

**Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación**

Lecturas en Ciencias de la Computación
ISSN 1316-6239

**Análisis Comparativo de
Emuladores WAN**

Prof. Karima Velásquez
Prof. Eric Gamess

RT-2013-01

Análisis Comparativo de Emuladores WAN

Karima Velásquez* y Eric Gamess†

RESUMEN

La evaluación de aplicaciones y protocolos de red en un ambiente WAN real puede ser difícil y compleja, debido a los costos y las distancias involucradas. Por lo tanto, muchos investigadores y desarrolladores usan simuladores de red o ambientes de prueba controlados donde la WAN es emulada sobre tecnologías LAN más comúnmente usadas, como Ethernet. En este reporte se presenta una revisión actualizada de herramientas de emulación WAN. Se discuten sus principales características y los parámetros de red que permiten configurar y modificar. Adicionalmente se presentan los resultados de varios experimentos realizados con estas herramientas, usando tanto IPv4 como IPv6, para verificar su precisión y analizar los resultados obtenidos.

1. INTRODUCCIÓN

Es común que investigadores y desarrolladores necesiten probar diferentes aplicaciones y protocolos de red que deben ser implementados sobre un ambiente WA. Para ofrecer resultados más realistas, la mejor solución sería evaluar la aplicación o protocolo sobre un escenario real. Sin embargo, esto no siempre es posible ya que la red a usar puede estar en producción, y configurar un ambiente de pruebas resulta costoso y complejo. Soluciones alternativas deben ser consideradas, como simulación [5], software de emulación de redes WAN, y uso de dispositivos tun/tap para emulación de características de red. Es importante que la solución alternativa modele los enlaces WAN con sus características típicas, como retardo, jitter, tasa de pérdida de paquetes, limitaciones de ancho de banda, reordenamiento de paquetes, etc.

Este reporte está enfocado en las herramientas de software para la emulación de comportamiento de enlaces WAN usando tecnologías baratas y ampliamente aceptadas, como Ethernet. Existen varias herramientas disponibles con este propósito, pero no todas presentan las mismas características ni proveen los mismos parámetros de red configurables. En este trabajo se presenta una revisión actualizada de emuladores WAN, a manera de ayudar a los investigadores y desarrolladores al momento de seleccionar la herramienta más apropiada para sus necesidades. Se eligieron seis herramientas (NetEm, WANem, NIST Net, Dummynet, WANSim, y VMware Workstation) por su popularidad y disponibilidad. Las herramientas se discuten a continuación.

2. HERRAMIENTAS DE EMULACIÓN WAN

Los emuladores WAN fueron diseñados para permitir la realización de experimentos controlados y reproducibles, sobre un escenario sencillo de laboratorio. En otras palabras, estas herramientas permiten que un sencillo PC multitarjeta (haciendo el papel de un router) emule varios escenarios más complejos, modificando diversos parámetros de red, como retardo, pérdida, reordenamiento, y duplicación de paquetes, así como también limitar el ancho de banda disponible.

2.1. NETEM

NetEm¹ provee funcionalidades de emulación de ambientes WAN al permitir al usuario configurar varios parámetros de red sobre un enlace [6]. Las reglas que se establecen con NetEm se aplican al tráfico saliente de la tarjeta de red. La herramienta se configura vía consola, no se ofrece una GUI. NetEm tiene soporte para redes IPv4 e IPv6.

La documentación para esta herramienta es escasa. No existe un manual oficial disponible. Sin embargo, la herramienta es sencilla de usar, así que la curva de aprendizaje no es muy larga. También, existen varias guías

* Laboratorio de Comunicación y Redes (LACORE) – karima.velasquez@ciens.ucv.ve

† Laboratorio de Comunicación y Redes (LACORE) – egamess@gmail.com

¹ <http://www.linuxfoundation.org/collaborate/workgroups/networking/netem>

disponibles en Internet (provistas por usuarios entusiastas) que muestran ejemplos prácticos para la realización de experimentos sencillos.

