañas Victoria

Supervisores

Ing. Rubén Enrique González Elixavide
M.C. Javier Vázquez Castillo

Chetumal, Quintana Roo, México. Noviembre, 2009

UNIVERSIDAD DE QUINTANA ROO

División de Ciencias e Ingeniería

Trabajo de tesis elaborado bajo supervision del Comité de Asesoría y aprobada como requisito parcial para obtener el grado de:

Ingeniero en Redes

Comité de Trabajo Tesis

Director:

M.T.I. Vladimir Veniamin Cabañas Victoria

Supervisor:

Ing. Rubén Enrique González Elixavide

Supervisor:

M.C. Javier Vázquez Castillo

Chetumal, Quintana Roo, México, Noviembre de 2009..

AGRADECIMIENTOS

A mis padres, hermanos y a Dios por su apoyo y amor, que me han proporcionado la fuerza y voluntad para continuar en los momentos más difíciles.

A mi Director de Tésis,
M.T.I. Vladimir Veniamin Cabañas Victoria
por su asesoramiento, paciencia y compromiso
para guiarme en este trabajo.

A todos mis amigos y compañeros de la carrera de Ingeniería en Redes ya que permitieron que este pequeño viaje sea interesante y ameno.

DEDICATORIA

Dedico este trabajo a la dama de los senderos infinitos por su enorme coraje, fuerza y amor, que me han permitido levantarme en incontables ocasiones. A mí antiguo y supremo maestro de la sabiduría; que pasen aún muchos años antes de reunirnos…

RESUMEN

En la actualidad la convergencia de los servicios de voz, datos y video se ha incrementado sustancialmente, esto se ha reflejado principalmente en el uso de Internet como la máxima plataforma de comunicaciones y al aumento y demanda del número de usuarios de este servicios. Uno de los factores claves en algunas tecnologías de red que emplean sistemas de voz, multimedia y/o aplicaciones en tiempo real, es el ancho de banda y la calidad de servicio brindado.

La calidad del servicio puede verse afectada en un tiempo determinado por diversos factores: alta demanda en el intercambio de información, calidad de los dispositivos de comunicaciones, tecnologías de comunicación, protocolos, aplicaciones, etc. Con una estimación adecuada del ancho de banda, los proveedores de estos servicios pueden planear estrategias para mejorar las condiciones y los factores que influyen en la calidad de sus redes.

A través del diseño e implementación de una arquitectura de red para simular un “ambiente controlado” en el cual se puede observar el desempeño de los equipos de comunicaciones, se utilizarán pruebas de medición de ancho de banda con herramientas de software libre y determinar si pueden ser llevarse acabo en la red interna de la Universidad de Quintana Roo.

El ambiente de pruebas consta del diseño de una arquitectura de red, la cual puede ser asociada a los dispositivos de red internos de la Universidad, una vez diseñada e implementada dicha arquitectura, se genera tráfico con la finalidad de establecer una referencia del porcentaje de utilización del canal de la red. De esta manera se puede saber si las herramientas de medición de ancho de banda y capacidad del canal realizan mediciones confiables y se comportan conforme a lo esperado.

CONTENIDO

Índice de figuras

Índice de tablas

Introducción al trabajo

La Universidad de Quintana Roo cuenta con una red de paquetes interna que soporta diversas aplicaciones como voz, datos y video. El ancho de banda determinado por los diversos componentes y medios de transmisión puede parecer suficiente, sin embargo, la transmisión de paquetes que implica salida hacia el Internet presenta un rendimiento bajo causando la insatisfacción del usuario.

Algunos de los problemas que han impedido determinar de forma precisa las razones del bajo rendimiento que posee el Internet en la actual red de la Universidad de Quintana Roo son: la falta de métricas que permitan determinar las causas que afectan al rendimiento del ancho de banda, la inexistencia de un mapa de todos los dispositivos o equipos conectados a Internet, la carencia de herramientas que midan el ancho de banda de extremo a extremo de todos y cada uno de los equipos conectados a la red y el grado o cantidad de ancho de banda utilizado por cada uno de los dispositivos en un momento determinado; no sólo de algunos puertos, nodos o segmentos en la red como se hace actualmente.

La importancia del presente trabajo radica en investigar y experimentar con las herramientas desarrolladas, en algunos casos, por investigadores de diversas universidades o instituciones de investigación que siguen una filosofía abierta para su implementación; aplicando métricas y ejecutando aplicaciones que nos permitan caracterizar la actual red, con el propósito de lograr determinar cuáles podrían ser los factores que afectan el rendimiento de la red; como paquetes broadcast, equipos dañados ó mal configurados que generan tráfico adicional,

protocolos activos que no se utilizan, grandes dominios de broadcast, horas pico, etc.

Se diseñó e implementó una arquitectura de simulación de tráfico de paquetes para diferentes protocolos en ambientes controlados. Dicha arquitectura permite observar el desempeño de herramientas de medición de ancho de banda. De esta manera se podrá determinar el grado de fiabilidad en esquemas de prueba controlados.

Posteriormente, en un futuro y a partir de las pruebas y resultados del presente trabajo, se podrán realizar investigaciones en la red interna de la Universidad de Quintana Roo para desarrollar estrategias que permitan, si no dar solución a estos problemas, mejorar la calidad del servicio de Internet que se ofrece actualmente.

Objetivos y justificación

Objetivo general

- Investigar, documentar e implementar una arquitectura de tráfico controlado que permita estudiar y poner a prueba, un conjunto de herramientas gratuitas, para la estimación del ancho de banda disponible en una red de paquetes. Esto permitirá en un futuro generar un conjunto de estrategias o políticas de acción que ayuden a mejorar la calidad del servicio de acceso a Internet en la red universitaria.

Objetivos particulares

- Estudiar las principales métricas y herramientas para estimación de ancho de banda que existen gratuitos.

- Investigar el funcionamiento de las técnicas de estimación del ancho de banda propuestas por diferentes investigadores.

- Implementar una arquitectura de tráfico controlado, con el objetivo de poner a prueba algunas de las herramientas disponibles para medición y estimación de ancho de banda.

- Construir un esquema de tráfico basado en sistemas y protocolos abiertos con el propósito de proveer interoperabilidad entre arquitecturas.

- Documentar los resultados y guías necesarias para la implementación de un ambiente de tráfico controlado para futuras investigaciones.
- Generar un concentrado de configuraciones, software y dispositivos necesarios para el desarrollo de la arquitectura en formato digital, que facilite futuras Implementaciones.

Justificación.

La importancia de ésta tesis radica en diseñar una arquitectura de tráfico controlado que permita investigar y experimentar diversos algoritmos de estimación de ancho de banda disponible. Dichos algoritmos son desarrollados por diversas universidades e investigadores siguiendo un esquema abierto; es decir, nos proporcionan los detalles de la implementación (código fuente). Lo anterior con la finalidad que en un futuro, nos permita caracterizar la actual red de la Universidad de Quintana Roo.

También provee una base para futuras investigaciones e implementaciones de ambientes controlados de tráfico para los estudiantes de la carrera de Ingeniería en Redes y en general para cualquier investigador o profesional relacionado con el área de las comunicaciones.

Marco teórico

Internet es el resultado de un proyecto académico-militar iniciado en E.U.A. en la década de los 60, impulsado principalmente por el Departamento de Defensa a través de la Agencia de Investigación de Proyectos Avanzados de Defensa (Defense Advanced Research Projects Agency), y desarrollado por algunas universidades, lo cierto es que han participado países y gobiernos que de manera conjunta impulsaron sus propias redes nacionales, para que años más tarde empezaran a interconectarse entre sí y formar lo que sería la red de redes.

Con el tiempo, la NSFnet (como se le conocía entonces) empezó a mostrar un crecimiento acelerado. Para 1984 había más de 10 mil computadoras interconectadas, y la NSF (National Science Fundation) tenía listo el backbone de Internet a 56 Kbps. Ocho años después, el backbone tenía una capacidad de 44.736 Mbps y ya eran más de un millón de equipos conectados en la red.

El crecimiento acelerado de la red y el desarrollo de ésta, se presentó con la generación de la "masa crítica", basada en la población académica de muchos países (no solamente EE.UU.). Este sistema y la creación de un sistema "amigable" conocido como el World Wide Web fue desarrollado por un inglés radicado en Suiza para la Organización Europea en Investigación Nuclear (CERN) y facilitó el acceso a los recursos de información esparcida en la red.[1]

El Centro Universitario de Cómputo (CUC), se fundó en septiembre de 1991, en las instalaciones de la biblioteca pública Javier Rojo Gómez, en la ciudad de Chetumal. Inició sus servicios con 20 computadoras personales para la implementación del primer curso de pre-requisito Universitario, nació como una

[1] (Garay, 2003)

unidad de servicios de apoyo académico cuya función principal era ofrecer al alumnado de la institución los conocimientos necesarios para el manejo y operación de los recursos computacionales, proporcionándole herramientas que le permitieran el mejor aprovechamiento de los conocimientos adquiridos.

En el mes de abril de 1993 se inauguraron las actuales instalaciones del Centro Universitario de Cómputo que constan de: un aula didáctica de cómputo, un área de servicios académicos, mantenimiento, soporte técnico y la jefatura del departamento de Cómputo.

Redes de telecomunicaciones

Una red de telecomunicaciones es un conjunto de medios técnicos instalados, organizados, operados y administrados con la finalidad de brindar servicios de comunicaciones a distancia. En particular, decimos que una red de computadoras, es una red de telecomunicaciones de datos que enlaza a dos o más equipos terminales o equipos finales de usuario.

Debe notarse que el concepto de red es independiente de la cantidad de enlaces que comprende. Para estudiar las redes, se pueden hacer distintos enfoques, según las características que se analizan, y cada una de éstas da lugar a uno o varios tipos de red específicos. Así, podemos estudiarlas según:

- Su carácter
- La naturaleza de los datos que transportan
- Su disponibilidad
- Su extensión o cobertura
- Su topología.[2]

[2] (UTN, 2008)

Todo este conjunto de equipos (computadoras y/o dispositivos) están conectados por medio de cables, señales, ondas o cualquier otro método de transporte de datos, comparten información (archivos), recursos (CD-ROM, impresoras, etc.) y servicios (acceso a Internet, e-mail, chat, juegos). Todo esto es lo que se conoce como redes informáticas, las cuales pueden estudiarse desde distintos enfoques, según las características que se analizan. Una red informática tiene como finalidad el brindar servicios de comunicaciones a distancia, ya que enlaza a dos o más equipos.[3]

Por extensión las redes pueden ser:

- Personal Area Network (PAN, Red de Área Personal)
- Local Area Network (LAN, Red de Área Loca)
- Metropolitan Area Network (MAN, Red de Área Metropolitana)
- Wide Area Network (WAN, Red de Área Extensa)

Como las redes informáticas pueden ser de gran magnitud, es importante poder representar gráficamente todos los elementos que la conforman. Un mapa de red es una representación gráfica de todas las computadoras y dispositivos en una red, que muestra como están conectados entre sí. Una ventaja de los mapas de red es que sirven para acceder rápidamente a cualquier elemento de la misma de forma rápida.

En conexiones a Internet el ancho de banda es la cantidad de información o de datos que se puede enviar a través de una conexión de red en un período de tiempo dado. El ancho de banda se indica generalmente en bits por segundo (bps), kilobits por segundo (Kbps), o megabits por segundo (Mbps), y a menudo

[3] (Groth & Skandier, 2005)

se utiliza como sinónimo para la tasa de transferencia de datos (la cantidad de datos que se puedan llevar de un punto a otro en un período dado, generalmente un segundo).[4]

En comunicaciones de la capa física de modelo de referencia OSI, el término ancho de banda se refiere a la anchura espectral de señales electromagnéticas o la propagación de las características de los sistemas de comunicación. En el contexto de las redes de datos, ancho de banda cuantifica la velocidad de transmisión de datos que un enlace de red o una ruta de acceso de red pueden transferir.