2.1.1. PLATAFORMAS SOPORTADAS

NetEm se encuentra disponible en las distribuciones Linux más comunes (e.g. Fedora, OpenSuse, Gentoo, Debian, Mandriva, Ubuntu) con un kernel 2.6 o superior. No requiere instalación de paquetes adicionales; esta herramienta viene preinstalada con el sistema operativo en su versión más básica.

NetEm es controlada por la herramienta de línea de comando tc (traffic control), que manipula la configuración del tráfico. tc es parte del paquete iproute2, que usa bibliotecas compartidas y los archivos de datos del directorio /usr/lib/tc.

2.1.2. PARÁMETROS DE RED CONFIGURABLES

NetEm permite al usuario definir varios parámetros de red, como retardo, tasa de pérdida, y duplicación y reordenamiento de paquetes. En el caso del retardo, es posible usar variables aleatorias (e.g. Normal y Pareto) para su emulación. Esto resulta conveniente ya que usualmente el retardo en la red no es uniforme, por lo que los experimentos pueden ser más realistas. Para la pérdida, reordenamiento, y duplicación de paquetes, también existe la posibilidad de asignar una correlación que permita emular ráfagas.

No existe la posibilidad de emular limitación de ancho de banda directamente con NetEm, pero es posible usar otras disciplinas de tc para hacer control de tasa de bits. Una alternativa es usar el filtro Token Bucket para limitar la salida del tráfico.

2.2. WANEM

WANem [7] provee emulación de varias características de redes WAN, como retardo, tasa de pérdida, y corrupción de paquetes. WANem permite una configuración de ambiente de pruebas rápida y sencilla, a través de una interfaz web (ver Figura 1). Esta herramienta es desarrollada en Mumbai, India, por el Performance Engineering Research Centre, y según el autor, se distribuye bajo la licencia GNU General Public License. Sin embargo, no se pudo ubicar el código fuente en su sitio web². Los parámetros configurados con WANem afectan el tráfico IPv4 e IPv6. A diferencia de NetEm, WANem afecta al tráfico de entrada y de salida. Existe documentación disponible en el sitio web oficial, incluyendo una Guía de Configuración, una Guía de Usuario, y una Guía de Troubleshooting.

The screenshot shows the WANem web interface. At the top, it says 'TATA CONSULTANCY SERVICES' and 'WANem The Wide Area Network Emulator'. Below that, there are navigation links: 'About', 'WANalyzer', 'Basic Mode', 'Advanced Mode', 'Save/Restore', and 'Help'. A message states 'All WANem values have been reset'. The main configuration area is for 'Interface: eth0'. It includes a 'Packet Limit' of 1000 (Default=1000) and a 'Symmetrical Network' dropdown set to 'Yes'. There are sections for 'Bandwidth' (Choose BW, Other), 'Delay' (Delay time(ms), Jitter(ms), Correlation(%)), 'Loss' (Loss(%), Correlation(%)), 'Duplication' (Duplication(%), Correlation(%)), 'Packet reordering' (Reordering(%), Correlation(%), Gap(packets)), and 'Corruption' (Corruption(%)). There are also sections for 'Idle timer Disconnect', 'Random Disconnect', and 'Random connection Disconnect', each with 'Type' and 'MTTF' settings. At the bottom, there are fields for 'IP source address', 'IP source subnet', 'IP dest address', 'IP dest subnet', and 'Application port if any'. Control buttons include 'Add a rule set', 'Apply settings', 'Reset settings', 'Refresh settings', 'Display commands only, do not execute them', and 'Check current status'.

Figura 1: Interfaz Web de WANem

² <http://wanem.sourceforge.net>

2.2.1. PLATAFORMAS SOPORTADAS

WANem es distribuida como una aplicación embebida en Knoppix³, una distribución GNU/Linux basada en Debian que se ejecuta completamente desde un CD o DVD. Existen dos posibilidades para descargar WANem desde el sitio oficial: (1) una imagen ISO de Knoppix para crear un live CD arrancable; y (2) una máquina virtual (compatible con VMWare). Ambas opciones ayudan al usuario a acelerar la ejecución de sus experimentos. Ya que no se pudo encontrar el código fuente de la herramienta, resultó imposible probarla sobre otra distribución de Linux.