El concepto de ancho de banda es fundamental para las comunicaciones digitales, específicamente en redes de paquetes, lo que se refiere a la cantidad de datos que un enlace o ruta de acceso de red puede suministrar por unidad de tiempo. Para muchas aplicaciones de datos tales como las transferencias de archivos o multimedia, el ancho de banda disponible para la aplicación repercute directamente en el rendimiento de la misma. Incluso aplicaciones interactivas que suelen ser más sensibles a la latencia que el rendimiento, pueden beneficiarse del bajo retraso de extremo a extremo con los enlaces de alto ancho de banda y baja latencia de transmisión de paquetes.[5]

Debido a la información que se maneja dentro de nuestra Red Informática Universitaria (datos del alumno, información administrativa, etc.), es de vital importancia poder asegurarla contra el uso malintencionado, de posibles fallas en el sistema para evitar pérdidas parciales y/o totales de la misma, monitorear el uso de los recursos.

[4] (masadelante.com, 2008)

[5] (Prasad & Dovrolis, 2003)

La disponibilidad es un factor muy importante al que normalmente se le suele menospreciar; sin embargo, es ésta misma la que garantiza que los usuarios autorizados puedan acceder a la información y recursos de red cuando los necesiten, y es precisamente aquí donde el ancho de banda constituye un factor relevante para conseguirlo.[6]

Hay que tener en cuenta que no basta sólo con implementar la red de forma segura, sino que también debemos establecer métricas o medidas que nos ayuden a tener un mejor rendimiento en el consumo, las cuales deben ser medidas desde diferentes puntos de vista como el análisis, construcción, funcionalidad, documentación, métodos, procesos, usuarios, etc.

Quality of Service (QoS, Servicio de Calidad) es la capacidad de que nuestra red dé un buen servicio al tráfico de información sobre las tecnologías en que se trabaja, es decir, garantizar la transmisión de cierta cantidad de datos en un tiempo dado o dicho en otra forma, que nuestros usuarios de la red obtengan los servicios que decimos ofrecer.[7]

Generador de tráfico D-ITG

D-ITG (Distributed Internet Traffic Generator) es una plataforma capaz de producir tráfico a nivel de paquetes; replicar con precisión adecuada ambos procesos estocásticos IDT (entre el horario de salida) y PS (Packet Size), variables aleatorias.

[6] (Prasad & Dovrolis, 2003)
[7] (Cisco, 1999)

D-ITG soporta IPv4 y la generación de tráfico IPv6 y es capaz de generar tráfico en la capa de red, de transporte, y la capa de aplicación. D-ITG muestra propiedades interesantes en comparación con los generadores de tráfico.

Está actualmente disponible en plataformas Linux y Windows. Se presenta tanto en aplicación multihilo y multitarea. Los protocolos soportados son: TCP, UDP, ICMP, DNS, Telnet, VoIP (G.711, G.723, G.729, Voice Activity Detection, RTP comprimido). Los procesos estocásticos que se proporcionan, tanto en el caso de PS y IDT son: constantes, distribuidos uniformemente, una distribución exponencial, Pareto distribuida o, distribución de Cauchy, Normal distribuido, distribución de Poisson, Gamma distribuidos.

D-ITG puede realizar tanto una forma de retraso (OWD) de medición y de tiempo ida y vuelta (RTT) de medición, evaluación de la pérdida de paquetes, jitter y medición de rendimiento. Es capaz de almacenar la información sobre el tráfico enviado y recibido en los remitentes y los receptores. Además, permite al emisor y el receptor delegar la operación de logging en un servidor de registro remoto. Esta opción es útil cuando la capacidad de almacenamiento del receptor es limitada, por ejemplo, PDA, PC de bolsillo, etc. y cuando la información del registro debe ser analizada "en la marcha", por ejemplo, en caso de que el remitente se pregunte la velocidad de transmisión sobre la base de la congestión del canal y la capacidad del receptor.[8]

Herramientas para estimación de ancho de banda

Metodología Variable Packet Size (VPS)

[8] Universita' degli Studi di Napoli "Federico II"

Estima la capacidad de hops individuales. Actualmente no existe una técnica para medir el ancho de banda disponible en hops individuales.

Utiliza el campo time-to-live (TTL) del encabezado IP, para forzar a expirar en un hop particular. El router descarta el paquete de prueba y retorna un mensaje de error ICMP especificando el tiempo excedido.

La fuente usa el paquete ICMP recibido para medir el RTT (round trip time) al hop.

El RTT a cada hop depende de tres componentes de atraso:

- Retraso por serialización: Es el tiempo en transmitir el paquete en el enlace.
- Retraso por propagación: Es el tiempo que le toma a cada bit del paquete recorrer el enlace. Es independiente del tamaño del paquete
- Retraso de encolamiento: Es el retraso provocado por los dispositivos como routers y switches en sus puertos de entarda y salida.

VPS produce errores de estimación de la capacidad si el camino de datos incluye dispositivos de capa 2 (como switches store-forward) ya que producen retraso de serialización y no producen replicas ICMP de expiración TTL.

Metodología Packet Pair/train Dispersion Probing

Es usada para medir la capacidad de extremo a extremo a lo largo de un camino. La fuente envía múltiples pares de paquetes al receptor, cada par de paquetes consiste de dos paquetes del mismo tamaño enviados uno tras otro. La dispersión del par de paquetes, es la distancia en tiempo, entre los últimos bits de los dos paquetes.

En este tipo de metodología se asume que no se transporta otro tipo de tráfico, más que los paquetes de prueba.

Metodología Self-Loading Periodic Stream (SLoPS)

Es una de las más recientes metodologías para la estimación de ancho de banda disponible. Mide el acho disponible de extremo a extremo (end-to-end). La fuente envía un número de paquetes de igual tamaño (p.e k=100) hacia el receptor a una tasa R. Si R es más grande que el ancho de banda disponible A, ocasionará una sobre carga en el buffer de encolamiento.

El receptor revisa que la dispersión sea igual a la enviada y con ello determina el ancho de banda disponible. El método de ajuste de las tasas de transmisión es llamada “búsqueda binaria”.

Desarrollo del trabajo

Metodología

Se estudiarán las diferentes aplicaciones de software que implementan distintos algoritmos para la estimación del ancho de banda. Lo anterior es con la finalidad de escoger al menos dos de ellas para proceder a la obtención dichas aplicaciones con los algoritmos ya implementados. Por último, realizar las mediciones y presentar resultados.

TIPO DE INVESTIGACIÓN:

- Descriptiva
- Experimental

MÉTODO UTILIZADO:

- Método Inductivo

OBJETO:

- Métricas para estimación de ancho de banda.

MEDIO:

- Recopilación de información bibliográfica y referencias electrónicas así como también investigación de campo.

FIN:

- Proporcionar las bases necesarias para implementar estrategias que permitan estimar el uso de ancho de banda; así como la de futuras

investigaciones que permitan mejorar la calidad del servicio en la red universitaria.

Requerimientos del entorno de pruebas

Dispositivo	Fabricante	Modelo	Unidades
Switch	CISCO	2960	1
Router	CISCO	1800	2
Desktop	Compaq	EVO	1
Laptop	Toshiba	Satellite	1

Diseño de la arquitectura de pruebas

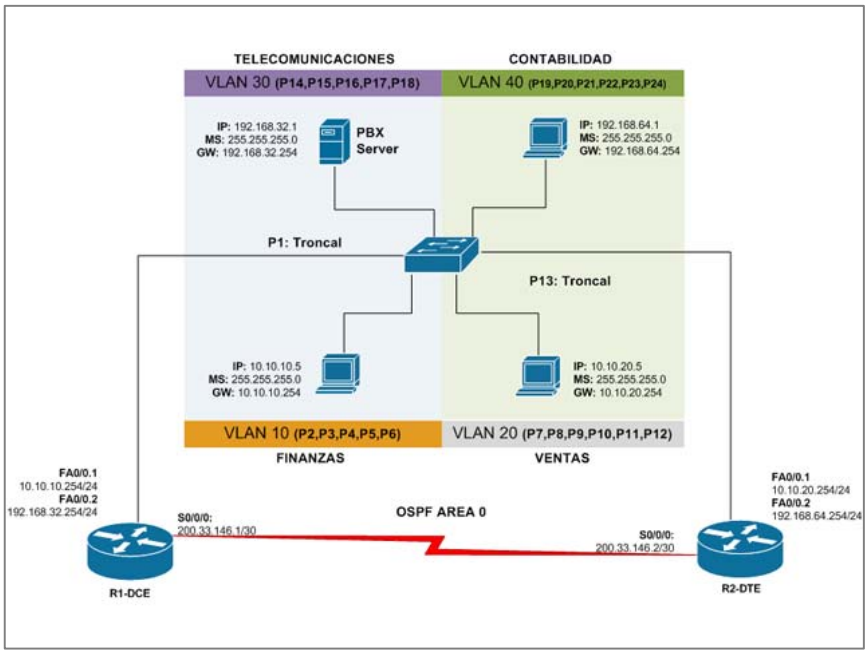

Figura 1: Esquema de tráfico controlado (elaboración propia)

En la figura 1, se observan tres dispositivos de red importantes: un switch y dos routers. Además cinco segmentos de red, es decir cuatro redes diferentes y configuradas cada una de ellas de forma virtual en un mismo dispositivo (VLANs), así como el segmento que conecta los dos routers; gracias a estos

dos últimos dispositivos existe comunicación total entre todos los segmentos mencionados.

Al observar la figura, se puede saber cuáles son los puertos disponibles para cada red o VLAN, así como la dirección IP, máscara de subred y puerta de enlace. Para la VLAN 10 se tienen los puertos P2, P3, P4, P5 y P6 del switch; la dirección de red 10.10.10.0 con máscara de 24 bits y puerta de enlace 10.10.10.254; por ejemplo.

Cada router está configurado para encaminar el tráfico de dos VLANs, de tal manera que el router con el nombre R1-DCE tiene configuradas las VLAN 10 y 30 por medio del protocolo de enlace troncal IEEE 802.1Q y el router R2-DTE las VLAN 20 y 40. De esta forma, si algún host de la VLAN 10 quiere enviar paquetes a otro host de la VLAN 40 tenga que pasar por el enlace serial que comunica los 2 routers; como si estuvieran en ciudades distintas (ver anexo: configuración de dispositivos de red).

De manera similar los paquetes de provengan de la VLAN 30 hacia la 20 tendrán que pasar por los dos routers para poder comunicarse, teniendo con esto un escenario básico de simulación de tráfico de paquetes.

Herramienta de generación de tráfico

Para la pruebas de generación de tráfico se optó por la herramienta D-ITG[9] disponible en línea en http://www.grid.unina.it/software/ITG/download.php. En este caso se descargó la versión para Windows ya que se está realizando las pruebas bajos este sistema operativo.

[9] (Alessio Botta, 2007)

Una vez descargada la versión zip de los binarios (ejecutables) se descomprimió en el directorio E:\proyecto1\d-itg en cada uno de los equipos de prueba, teniendo el nuevo directorio descomprimido, la estructura siguiente.

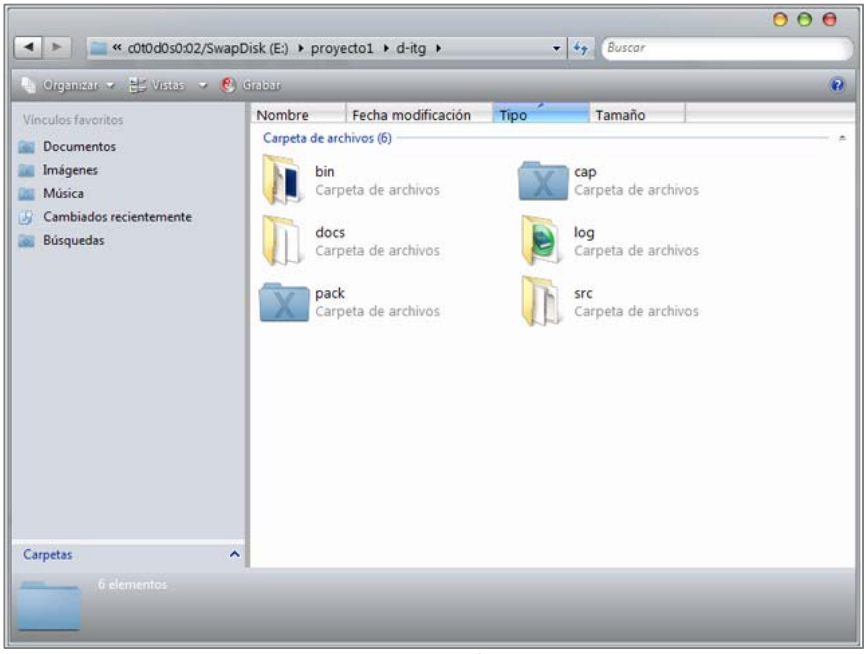

Figura 2: Estructura de archivos de D-ITG

En el directorio *bin* se encuentran las aplicaciones para generar y recibir el tráfico y otras que a continuación se describen brevemente.

ITGSend.exe: Esta aplicación nos permite enviar paquetes de diferente tipo (voz, datos, personalizado) a un determinado equipo. Adicionalmente podemos especificar el tiempo que durará la prueba, y el protocolo bajo el que se ejecutará (TCP, UDP, ICMP).

ITGRecv.exe: Esta aplicación es la encargada de recibir los paquetes en el equipo de otro extremo. Adicionalmente podemos especificar los parámetros para poder almacenar los datos recibidos en un archivo de texto sin formato (logFile).

ITGLog.exe: Esta aplicación se integra con ITGRecv.exe para poder generar el archivo log, durante el tiempo que dure la prueba de tráfico.
ITGDec.exe: Nos permite convertir los datos recibidos durante el tiempo de la prueba en un conjunto de valores numéricos que posteriormente podemos analizar y graficar.

ITGGUI.jar: Es un archivo de aplicación Java que nos permite de manea sencilla, mediante una interfaz gráfica, configurar las opciones de recepción y envío de paquetes.

Ejecución de las pruebas

Para poder realizar la primera prueba se conectaron dos máquinas al switch en diferentes segmentos de red o VLAN, de tal manera que cuando se genere tráfico, los paquetes tengan que pasar por ambos routers.

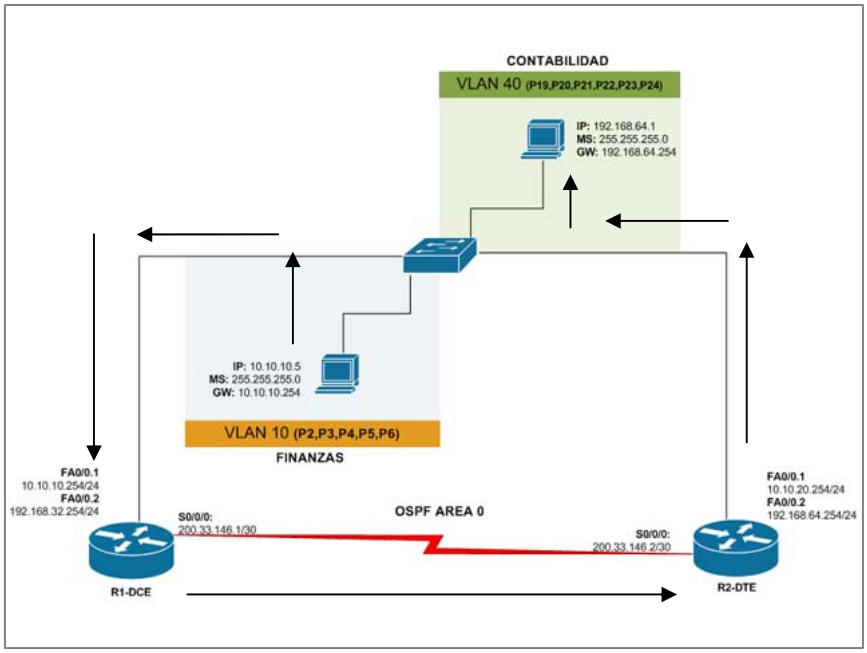

Figura 3: Configuración de la primera prueba (elaboración propia)

Para las pruebas se hará referencia al equipo con la dirección IP 10.10.10.5 de la VLAN 10 como Host A y al host de la VLAN 40 como Host B.

A continuación se ejecuta el archivo ITGGui.jar para configurar la utilidad de envío de paquetes por la red.

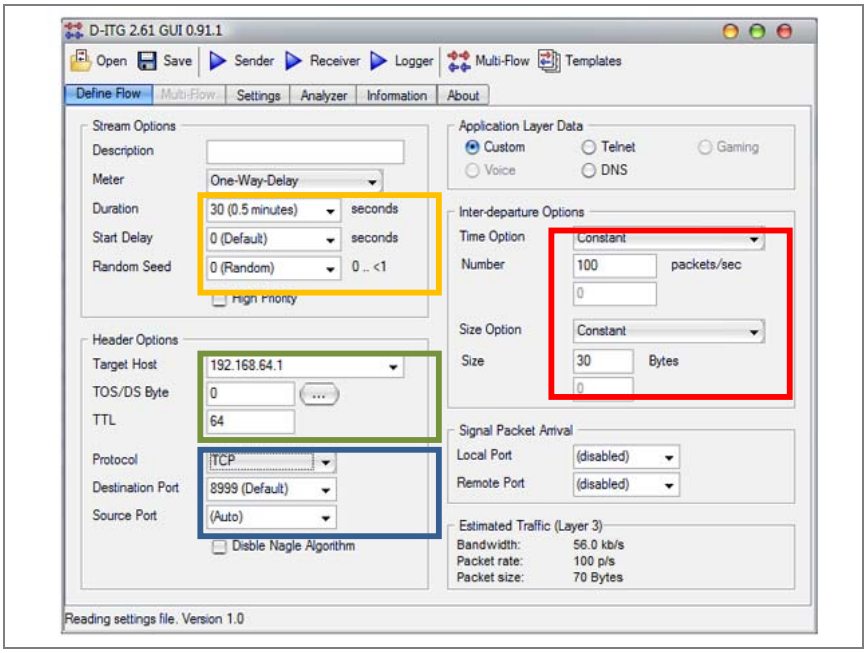

Figura 4: Configuración de la interfaz gráfica ITG-GUI.jar

Mediante esta aplicación se puede configurar fácilmente el tiempo que durará la prueba (recuadro en amarillo), el tipo de paquetes que se enviará (recuadro rojo), la dirección del destino de dichos paquetes (recuadro verde), y el protocolo mediante el cual se ejecutará la transferencia (recuadro azul). Una vez configurados estos parámetros se puede iniciar el proceso de transmisión.

Para poder recibir los paquetes del otro extremo es necesario iniciar la herramienta para captura de paquetes ITGRecv.exe.

La figura siguiente muestra como inicializar la herramienta ITGRecv.exe.

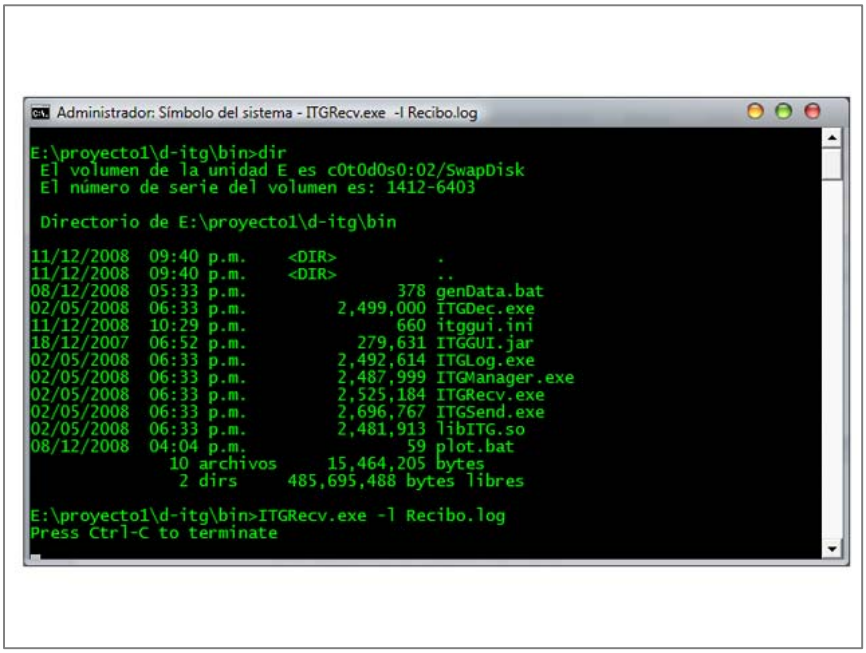

Figura 5: Inicialización de la aplicación de escucha ITGRecv.exe

La instrucción ITGRecv.exe –l Recibido.log, pone a “escuchar” la aplicación del tráfico en este caso el host de pruebas A 10.10.10.5 en el puerto 8999, y lo comienza a almacenar en el archivo Recibido.log. Este archivo contiene los datos de la transmisión, sin embargo no es posible analizarlos ya que no contiene un formato el cual permita determinar información a simple vista, por lo que se hace uso de otra de las utilidades de esta herramienta, la cual permite generar valores numéricos a partir de todo este conjunto de datos almacenados en el archivo.

La herramienta que permite generar valores numéricos a partir de los datos recibido es ITGDec.exe. Esta aplicación recibe como parámetro el nombre del archivo log generado, en este caso “Recibido.log” y a continuación permite generar mediante otros cuatro parámetros importantes, archivos de salida que

contienen información para poder analizar o graficar los resultados de la prueba.

La opción –j <milisegundos>permite generar un archivo con el nombre jitter.dat esta archivo contiene datos sobre la variación en el retardo promedio que se generó durante la prueba. La opción –b <milisegundos> generará el archivo bitrate.dat que contiene datos respecto a la tasa de bits de la transmisión. La opción –d <milisegundos> genera el archivo delay.dat referente al retardo y finalmente la opción –p <milisegundos> generará el archivo packetloss.dat que contiene información sobre los datos perdidos.

Adicionalmente la ejecución de la herramienta ITGDec.exe genera un resumen con datos importantes de la transmisión.

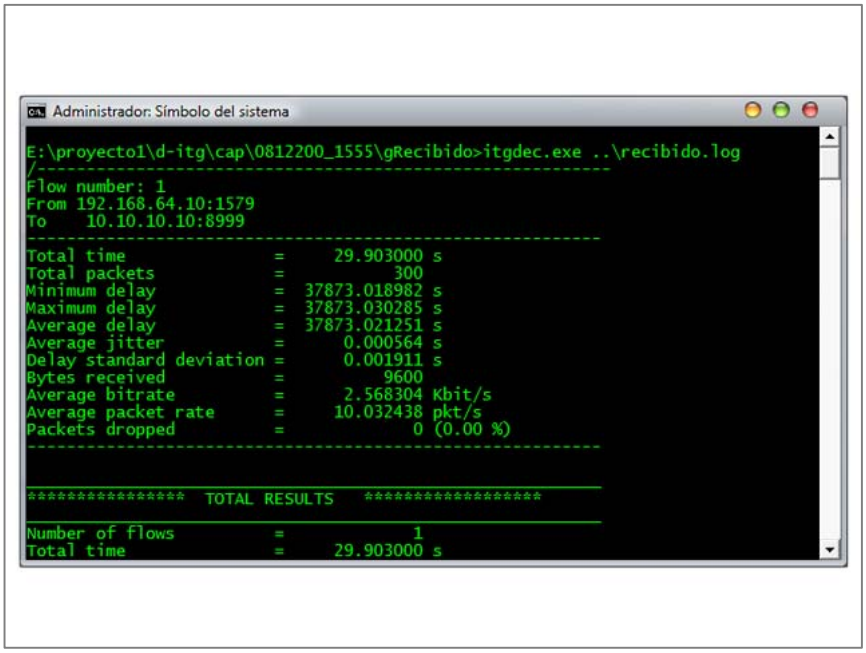

Figura 6: Resumen de la prueba mediante ITGDec.exe

Para poder graficar estos datos, necesitamos hacer uso de un Script de gnu Octave, el cual permitirá, a partir de los archivos *.dat* generados con la utilidad

ITGDec.exe, crear gráficas que proporcionarán una mejor idea de los resultados obtenidos.

Para la generación de dichas gráficas se utiliza Octave, se puede descargar del sitio oficial de Octave http://www.gnu.org/software/octave/download.html.

Una vez instalado Octave, se procede a configurar las variables de entorno del sistema, con el propósito de poder usar esa configuración al momento de ejecutar unos scripts que nos permitirán generar los cuatro archivos Bitrate.dat, Jitter.dat, Delay.dat y Packeloss.dat además de generar las gráficas con Octave, todo al mismo tiempo.