2.2.2. PARÁMETROS DE RED CONFIGURABLES

Las características de red controladas por WANem son pérdida de paquetes, retardo (fijo y variable), duplicación, reordenamiento, corrupción de paquetes, limitación de ancho de banda, y desconexión. La configuración del retardo, pérdida y duplicación de paquetes permite agregar una correlación para recrear experimentos más realistas. Se soportan tres tipos de desconexión: (1) desconexión por temporizador de inactividad, (2) desconexión aleatoria, y (3) desconexión de una conexión aleatoria.

2.3. NIST NET

NIST Net⁴ es un emulador de redes WAN desarrollado por el National Institute of Standard and Technology (NIST). Al operar al nivel de IP, NIST Net puede emular las características críticas impuestas por varias redes de área amplia (como congestión y pérdida) así como también por varias tecnologías de capas inferiores.

NIST Net trabaja a través de una tabla de entradas de emulador. Cada entrada consiste de tres partes: (1) un conjunto de especificaciones acerca de qué paquetes coinciden con esta entrada, (2) un conjunto de efectos a aplicar a los paquetes que coincidan, y (3) un conjunto de estadísticas que se mantienen acerca de los paquetes que han coincidido con esta entrada. NIST Net trata a cada flujo (conjunto de paquetes coincidentes) de manera separada. Miles de entradas pueden ser cargadas simultáneamente, cada una con reglas diferentes. Esto permite la emulación de un conjunto diverso de características de red en un escenario relativamente pequeño (dos o tres PCs). NIST Net no permite la configuración de reglas para direcciones IPv6; y estas reglas sólo afectan el tráfico entrante [3].

Aunque el sitio oficial incluye documentación para la instalación y el uso de la herramienta, esta no es muy detallada. Adicionalmente, se incluyen algunos archivos README en el directorio principal de la aplicación.

2.3.1. PLATAFORMAS SOPORTADAS

NIST Net es implementado como un módulo del kernel de Linux, y posee una interfaz basada en X Windows (ver Figura 2). La interfaz gráfica permite al usuario seleccionar y monitorear flujos específicos de tráfico que pasan a través del router, y aplicar efectos particulares a los paquetes IP de dichos flujos.

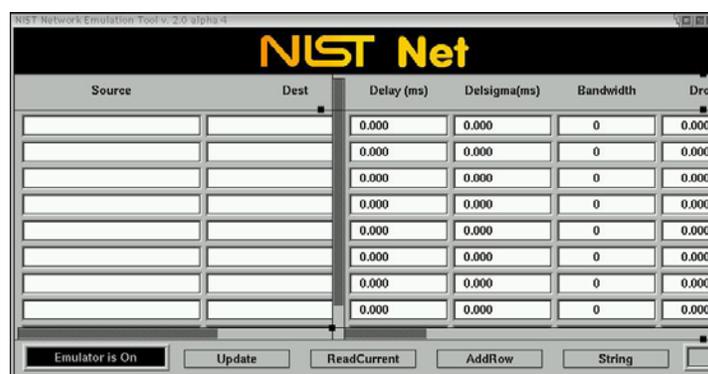


Figura 2: Interfaz Gráfica de NIST Net

³ <http://knoppix.net>

⁴ <http://snad.ncsl.nist.gov/nistnet>

Durante la instalación de esta herramienta, se enfrentaron algunos inconvenientes. Existen dos versiones disponibles de NIST Net en el sitio web: 2.0.12b para kernels 2.4, y 2.0.12c para kernels 2.6. Para la versión 2.0.12c, se intentó la instalación en un kernel versión 2.6.26-1 (Debian Lenny), pero surgieron algunos problemas con algunas bibliotecas al momento de la compilación, lo que evitó completar el proceso de instalación. Finalmente, se logró compilar la versión 2.0.12b de NIST Net usando el kernel versión 2.4.27-3 (Debian Sarge). Como NIST Net está implementado como un módulo del kernel, se requiere que se cargue el módulo antes de realizar los experimentos. Para cargar el módulo, se provee un script (`./Load.Nistnet`) en el directorio principal de la herramienta una vez instalada.