Es necesario especificar en las variables *PATH* y *BIN* del usuario actual, las rutas para acceder los directorios *bin* de las aplicaciones D-ITG y *Octave*, estas rutas deberán estar separadas por ";" (punto y coma) unas de otra. Por ejemplo, si se instala Octave en el directorio D:\Programas\Octave y el D-ITG en E:\Proyecto1\d-itg; las rutas para las variables de entorno tanto *bin*, como *path* quedarán de la siguiente manera:

- D:\Programas\Octave\bin; E:\Proyecto1\d-itg\bin

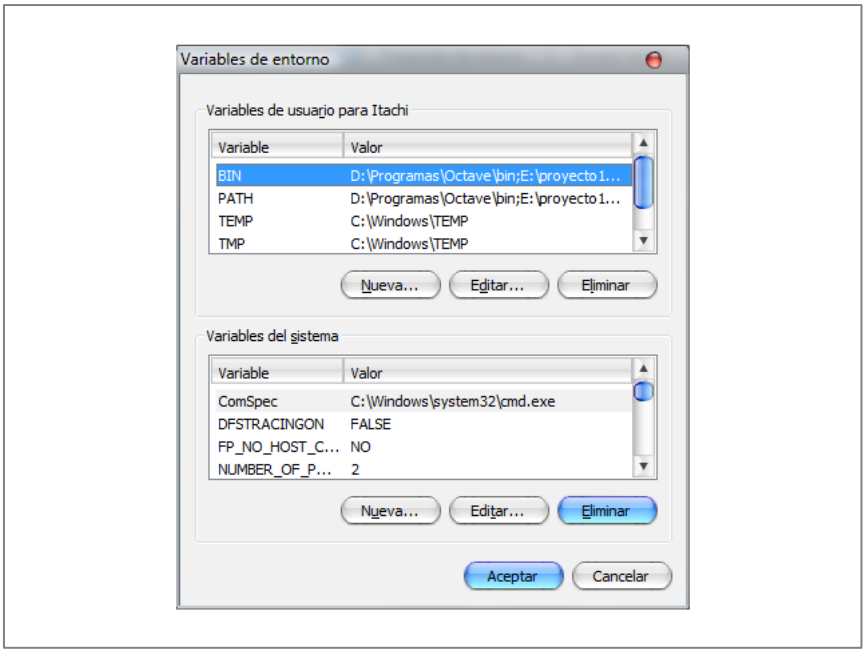

Figura 7: Configuración de las variables del entorno

Para facilitarla generación de gráficas y los archivos de datos, se escribieron y utilizaron dos scripts: genData.cmd y plot.cmd. El primero contiene las instrucciones de símbolo de sistema de Windows para generar (haciendo uso de la utilidad ITGDec y los archivos generados durante la prueba delay.dat, Jitter.dat, packetloss.dat y bitrate.dat) valores numéricos que posteriormente pueden ser graficados. El segundo script contiene las instrucciones necesarias para que a través de GNU Octave dichos valores generados por el primer script, sean graficados. En la sección de anexos se detalla el contenido de cada uno.

Los scripts deberán estar situados dentro del directorio *bin* de la herramienta D-ITG. Esto permitirá a los mismos obtener como parámetros el archivo LOG capturado por ITGRecv.exe y los intervalos en milisegundos de separación entre cada muestra, por ejemplo si se quiere conocer cuál es el comportamiento de los paquetes enviados cada segundo, especificamos un valor de 1000 (mil milisegundos).

Adicionalmente se pueden crear carpetas para cada una de las capturas especificando como nombre la fecha y hora de la realización de la prueba; esto con el propósito de tener organizados cada uno de los archivos .dat que se generan, ya que cuando la herramienta ITGDec.exe los genera, reemplazará cualquier otro existente. Es recomendable tener organizado en carpetas diferentes cada una de capturas o pruebas que se vayan realizando (archivos de salida .LOG).

Mediante la instrucción ***genData recibido.log 100***, se especifica que el nombre del archivo a analizar seguido del intervalo en milisegundos de muestreo. Se pueden generar los diferentes gráficas que se describen a continuación.

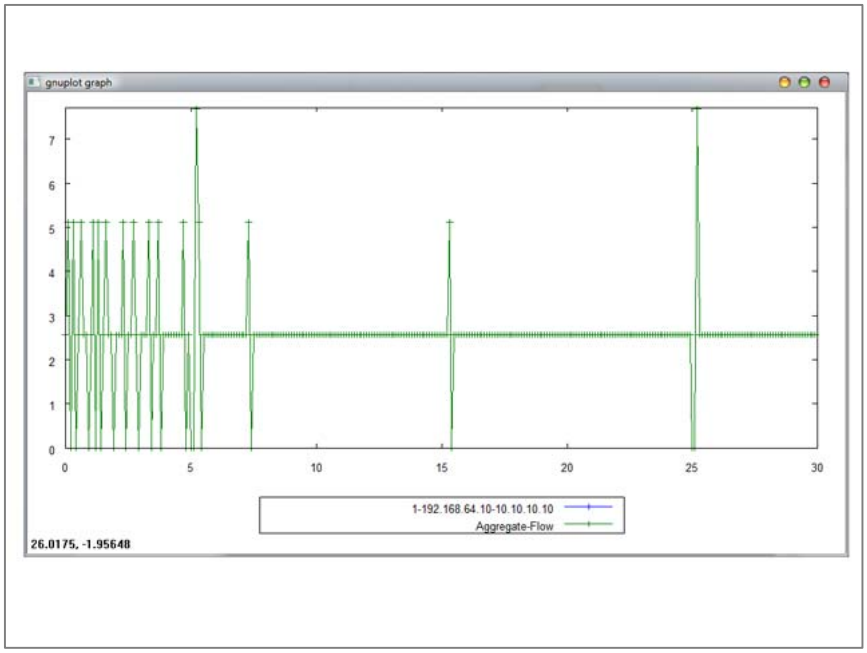

Figura 8: Retardo promedio del tráfico en el ambiente controlado

La gráfica que se muestra a continuación representa la tasa de bits promedio que se mantuvo durante la transferencia, podemos ver claramente que este valor fue de aproximadamente 2.5 Kbps. Como si ilustra en la figura cuatro, el

protocolo de la prueba fue TCP y se envió una cantidad de 100 paquetes constantes de 30 bytes durante un periodo tiempo de 30 segundos. La figura seis contiene un resumen de los datos generados en la prueba como podemos apreciar coincide con el generado en la grafica (2.5 Kbps).

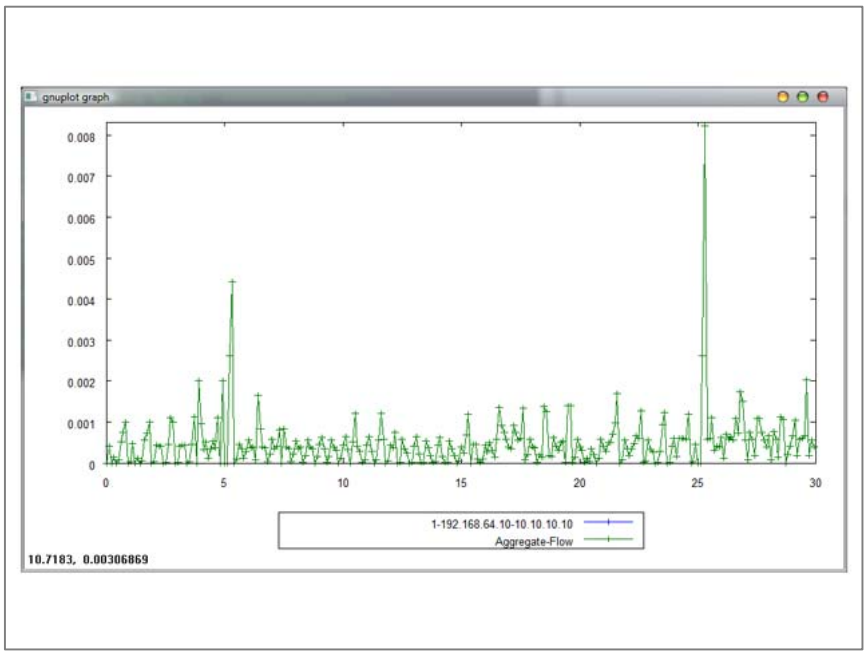

Figura 9: Jitter promedio de la transferencia en el ambiente controlado

La gráfica mostrada a continuación representa *jitter* promedio o variación del retardo promedio en segundos que se mantuvo durante la transferencia, podemos ver que este valor fue de aproximadamente 0.0005seg.

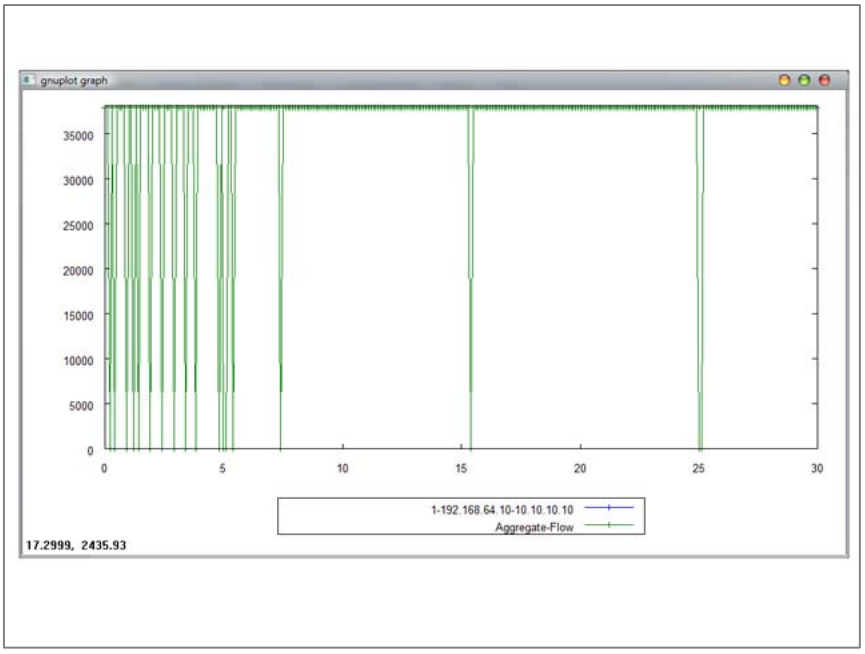

Figura 10: Retardo promedio de la transmisión en el ambiente controlado

La gráfica mostrada a continuación representa el retardo promedio en milisegundos que se mantuvo durante la transferencia, en este caso aun no se ha podido determinar con exactitud el significado de estos resultados, puesto que muestra una elevada cantidad de retardo en segundos que no concuerda con la realidad por lo que se debe considerar que dicha cantidad esté elevada a una potencia negativa, ya que D-ITG no proporciona ningún método de sincronización entre el emisor y el receptor; por lo tanto, se debe utilizar métodos adicionales para la sincronización entre ambos puntos en lo que se realiza la prueba como NTP (Network Time Protocol).

Por último la gráfica de paquetes perdidos nos muestra una contundente pérdida de 0.0 de paquetes enviados, lo cual significa que todos los paquetes fueron recibidos, en este escenario ideal de pruebas era uno de los objetivos primordiales.

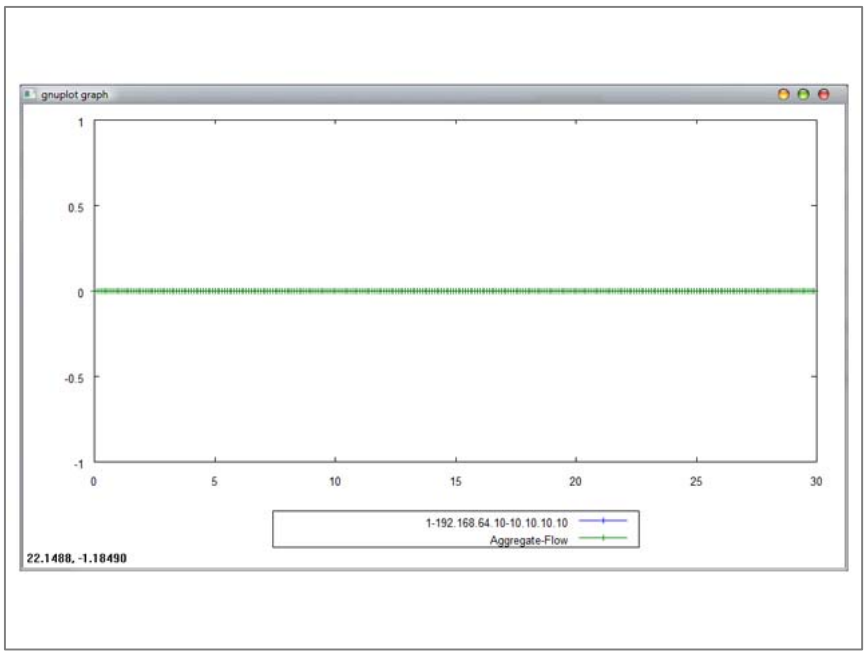

Figura 11: Pérdida de paquetes promedio es 0.0 en el ambiente controlado.

Pruebas de ancho de banda disponible en entorno virtual

En este punto se han realizado pruebas de trasmisión de datos a través de la red implementada. También se ha comprobado la funcionalidad de la arquitectura mediante el empleo de la herramienta de generación de tráfico D-IGT descrita anteriormente.

Hasta el momento, se ha obtenido una parte de la información necesaria para estimar el ancho de banda disponible en la red, información que ha permitido observar como incluso en los entornos controlados existen variaciones en factores como el jitter, el retardo y la tasa promedio de transferencia. Definir estas variables puede ser útil a la hora de realizar pruebas para estimar el ancho de banda disponible ya que estadísticamente, se puede tomar un porcentaje promedio de retardo del jitter o la tasa de bits promedio de la transferencia con el propósito de proveer a este entorno ideal de pruebas un enfoque más realista.

Todo lo anterior son variables que pueden influir y que deben tomarse en cuenta para un análisis de las posibles causas de pérdidas de datos o inesperado comportamiento de la red, como podría ser la obtención de un valor de ancho disponible mucho menor al esperado de la red.

Para poder realizar las pruebas del ancho de banda disponible se analizaron tres soluciones disponibles que utilizan distintos algoritmos para estimar en ancho de banda. Estas herramientas de software están basadas en código abierto el cual se encuentra en descarga directa del sitio de los autores y pueden ser ejecutadas utilizando el sistema operativo Linux.

Haciendo uso nuevamente de la arquitectura de red anteriormente empleada, se implementa un entorno virtual de pruebas haciendo uso de dos equipos GNU/Linux ejecutando cada una de las herramientas que se mencionarán, además se compararán los resultados para poder tener un panorama más general sobre lo que hacen estas herramientas y tener una mayor certeza de sus resultados; contrastar los mismos de tal manera que se pueda tener una verdadera base que permita generar un modelo para estimar el ancho de banda disponible de manera estadística.

Para poder implementar el entorno de pruebas se utilizan equipos para la generación de tráfico; host A y host B (ver anexo). Adicionalmente se requiere la instalación de un software de virtualización que permita ejecutar el sistema operativo Linux, ya que las herramientas utilizadas en este trabajo de investigación para la estimación de ancho de banda disponible, hacen uso de este sistema operativo.

Como primer punto se instalará y configurará la red del software de virtualización (VMware), ya que es una de las aplicaciones más utilizadas y con más experiencias en el ámbito de la virtualización.

Instalación del software de virtualización

Para poder instalar el software de virtualización VMware podemos descargar una versión de prueba directamente del sitio del fabricante. Para esto debemos ingresar a http://www.vmware.com una vez descargada la aplicación, procederemos a localizar el archivo VMware Workstation 6.exe en el directorio donde descargamos la aplicación y se iniciará un asistente de instalación que no es más complicada que dar unos cuantos clicks en el botón siguiente del asistente.

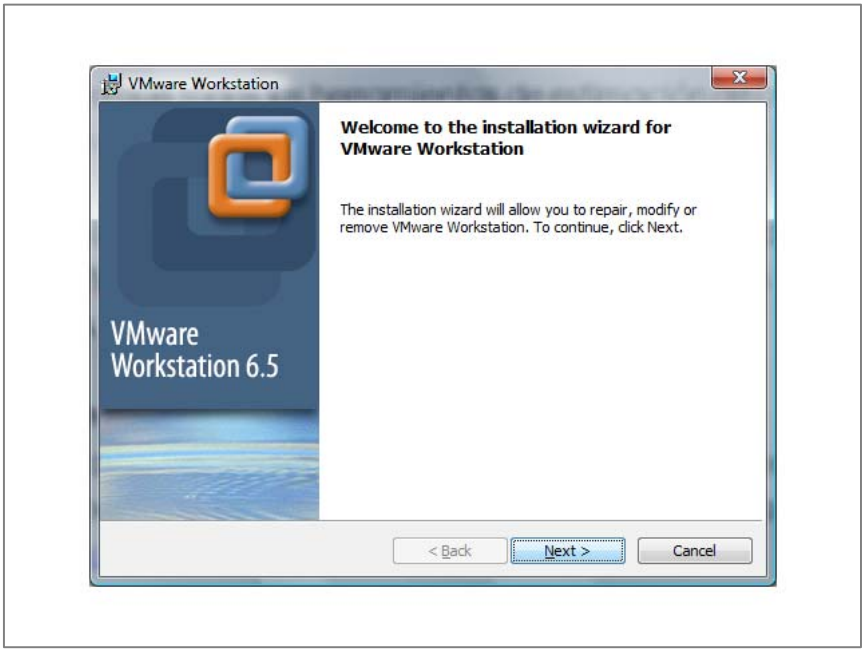

Figura 12: Instalación del software de virtualización

Una vez instalada la aplicación podemos accesar a ella a través del acceso directo creado en el escritorio por el programa de instalación. El programa nos instala dos servicios básicos de red que no utilizaremos por ahora.

Los servicios de NAT y DHCP son innecesarios para la implementación del entorno de pruebas, e incluso pueden entorpecer los resultados y análisis que se espera realizar, ya que estas aplicaciones generarán tráfico adicional en la red; minimizando así la cantidad de tráfico indeseable y que se pueda controlar con el propósito de obtener resultados y análisis más precisos sobre lo que se está ejecutando en la red.

Para detener los servicios, requerimos ejecutar una consola de Símbolo del Sistema de Windows Vista en modo administrador; e introducir los comandos siguientes. Para esto buscamos el ícono de símbolo de sistema en Windows y

presionamos el botón derecho del mouse sobre el mismo y seleccionamos la opción “Ejecutar como administrador”

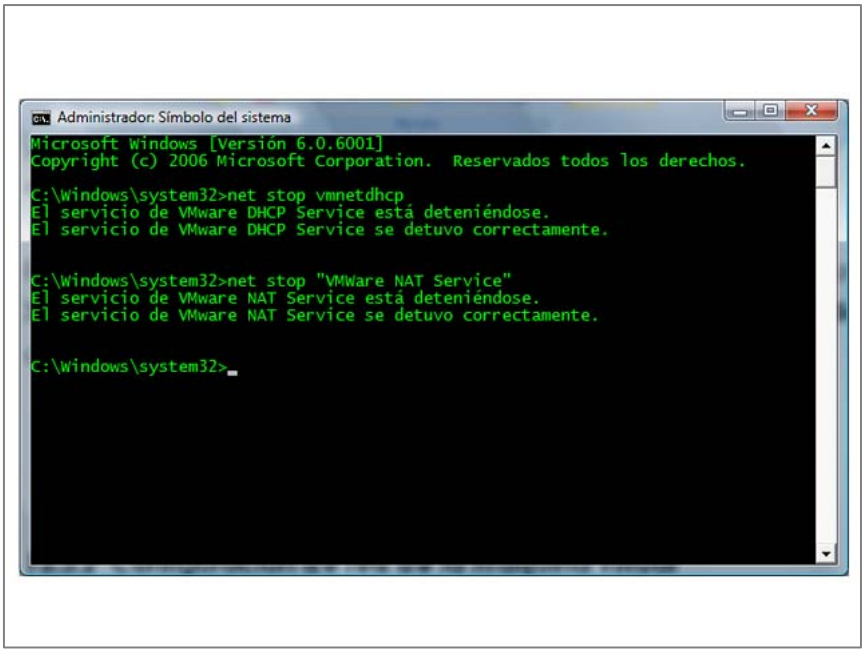

Figura 13: Detener los servicios de NAT y DCHP virtuales

Instalado el software de máquina virtual, y detenidos los servicios pertinentes; se debe configurar la red virtual de la máquina virtual. La tarjeta de red se configura en modo Bridge. Este modo de VMware permite a la tarjeta adaptadora de red virtual replicar el estado físico de un adaptador de red real del equipo es decir, permite a la tarjeta de red virtual comportarse, adoptar la configuración, estado y tráfico de la tarjeta de red física; con ello se puede simular que el sistema operativo virtual tendrá una tarjeta de red en modo bridge dedicada únicamente para realizar pruebas.

Es muy importante tener en cuenta que durante este tipo de pruebas ningún otro tráfico será permitido a través de la interfaz Ethernet, ya que estará

reservado en forma exclusiva para las pruebas de medición del ancho de banda.

Para poder configurar el adaptador de virtual en modo bridge debemos primeramente iniciar la aplicación VMware y asegurarnos que los servicios de NAT y DHCP hayan sido detenidos aunque esto no es indispensable, debemos hacerlo para poder tener un tráfico más limpio.

Posteriormente nos dirigiremos a la opción Virtual Networks Editor del menú Edit; se presentará un interfaz de configuración de los dispositivos de red que el software de máquina virtual puede gestionar. Debemos de inmediato dirigirnos a la ficha o pestaña Host Virtual Network Mapping, En esta sección seleccionaremos de nuestra lista desplegable, en el dispositivo vmnet0, el adaptador Ethernet físico de nuestro equipo, como se muestra en la figura siguiente.

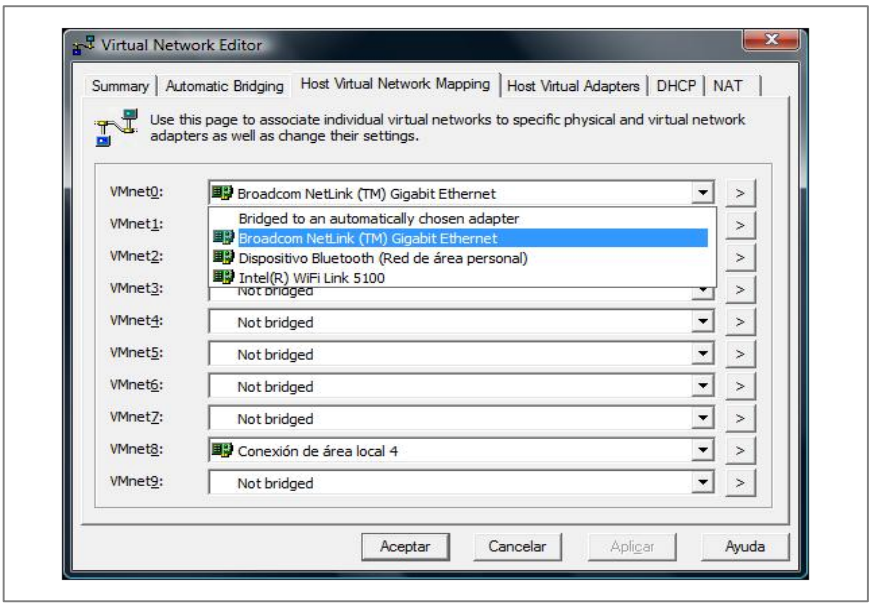

Figura 14: Configuración en modo bridge de la interfaz de red.

El sistema operativo de pruebas seleccionado es GNU/Linux Ubuntu. Este sistema operativo que sigue la filosofía de software libre, es ligero, compacto, sencillo de usar y configurar y provee actualmente una amplia promoción para los entornos académicos en el mundo (consultar http://www.ubuntu.com). Basado en GNU/Linux Debían, uno de las distribuciones Linux más antiguas y con más experiencia, en el ámbito del software libre; Ubuntu nos presenta un entorno estable y probado para realizar múltiples tareas de propósito académico y de investigación.

Para que podamos instalar nuestro sistema operativo Linux necesitaremos descargar la imagen ISO de instalación del sitio oficial de Ubuntu. Una vez descargada la imagen del sistema operativo, podremos grabarla en un CD-ROM ó montarla en un dispositivo virtual. En el caso de VMware permite realizar instalaciones de sistemas operativos directamente desde una imagen ISO.