2.3.2. PARÁMETROS DE RED CONFIGURABLES

El conjunto de parámetros configurables incluye: retardo, tasa de pérdida, reordenamiento y duplicación de paquetes, así como limitación de ancho de banda.

2.4. DUMMYNET

Dummynet⁵ es desarrollado por investigadores de la Universidad de Pisa, Italia. Fue diseñado originalmente para probar protocolos de red, y desde entonces ha sido usado para una variedad de aplicaciones incluyendo manejo de ancho de banda [8]. Simula limitaciones de ancho de banda, retardo, tasa de pérdida, y efecto multicamino. Además, implementa una variante de Colas Justas Pesadas (Weighted Fair Queueing - WFQ) llamada WF2Q+. Puede usarse en estaciones de trabajo o en PCs multitarjeta actuando como routers [2].

Dummynet trabaja interceptando los paquetes (seleccionados por colas de ipfw, uno de los firewalls de los sistemas FreeBSD) en su paso a través de la pila de protocolos, con lo que logra simular los efectos de limitación de ancho de banda, retardo de propagación, colas de espacio limitado, pérdida de paquetes, y efecto multicamino. Las reglas de Dummynet aplican al tráfico IPv4 e IPv6, al tráfico tanto entrante como saliente de la NIC (Network Interface Card).

En el sitio web oficial, existen varios artículos que documentan las características principales de Dummynet. Sin embargo, no existen manuales que puedan guiar al usuario en el uso de la herramienta.

2.4.1. PLATAFORMAS SOPORTADAS

Dummynet es una aplicación estándar en sistemas FreeBSD y Mac OS X, por lo que en cualquiera de estos sistemas sólo se debe cargar el módulo del kernel apropiado (`ipfw.ko` y `dummynet.ko`). También es posible encontrar versiones de Dummynet para otros sistemas operativos, como Linux y Windows. Para la evaluación de Dummynet, se usó la versión de FreeBSD.

2.4.2. PARÁMETROS DE RED CONFIGURABLES

Los siguientes parámetros pueden ser configurados: limitación de ancho de banda, tamaño de la cola, retardo, y tasa de pérdida. Es posible configurar máscaras para que las limitaciones especificadas se apliquen a diferentes flujos. Otra característica interesante es el soporte de múltiples caminos entre el emisor y el servidor. Esto se logra haciendo match probabilísticos.

2.5. WANSIM

WANSim⁶ permite al usuario modificar el ancho de banda, retardo, jitter, tasa de pérdida, y duplicación y corrupción de paquetes. WANSim es usualmente configurado como un puente que se establece entre dos NICs en tiempo de arranque. WANSim utiliza la herramienta previamente descrita, NetEm, para alcanzar su objetivo, actuando como un script que puede ser accedido desde una consola de Linux. Ya que depende de NetEm para la emulación de características WAN, y NetEm afecta tanto el tráfico IPv4 como el IPv6, WANSim también debe soportar ambos protocolos de red. Sin embargo, esto no pudo ser corroborado a través de experimentos; ya que los resultados arrojados no estaban dentro de un umbral aceptable. De hecho, los

⁵ <http://info.iet.unipi.it/~luigi/dummynet>

⁶ <http://code.google.com/p/wansim>

resultados eran equivalentes a no tener un emulador WAN configurado en el camino. Este problema pudo ser causado por un error de configuración, o por un error de la herramienta.

Existe una guía de instalación paso-a-paso y una guía de configuración en el sitio web, así como también una sección de preguntas frecuentes (FAQ).

2.5.1. PLATAFORMAS SOPORTADAS

WANsim trabaja sobre plataformas Linux, usando la funcionalidad del kernel descrita en la Sección 2.1, NetEm.