Es necesario para instalar nuestro sistema operativo Ubuntu directamente desde la imagen ISO, sin necesidad de software de terceros o grabarla en un CD-ROM; crear una maquina virtual compatible con Linux y especificarle a la unidad de CD-ROM virtual que debe iniciar desde una imagen ISO.

Para hacer esto debemos crear la máquina virtual primeramente para esto, seleccionaremos el menú File/New/Virtual Machine y cuando se nos pregunte el tipo de configuración seleccionaremos personalizada (Custom).

Figura 15: Asistente de creación de maquina virtual

Posteriormente presionamos el botón izquierdo de mouse en la opción “Next” (siguiente) para especificar la imagen ISO de instalación que anteriormente habíamos descargado del sitio oficial de Ubuntu.

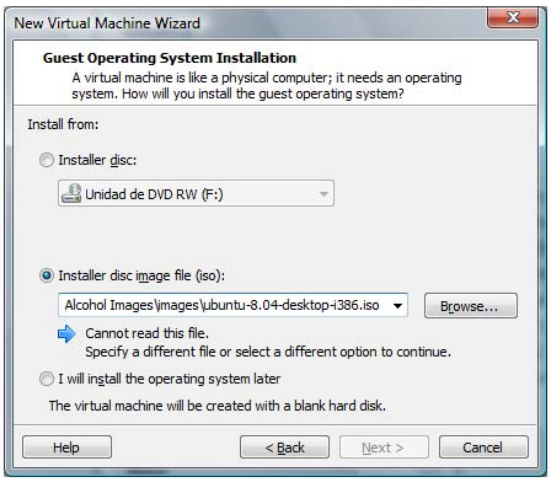

Figura 16: Seleccionar el origen de las instalación

Cuando se nos solicite especificar la configuración de red que deberá tener la máquina virtual que estamos creando, especificaremos que será en modo bridge ya que éste es el modo que anteriormente habíamos configurado.

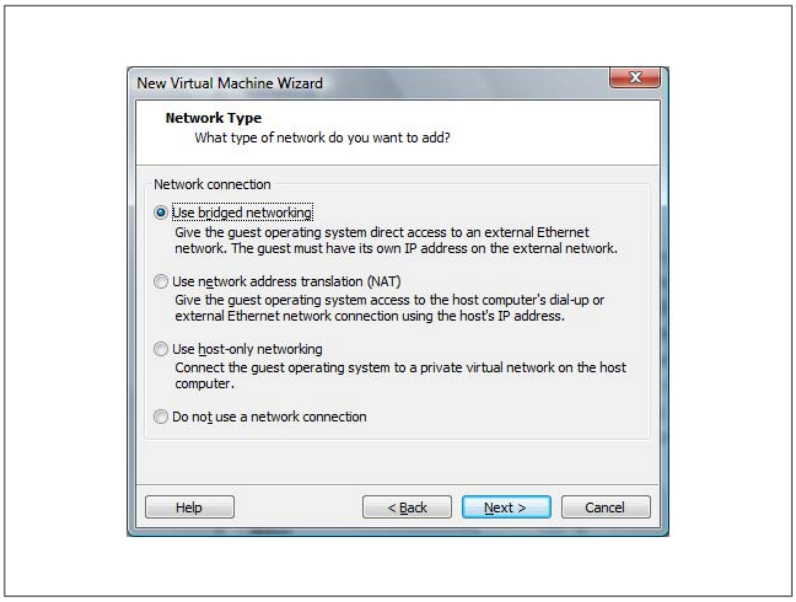

Figura 17: Configuración de red para la maquina virtual

Al terminar el asistente de instalación nos solicitará si deseamos iniciar la máquina virtual creada al terminar dicho asistente, a lo cual especificaremos que sí queremos iniciarla.

Al arrancar la máquina virtual se presenta de tal manera como si estuviéramos iniciando o encendiendo una computadora de manera ordinaria.

Después de unos segundos la imagen ISO especificada durante el proceso de creación de la máquina virtual se iniciará y nos mostrará una pantalla en donde deberemos especificar que se trata de una nueva instalación de Ubuntu. El proceso de instalación tarda aproximadamente entre 13 y 20 minutos empleando las características de hardware para los host A y B detallados en el apartado de anexos.

Iniciado el proceso de instalación se presentará un asistente sencillo que nos guiará en el proceso de instalación. Este proceso suele ser casi automático y puesto que estamos trabajando en una máquina virtual, no tenemos que tener en cuenta consideraciones adicionales más que dejarnos guiar de la mano del proceso de instalación de Ubuntu ya que cualquier cambio o formateo de las unidades de disco únicamente se reflejará en la configuración de la máquina virtual sin afectar nuestro equipo real.

Una de las razones más importantes del por qué utilizar un entorno virtual para este tipo de pruebas, es que cualquier error que cometamos simplemente bastará con restaurar una copia en disco de la máquina virtual para tener todo configurado y funcionando de nuevo.

Una vez instalado nuestro sistema operativo Linux y configurado nuestro ambiente virtual de red y aplicaciones procederemos a instalar y probar las herramientas de estimación de ancho de banda disponible para nuestra red.

Herramientas de estimación de ancho de banda

Las tres herramientas que instalaremos y analizaremos están basadas en código abierto y están disponibles para su descarga y modificación directamente de las páginas de sus autores. Dichas herramientas están disponibles para el sistema operativo GNU/Linux y requieren de un compilador gcc y del ligador make instalados para generar las versiones binarias ejecutables.

En cualquiera de los casos la compilación y puesta en marchas de las herramientas es sencilla y rápida.

Haciendo uso de la arquitectura de red construida anteriormente para la generación de tráfico, se emplea la misma para estas pruebas y se trabaja con los mismos segmentos de red que anteriormente definimos.

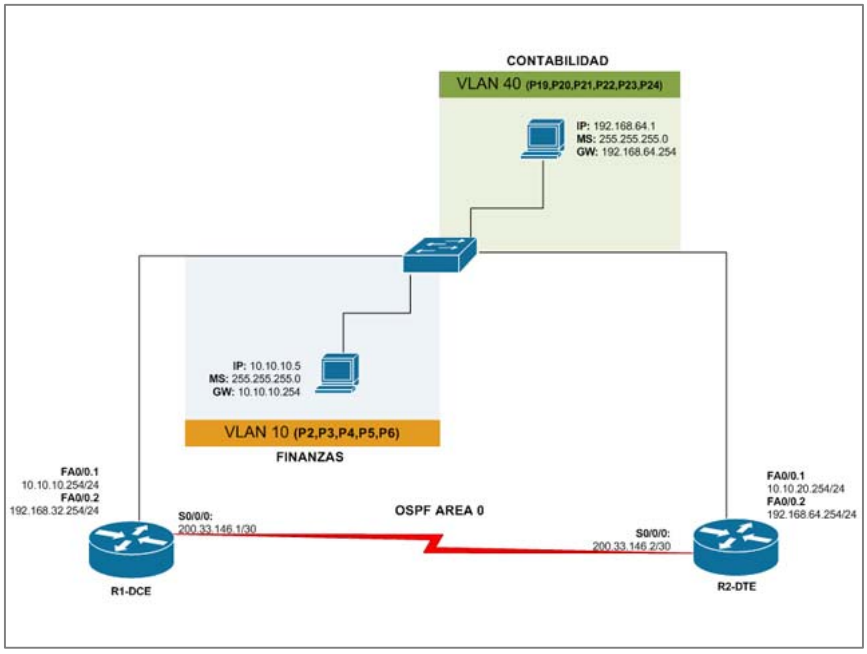

Figura 18: Topología para las pruebas de ancho de banda

También debemos configurar los dos equipos de pruebas con una dirección válida dentro de los segmentos 10.10.10.0/24 y 192.168.64.0/24. Esto lo hacemos con el comando ifconfig

- Host A: ifconfigeth0 10.10.10.10 netmask 255.255.255.0
- Host B: ifconfigeth0 192.168.64.10 netmask 255.255.255.0

Instalación de IGI-PTR

Esta herramienta nos permite estimar el ancho de banda disponible de extremo a extremo, es decir entre dos host o puntos de la red. Define el ancho de banda disponible como el ancho de banda residual en la ruta, y el cual puede ser

calculado como la capacidad menos la carga (Algoritmo SLoPS). Está disponible para su descarga directamente en http://www.cs.cmu.edu/~hnn/igi/ en formato de código fuente, por lo que para poder utilizarla tendremos que descargarla y compilarla.

En este caso se trabajará con la versión 2.1. El nombre del archivo que nos proporciona el sitio es igi-ptr-2.1.tgz (versión normal para UDP). Una vez que lo tengamos debemos descomprimirlo dentro de algún directorio; para esto necesitamos abrir una aplicación de consola y usar el comando tar –xvzf igi-ptr-2.1.tgz para descomprimir el código fuente de la aplicación. Posteriormente tras descomprimir el archivo se procede a la generación de ejecutables mediante la instrucción make. Para poder compilar y generar los ejecutables de IGI/PTR, basta con cambiarnos de directorio a igi-ptr.2.1 después de ser descomprimido y teclear la instrucción *make*.

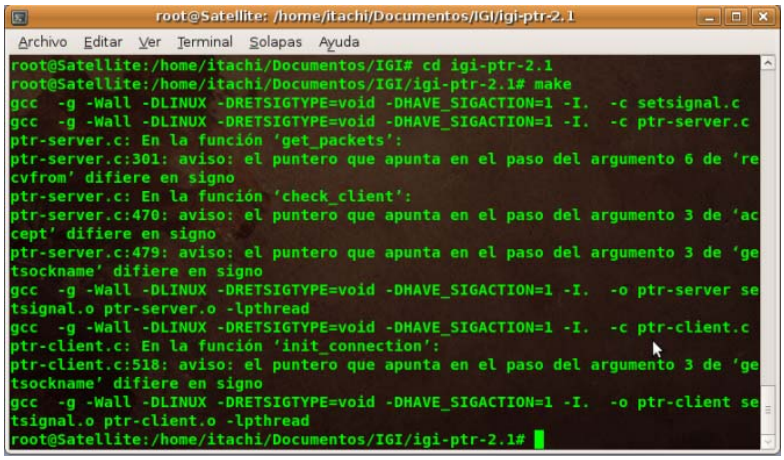

Figura 19: Generación de ejecutables de IGI-PTR

Para poder realizar la primera prueba debemos haber configurado en ambos equipos nuestra dirección IP correspondiente además inicializar la aplicación servidor de la herramienta IGI-PTR. Esta aplicación abrirá un puerto en el Host A y se quedará a la escucha o espera de la petición de conexión de la aplicación cliente.

Una vez inicializada la aplicación del servidor, debemos utilizar la aplicación cliente en el otro equipo de la red (Host B). En este caso inicializaremos el cliente con la dirección IP 10.10.10.10/24. Cuando la aplicación se conecte con el servidor, comenzará a desplegarnos información sobre los paquetes enviados y al final nos dará un resumen de la prueba y el ancho de banda disponible que fue calculado.

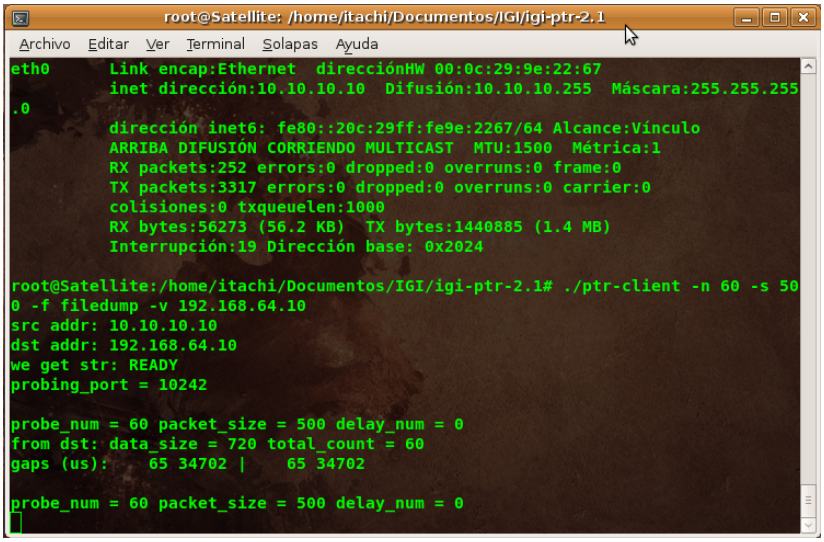

Figura 20: Inicialización de la aplicación cliente de IGI-PTR

Tras la ejecución de la instrucción. */ptr-client –n 60 –s 500 –f filedump –v 192.168.64.10* se comienza a generar el análisis de ancho de banda entre los dos puntos. La prueba se ejecuta sobre el protocolo UDP ya que esta versión de IGI/PTR (versión normal) se ejecuta bajo este protocolo.

Las opciones especificadas en el cliente se refieren a:

- -n 60: Cantidad de paquetes de la prueba
- -s 500: Longitud o tamaño de los paquetes en bytes
- -f filedump: Guardar en un archivo de texto los resultados de la prueba
- -v: Desplegar en pantalla información sobre el proceso de prueba.

- 192.168.64.10: Dirección IP del equipo que ejecuta la aplicación servidor (ptr-server)

Terminada la prueba se nos presenta un ancho de banda disponible de 0.116 Mbps. Cabe recordar que los equipos de prueba están conectados a cables de 100 Mbps, sin embargo dado que todo el trafico tiene que pasar por los encales seriales de 128kbps de cada router (ver figura 1), la velocidad en este punto se limita a esa velocidad ideal.

Instalación de Pathload

El procedimiento de instalación de Pathload es muy similar al que llevamos a cabo con la herramienta IGI-PTR al igual que en esta última, Pathload es una aplicación que nos permite estimar el ancho de banda disponible entre dos puntos y utiliza el algoritmo SLoPS para estimar el ancho de banda disponible.

Se puede descargar el código fuente de Pathload directamente del sitio http://www.cc.gatech.edu/fac/Constantinos.Dovrolis/bw-est/ Pathload.tar.gz una vez descargado se procede a descomprimir y compilar el código para generar las aplicaciones cliente y servidor de la herramienta.

Posteriormente al descomprimir el archivo Pathload.tar.gz y compilar la aplicación debemos generar los archivos ejecutables de la aplicación cliente (pathload_rcv) y servidor (pathload_snd). Para compilar debemos descomprimir mediante la instrucción tar –xvzf Pathload.tar.gz y cambiarnos al directorio Pathload que se crea después de la descompresión. Estando dentro de este directorio, debemos ejecutar dos instrucciones. La primera es *./configure* finalizada dicha instrucción ejecutar la instrucción *make*.

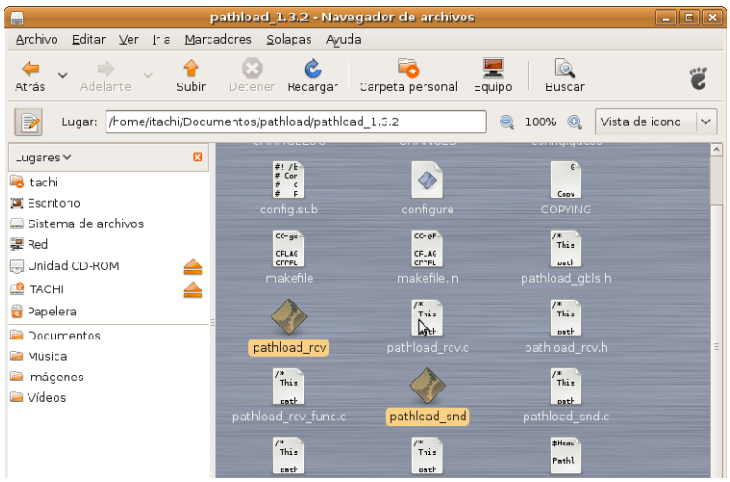

Figura 21: Aplicaciones cliente y servidor de pathload

Al realizar las pruebas de ancho de banda con Pathload emplearemos las dos aplicaciones que la herramienta nos proporciona, el Pathload_rcv y el Pathload_snd. Poner en marcha la herramienta es muy similar al de IGI-PTR ejecutamos un servidor que estará esperando solicitudes y posteriormente un cliente se conectará con éste y comenzarán a analizar el ancho de banda disponible durante la transmisión de datos de prueba.

En este caso en particular iniciaremos la aplicación servidor (Pathload_snd) en el equipo con la dirección IP 10.10.10.10 y pondremos al equipo 192.168.64.10 a enviar solicitudes al servidor.

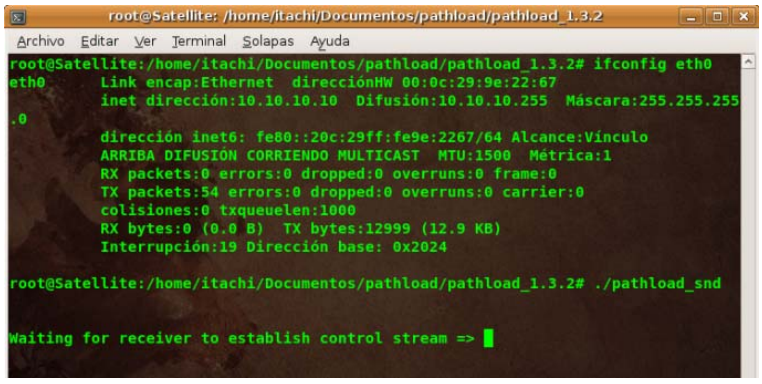

Figura 22: Iniciar la aplicación servidor de pathload

Terminada la prueba la aplicación cliente nos genera un resumen que proporciona un ancho de banda disponible de 0.12 Mbps a diferencia de IGI-PTR que nos reportó un ancho de banda de 0.116 Mbps.

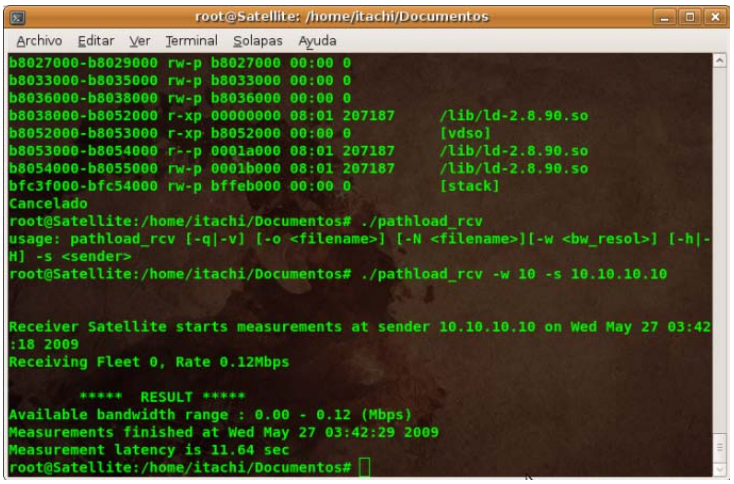

Figura 23: Resultados de la prueba reportados por el cliente

Instalación de Pathchirp

Para poder utilizar Pathchirp, debemos obtener el código fuente (http://www.spin.rice.edu/Software/pathChirp/pathchirp-2.4.1.tar.gz), compilar la aplicación para generar los binarios y ejecutar las aplicaciones para enviar y recibir paquetes (servidor y cliente). Lo pasos para compilar el Pathchirp son muy parecidos a los de Pathload, debemos descomprimir el archivo pathchirp-2.4.1.tar.gz mediante el comando *tar –xvzf pathchirp-2.4.1.tar.gz* y tras la descompresión cambiarnos al directorio generado mediante el comando *cd parthchirp-2.4.1*, estando dentro, ejecutamos primero la instrucción *./configure* y finalmente tras su finalización ejecutar la instrucción *make*.

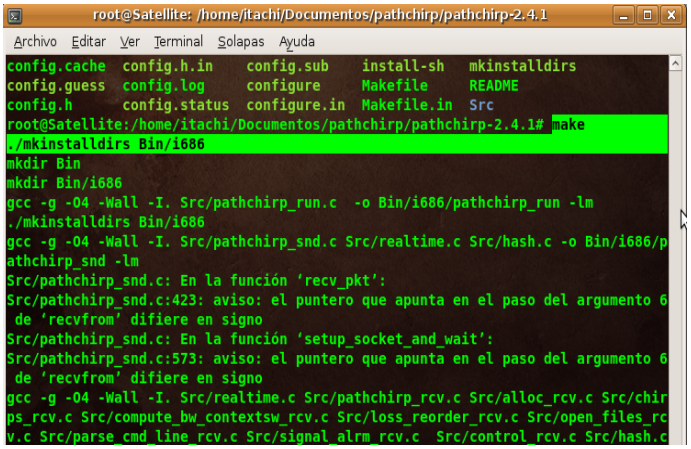

Figura 24: Compilación de la herramienta pathchirp

A diferencia de las otras herramientas como IGI-PTR y Pathload, Pathchirp utiliza tres aplicaciones fundamentales para poder operar. El pathchirp_snd, el pathchirp_rcv, y el pathchirp_run éste último es el que analiza la comunicación entre el cliente y el servidor; a diferencia del Pathload y el IGI-PTR en los cuales el cliente es el que generaba la información de salida.

Ni el cliente, ni el servidor desplegarán nada en pantalla y tampoco se generará tráfico hasta que pathchirp_run sea ejecutado. En ese momento el servidor y el cliente inician la comunicación y se empieza a generar información de salida.

Para poder ejecutar pathchirp_run debemos especificarle mediante los parámetros –R y –S cuales son los host que están ejecutando las aplicaciones pathchirp_rcv y pathchirp_snd respectivamente.

Cuando el pathchirp_run se ejecuta, además de iniciar los procesos de tráfico entre el cliente y servidor; se genera un archivo de salida con el nombre 10_192_1293823.instbw este archivo es llamado de esta forma por que las primeras caracteres indican los segmentos en donde se hace la prueba y las últimas hacen referencia a una marca de tiempo de inicio de la prueba.

Por alguna razón que aun no he podido determinar, este archivo permanece vacío antes y después de la ejecución de las aplicaciones, snd,rcv y run utilizando la topología antes descrita donde existe dos equipos en VLANs distintas y con diferentes rangos de direcciones IP. Sin embargo al ejecutar patchirp (snd,rcv y run) directamente sobre localhost el archivo si es generado.

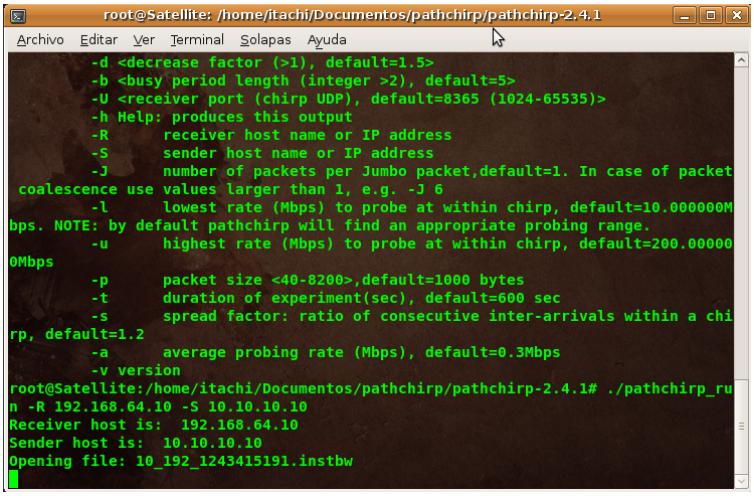

Figura 25:Pathchirp_run inicialización del servidor y el cliente

Conclusiones

Los resultados generados a partir de la aplicación de diversas herramientas de medición se ancho de banda brinda una pauta para futuros estudios de estimación en el consumo y disponibilidad del ancho de banda de las redes de paquetes, ya que al contar con una arquitectura de tráfico controlado, podemos ejecutar múltiples herramientas para la generación de tráfico y simular distintos ambientes en producción.

Uno de los puntos importantes es que la arquitectura diseñada para la generación de tráfico hace uso de dispositivos altamente empleados en el mercado como los son CISCO, con lo que podemos tener la confianza, que el esquema que estamos empleado está altamente soportado por los dispositivos y por muchas implementaciones en diferentes organizaciones. Al manejar protocolos abiertos como los son 802.11q para VLANs y OSPF para enrutamiento, se garantiza que la configuración e implementación pueda ser fácilmente difundida o migrada en equipos de otros proveedores.