2.5.2. PARÁMETROS DE RED CONFIGURABLES

Los parámetros de red afectados por WANsim son retardo, jitter, tasa de pérdida, así como duplicación y corrupción de paquetes. Adicionalmente, configurando un retardo y un jitter altos, causará el envío de paquetes fuera de orden, permitiendo la emulación de reordenamiento de paquetes.

2.6. VMWARE

VMware⁷ es una compañía que provee soluciones de virtualización. Ofrecen muchos productos, que incluyen el famoso ESX, VMware Player, y VMware Workstation. Desde el lanzamiento de la versión 8 de VMware Workstation, es posible usar opciones avanzadas de virtualización en la configuración de adaptadores de red, para limitar el ancho de banda y especificar una tasa de pérdida para el tráfico entrante y saliente de una máquina virtual, convirtiéndose así en otra opción para la emulación de características específicas de una WAN. El procedimiento para lograr este objetivo es bastante sencillo. El usuario debe seleccionar la opción VM>Settings en la pestaña Hardware, elegir el adaptador de red y hacer clic en la opción Advanced, y allí especificar la limitación de ancho de banda y porcentaje de tasa de pérdida para el tráfico entrante y/o saliente. Finalmente se debe presionar OK para salvar los cambios (ver Figura 3). Este procedimiento está documentado en [10].

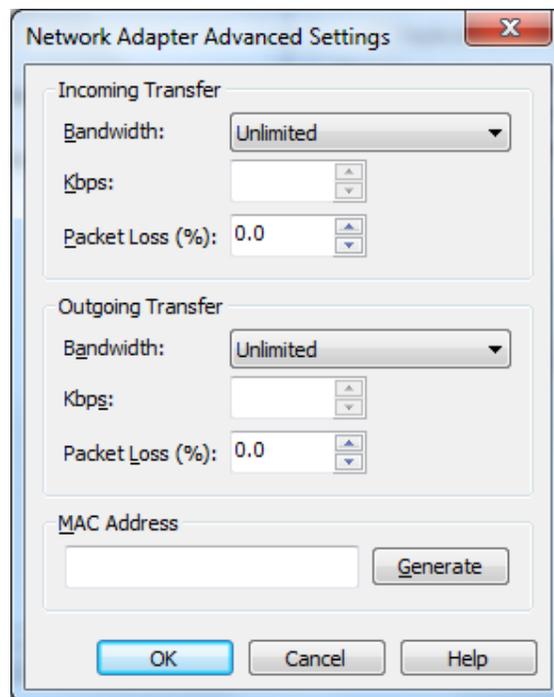


Figura 3: Configuración con VMware

⁷ <http://www.vmware.com>

2.6.1. PLATAFORMAS SOPORTADAS

VMware Workstation corre sobre Windows, Linux, y Mac OS X como host. Los parámetros de red configurados en la máquina virtual (guest) tendrán efecto sobre cualquier sistema operativo anfitrión.

2.6.2. PARÁMETROS DE RED CONFIGURABLES

Como se mencionó previamente, los únicos parámetros de red configurables son el ancho de banda y la tasa de pérdida. Para el ancho de banda existen varias opciones preconfiguradas (e.g. T1, T3, etc.) para facilitar la elección del usuario. La tasa de pérdida debe ser especificada por el usuario. El valor por defecto es 0.0%.

3. ANÁLISIS COMPARATIVO

La Tabla 1 muestra un resumen de las características principales de los emuladores WAN estudiados. La fila Versión indica la versión evaluada. Última Actualización especifica la fecha de lanzamiento de la última versión del software, que brinda una idea de qué tan activo es el proyecto. Las otras características fueron explicadas con mayor detalle en la descripción de cada herramienta. Ya que VMware es un software comercial, es actualizado constantemente. Por otro lado, hay proyectos que parecen no estar mantenidos por la comunidad (e.g. NIST Net no muestra actividad por más de 7 años).