Mediante la herramienta de generación de tráfico se pudieron generar distintas gráficas e información de resumen de la prueba, que me permitieron corroborar los valores configurados en los dispositivos y dar la pauta para iniciar con las pruebas para la estimación de ancho de banda disponible utilizando herramientas como IGI/PTR, Pathload y Pathchirp

En las gráficas generadas, se puede observar, entre otras cosas, que la pérdida de paquetes fue de 0.0. Por tal motivo podemos tener la certeza de que en estos ambientes controlados, no existe pérdida de paquetes, sin embargo; la tasa bits promedio de la transferencia varió considerablemente en un inicio;

pero posteriormente esta tasa permaneció constante durante un tiempo mayor en la transferencia. Así mismo los niveles de jitter esperados no fueron realmente del todo ideales ya que esperaba que en un ambiente de pruebas controlado como el que se desarrolló no existieran variaciones entre el retardo promedio, ya que únicamente se ejecutaron las herramientas para estimación de ancho de banda sobre la red, y ningún otro tipo de aplicaciones El retardo promedio tuvo un comportamiento similar al observado en la tasa de bits promedio, es decir con fuertes variaciones al inicio y retardos más constantes a lo largo de la transferencia.

Estas variaciones observadas tanto en la tasa de bits, el jitter y el retardo promedio de las transferencias, nos permiten cuestionarnos y analizar de manera más profunda dicho resultados, los cuales actualmente están sujetos a más análisis para poder determinar el origen de las dichas variaciones. Lo que puedo determinar de manera directa es que el ancho de banda puede variar incluso en un entorno de tráfico controlado ya sea por la carga de trabajo de los dispositivos, aunque cabe mencionar que en ningún momento se utilizaron comandos debug en ningún dispositivo puesto que esto generaría una mayor carga de trabajo de los mismos y probablemente tráfico adicional. Sin embargo podrían existir otros factores que podrían ser el tema principal de futuros trabajos de investigación para la estimación del ancho de banda.

Referencias

Alessio Botta, A. D. (2007). *Multi-protocol and multi-platform traffic generation and measurement.* Anchorage, Alaska: INFOCOM.

C. S. (1999). *Cisco IOS 12.0 Quality of Service.*Indianapolis: Cisco Press.

Garay, Ó. R. (2003). *enegi.org.mx*. Recuperado el 2008 de Marzo de 3, de http://www.inegi.gob.mx/inegi/contenidos/espanol/prensa/contenidos/articulos/tecnologia/dns03.pdf

Groth, D., & Skandier, T. (2005). *Guía del estudio de redes, cuarta edición.* Sybex, Inc.

masadelante.com. (24 de Enero de 2008). *masadelante.com*. Recuperado el 21 de Mayo de 2009, de http://www.masadelante.com/faq-ancho-de-banda.htm

Prasad, R., & Dovrolis, C. (26 de septiembre de 2003). “Bandwidth Estimation: Metrics, Measurement Techniques, and Tools.” . *IEEE Network* , 27-29.

Puentes, P. 2. (2008). *"Análisis del movimiento en la Red".* Mérida - Venezuela (CIDIAT).

UTN, S. F. (2008). *Conceptos básicos sobre redes*. recuperado el 3/02/2008 http://www.dednet.net/institucion/itba/cursos/000183/demo/biblioteca/121redesUTN.pdf

D-ITG, Universita' degli Studi di Napoli "Federico II". Alessio Botta, Alberto Dainotti, Antonio Pescapè, "Multi-protocol and multi-platform traffic generation

and measurement", INFOCOM 2007 DEMO Session, May 2007, Anchorage (Alaska, USA).

IGI/PTR site. www.cs.cmu.edu, consultado 20 de noviembre de 2008, disponible en línea en: http://www.cs.cmu.edu/~hnn/igi/#Document.

Pathload site. www.cc.gatech.edu, consultado 23 de noviembre de 2008, disponible en línea en: http://www.cc.gatech.edu/fac/Constantinos.Dovrolis/bw-est/pathload_tutorial.html.

Pathchirp site. www.spin.rice.edu, consultado 28 de noviembre de 2008, disponible en línea en: http://www.spin.rice.edu/Software/pathChirp/.

Anexos

Especificaciones de los dispositivos de pruebas

Tabla 1: Especificación del switch de prueba

Switch CISCO 2960 de 24 puertos	
Nivel en topología de red:	Acceso
Número y tipo de puertos:	24 Puertos 10/100 BASETX, 2 AUX 2x10/100/1000Base-T/SFP
Velocidad Mpps:	10.1 Mpps
Capa del modelo OSI:	Capa 2
Tipo de procesamiento:	Store and Forward
Tecnologías LAN soportada:	1000BASE-SX, 100BASE-LX, 10BASE-T, 10/100/100BASE-TX
Medios soportados:	Cable, Fibra
Cantidad de memoria :	64 MB DRAM, 32 MB flash memory
Soporte:	CCNA1, CCNA2,CCNA3 y CCNA4
Soporte para MACs:	Arriba 8000 MAC addresses
Soporte VLANs:	32 VLANs (1000 range)
MTBF:	243,595 Horas

Tabla 2: Especificaciones de los routers de prueba

Router CISCO serie 1800	
Número y tipo de puertos:	2 -10/100 Mbits, 2 AUX ports
Velocidad de procesamiento:	1.6 Gbps
Capa del modelo OSI:	Capa 3
Tecnología LAN soportada:	Ethernet, FDDI
Medios soportados:	Ethernet, Fibra óptica(modulo adicional)
Protocolos soportados:	SNMP, TCP/IP,SDM, RIP2, OSPF,IGRP, EIGRP
Cantidad de memoria:	Compact Flash 62 MB.
Soporte de fuentes de poder redundante:	External only, connector for RPS provided by default
Soporte de características QoS:	1 año de garantía
Soporte para aplicaciones multicast:	Si
Soporte:	CCNA1, CCNA2, CCN3, CCNA4

Tabla 3: Especificaciones del Host A de pruebas

Laptop TOSHIBA Satellite A135-SP4116	
Procesador	Intel Core® Duo T2350 @ 1.866 GHz
Memoria	2 GB en DDR2 @ 667 MHz
Disco duro	80 GB en disco duro
Sistema operativo	Microsoft Windows Vista Ultimate SP1
Tarjeta de red	RealtekRTL-8101 PCI-E FastEthernet

Tabla 4: Especificaciones del host B de pruebas

Equipo de escritorio (Desktop) Compaq Evo	
Procesador	Intel Pentium 4 CPU @ 1.6 GHz
Memoria	1 GB en SDRAM @ 133 MHz
Disco duro	80 GB en disco duro
Sistema operativo	Microsoft Windows XP Profesional
Tarjeta de red	Intel FastEthernet

Configuración de los dispositivos de red

Switch:

Hostname SW0

enable secret 654321

enable password 123456

no ip domain-lookup

interfaceFastEthernet0/1 <┘ switchport mode trunk

interfaceFastEthernet0/13 <┘ switchport mode trunk

interfaceFastEthernet0/2 <┘ switchport access vlan 10

interfaceFastEthernet0/3 <┘ switchport access vlan 10

interfaceFastEthernet0/4 <┘ switchport access vlan 10

interfaceFastEthernet0/5 <┘ switchport access vlan 10

interfaceFastEthernet0/6 <┘ switchport access vlan 20

interfaceFastEthernet0/7 <┘ switchport access vlan 20
interfaceFastEthernet0/8 <┘ switchport access vlan 20
interfaceFastEthernet0/9 <┘ switchport access vlan 20
interfaceFastEthernet0/10 <┘ switchport access vlan 20
interfaceFastEthernet0/11 <┘ switchport access vlan 20
interfaceFastEthernet0/12 <┘ switchport access vlan 20
interfaceFastEthernet0/14 <┘ switchport access vlan 30
interfaceFastEthernet0/15 <┘ switchport access vlan 30
interfaceFastEthernet0/16 <┘ switchport access vlan 30
interfaceFastEthernet0/17 <┘ switchport access vlan 30
interfaceFastEthernet0/18 <┘ switchport access vlan 30
interfaceFastEthernet0/19 <┘ switchport access vlan 40
interfaceFastEthernet0/20 <┘ switchport access vlan 40
interfaceFastEthernet0/21 <┘ switchport access vlan 40
interfaceFastEthernet0/22 <┘ switchport access vlan 40
interfaceFastEthernet0/23 <┘ switchport access vlan 40
interfaceFastEthernet0/24 <┘ switchport access vlan 40
linevty 0 4
passwordcisco
login

RouterDCE:

hostname R1-DCE
enable secret 654321
enable password 123456
no ip domain lookup
interface FastEthernet 0/0
no shutdown
interface FastEthernet 0/0.1
encapsulationdot1Q 10

ip address 10.10.10.1 255.255.255.0
no snmp trap link-status
interface FastEthernet 0/0.2
encapsulation dot1Q 30
ip address 192.168.32.1 255.255.255.0
no snmp trap link-status
interface Serial 0/0/0
ip address 200.33.146.1 255.255.255.252
clockrate 128000
router ospf 1
log-adjacency-changes
network 10.10.10.0 0.0.0.255 area 0
network 192.168.32.1 0.0.0.255 area 0
network 200.33.146.1 0.0.0.3 area 0
ip classless
line vty 0 4
password cisco
login
end

RouterDTE:

hostname R1-DCE
enable secret 654321
enable password 123456
no ip domain lookup
interface FastEthernet 0/0
no shutdown
interface FastEthernet 0/0.1
encapsulation dot1Q 10
ip address 10.10.10.1 255.255.255.0

interface FastEthernet 0/0.2
encapsulation dot1Q 30
ip address 192.168.32.1 255.255.255.0
interface Seria l0/0/0
ip address 200.33.146.1 255.255.255.252
clockrate 128000
router ospf 1
log-adjacency-changes
network 10.10.10.0 0.0.0.255 area 0
network 192.168.32.0 0.0.0.255 area 0
network 200.33.146.0 0.0.0.3 area 0
ip classless
line vty 0 4
password cisco
login
end

Configuración del Host de pruebas A

Dirección IP: 10.10.10.5
Máscara de subred: 255.255.255.0
Puerta de enlace: 10.10.10.254
Puerto #5 del Switch

Configuración del Host de pruebas B

Dirección IP: 192.168.64.5
Máscara: 255.255.255.0
Puerta de enlace: 192.168.64.254
Puerto #23 del Switch

Desarrollo de los Scripts para generación y graficas de datos

Script: plot.cmd

```
@echo off
octave -qf E:\proyecto1\d-itg\bin\ITGplot.m %1
```

Script genData.cmd

```
@ECHO OFF
itgdec %1
itgdec %1 -o octaveFile
itgdec %1 -b %2
itgdec %1 -d %2
itgdec %1 -j %2
itgdec %1 -p %2
echo Graficando BitRate.dat para %1
cmd /c plot .\bitrate.dat
echo Graficando Jitter.dat para %1
cmd /c plot .\jitter.dat
echo Graficando Delay.dat para %1
cmd /c plot .\delay.dat
echo Graficando PacketLoss.dat para %1
cmd /c plot .\packetloss.dat
pause
```

Sistemas operativos y software para la estimación de ancho de banda.

Tabla 5: Especificaciones de los sistemas operativos y software de pruebas

Sistemas operativos y virtualización	
Sistema operativo base	Microsoft Windows Vista
Versión de service pack	Service Pack 1
Sistema operativo virtual	Ubuntu GNU/Linux
Versión	8.10
Maquina virtual	VMWare Workstation
Versión	6.5

Tabla 6: Detalles del software de estimación de ancho de banda

Software de estimación de ancho de banda	
Nombre de la herramienta 1	IGI-PTR
Home page	www.cs.cmu.edu/~hnn/igi
Tipo de algoritmo	Initial Gap Increasing
Prueba que realiza	Ancho de banda disponible de extremo a extremo
Nombre de la herramienta 2	Pathload
Home page	http://www.cc.gatech.edu/fac /pathload.html
Tipo de algoritmo	The link with the minimum capacity
Prueba que realiza	Ancho de banda disponible de extremo a extremo
Nombre de la herramienta 3	Pathchirp
Home page	http://www.spin.rice.edu/Software/pathChirp
Tipo de algoritmo	Self-induced congestion
Prueba que realiza	Ancho de banda disponible de extremo a extremo