Tabla 1: Características Principales de los Emuladores WAN Evaluados

Característica	NetEm	WANem	NIST Net	Dummysnet	WANsim	VMware
Versión	N/A	2.3	2.0.12b	1.10	2.0.0-1	Workstation 8.0.2
Última Actualización	Octubre 2010 ⁸	Marzo 2011	Julio 2005	Febrero 2010	Mayo 2011	Mayo 2012
Punto de Intercepción	Salida	Entrada y Salida	Entrada	Entrada y Salida	Salida	Entrada y Salida
Soporte IPv6	Sí	Sí	No	Sí	N/A	Sí
Plataforma	Incluido en Linux 2.6	Knoppix	Disponible para Linux 2.4 y 2.6	Incluido en FreeBSD	Disponible para Linux 2.6	Disponible para Windows, Linux, y Mac OS
Interfaz de Usuario	Consola	GUI	GUI y consola	Consola	Consola	GUI
Limitación de Ancho de Banda	Sí	Sí	Sí	Sí	Sí	Sí
Tasa de Pérdida	Sí	Sí	Sí	Sí	Sí	Sí
Retardo	Sí	Sí	Sí	Sí	Sí	No
Reordenamiento de Paquetes	Sí	Sí	Sí	No	Sí	No
Corrupción de Paquetes	Sí	Sí	Sí	No	Sí	No
Duplicación de Paquetes	Sí	Sí	Sí	No	Sí	No

4. EXPERIMENTOS REALIZADOS

Para los experimentos se emplearon dos tipos de PCs:

- PC de tipo H: Generadores y medidores de tráfico. Esto quiere decir que son los hosts que generan el tráfico de paquetes y realizan la medición de throughput, retardo, tasa de pérdida, etc. Ambos PCs de este tipo eran idénticos, equipados con un procesador Intel Core 2 Duo E6750 (2.66 GHz), 4 GB de memoria RAM, 250 GB de disco duro, y un adaptador Ethernet Broadcom 10/100/1000 Mbps. En cada PC de tipo H se instaló Debian 6.0.5 (Squeeze).

⁸ Fecha de la última actualización del kernel

- PC de tipo R: Se usó un único PC de este tipo. A nivel de hardware, era idéntico a los PCs de tipo H, pero con dos adaptadores Ethernet adicionales (Intel PRO/100 S). En este PC se realizaron cuatro particiones en el disco duro y se instalaron los sistemas operativos Debian 6.0.5 (Squeeze), Debian 3.1 r8 (Sarge), FreeBSD 9.0, y Windows 7. Para Knoppix, sólo fue necesario iniciar el PC desde el *live CD* descargado del sitio web de WANem.

La configuración es mostrada en la Figura 4. Para el ambiente de máquinas virtuales se usó Windows 7 como sistema operativo *host*, y Debian 6.0.5 (Squeeze) en la máquina virtual.

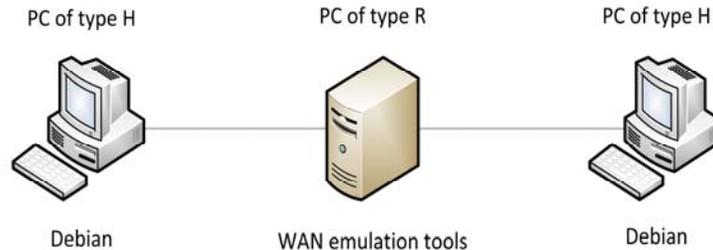


Figura 4: Escenario de Pruebas

Se realizaron mediciones antes de fijar cualquier regla en los emuladores WAN, para poder comparar los efectos del uso de estas herramientas. Luego, para cada herramienta, se realizaron una serie de experimentos midiendo tres parámetros de red: (1) tasa de pérdida de paquetes, (2) ancho de banda, y (3) retardo. Para cada parámetro de red se probaron varios valores, listados en la Tabla 2. Para medir la tasa de pérdida y el ancho de banda se usó D-ITG [1], una plataforma de generación de tráfico a nivel de paquetes de código abierto para producir tráfico IPv4 e IPv6. Algunos valores de retardo reportados por D-ITG no son confiables, como se discute en [9]. Para evitar este inconveniente, se realizó una herramienta de benchmarking propia usando el lenguaje de programación C para medir el retardo de una vía (OWD, One Way Delay) de un paquete enviado entre los dos PCs de tipo H (ver Figura 4). Los benchmarks son basados en el paradigma cliente/servidor. Básicamente, un paquete (IPv4 o IPv6) de tamaño fijo se intercambia entre el cliente y el servidor un número de veces (definido por el usuario). Se toma una marca de tiempo antes y después del intercambio. La diferencia de las marcas de tiempo se divide por el número de veces que el paquete se intercambia para obtener el RTT promedio del enlace, y se divide entre dos (2) para obtener el retardo.

Tabla 2: Parámetros Incluidos en los Experimentos

Parámetro de Red	Valor
Tasa de Pérdida de Paquetes	5%
	10%
	20%
	50%
Ancho de Banda	E1 (2.048 Mbps)
	20 Mbps
	E3 (34.368 Mbps)
	50 Mbps
Retardo	5 ms
	10 ms
	20 ms
	50 ms
	100 ms

Se configuró D-ITG para generar tráfico CBR de 9000 paquetes por segundo, con una carga útil UDP de 1200 bytes para evitar fragmentación. La herramienta desarrollada fue configurada para enviar y recibir un total de 1000 paquetes con una carga útil UDP de 1200 bytes. Los experimentos fueron repetidos varias veces para obtener resultados consistentes. El promedio es reportado como resultado.

Luego de realizar los experimentos, se notó que los resultados reportados por WANsim no estaban dentro de los valores esperados según los parámetros configurados. Esto podría ser consecuencia de una mala configuración o de una falla en la herramienta. Por este motivo no se incluyó a WANsim en los resultados reportados.

La Figura 5 muestra los resultados de los experimentos de medición de tasa de pérdida de paquetes para IPv4. VMWare fue la herramienta más precisa, seguida por NetEm, WANem y NIST Net. Dummynet mostró buen comportamiento para pequeñas tasas de pérdida, pero se alejó de los resultados esperados para tasas más altas.

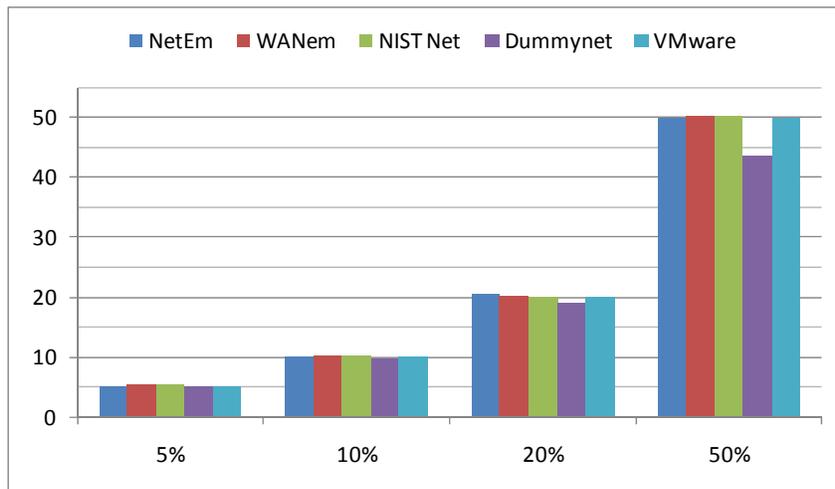


Figura 5: Experimentos de Tasa de Pérdida con IPv4

En la Figura 6 se despliegan los resultados para los experimentos de ancho de banda usando IPv4. Es notable que todas las herramientas mostraran un resultado similar, siendo NetEm el más cercano a los parámetros configurados.

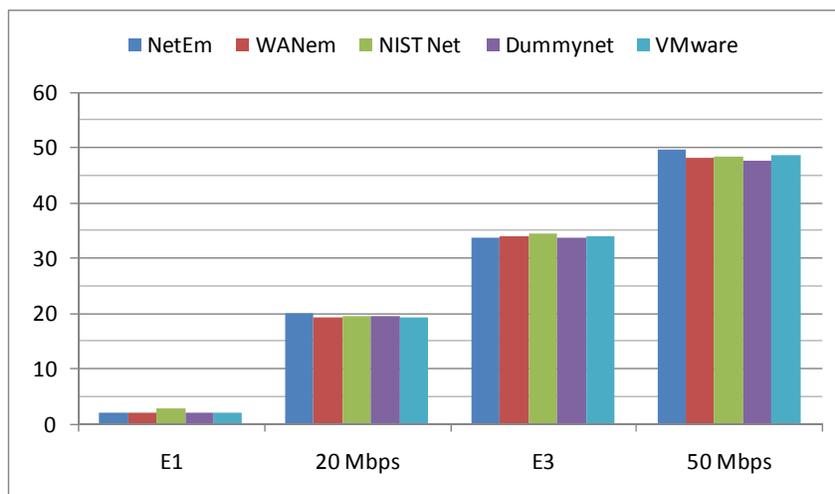


Figura 6: Experimentos de Ancho de Banda con IPv4

Para los experimentos de retardo no se incluyó VMWare, ya que no tiene soporte para emular este parámetro de red. La Figura 7 muestra los resultados obtenidos con IPv4. Como se especifica en la Tabla 2, se configuraron diferentes valores de retardo que incluyen 5 ms, 10 ms, 20 ms, 50 ms, y 100 ms. Se puede observar que NetEm, NIST Net y Dummysnet muestran resultados similares. Sin embargo, WANem arroja retardos mayores. Esto puede ser ocasionado porque la herramienta no está directamente instalada sobre el disco duro, sino que se ejecuta sobre un *live CD* de Knoppix, que puede agregar cierta latencia al proceso de reenvío y eventualmente demorando la entrega. Los resultados más precisos fueron arrojados por Dummysnet, aunque la diferencia con NetEm y NIST Net estuvo en el orden de los microsegundos.

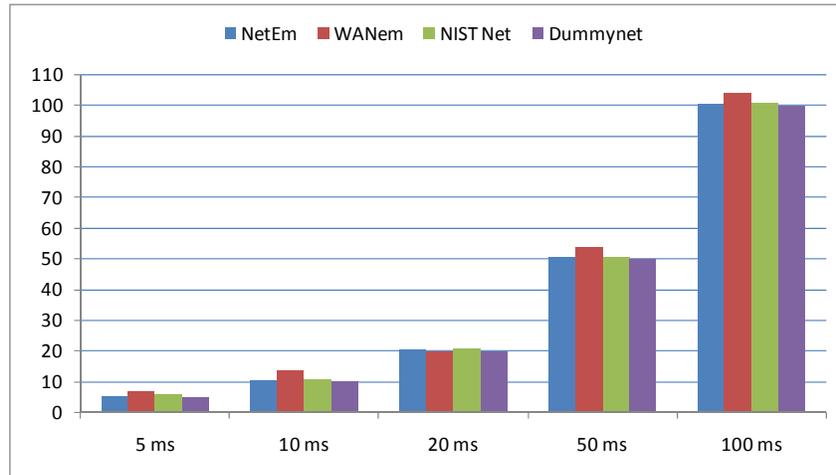


Figura 7: Experimentos de Retardo con IPv4

Se repitió cada prueba esta vez usando IPv6 como protocolo de red. En estos experimentos, se excluyó WANSim por los mismos motivos que fue excluido de las pruebas con IPv4 (una posible falla en la configuración o en la herramienta). Además, NIST Net también fue eliminada de estos experimentos, ya que no ofrece soporte de tráfico IPv6. La última versión de NIST Net (2.0.12b) fue liberada en julio del 2005, cuando la escasez de direcciones IPv4 no era un problema tan grave como lo es hoy en día, lo que debe ser la razón principal para la falta de soporte de IPv6.

La Figura 8 refleja los resultados de los experimentos de tasa de pérdida para IPv6. Se puede ver que aunque se ofrecen buenos resultados para tasas de pérdida pequeñas, Dummysnet no es muy preciso para mayores tasas de pérdida. Nuevamente VMWare fue el más cercano a los valores esperados. Este comportamiento es consistente al obtenido con IPv4.

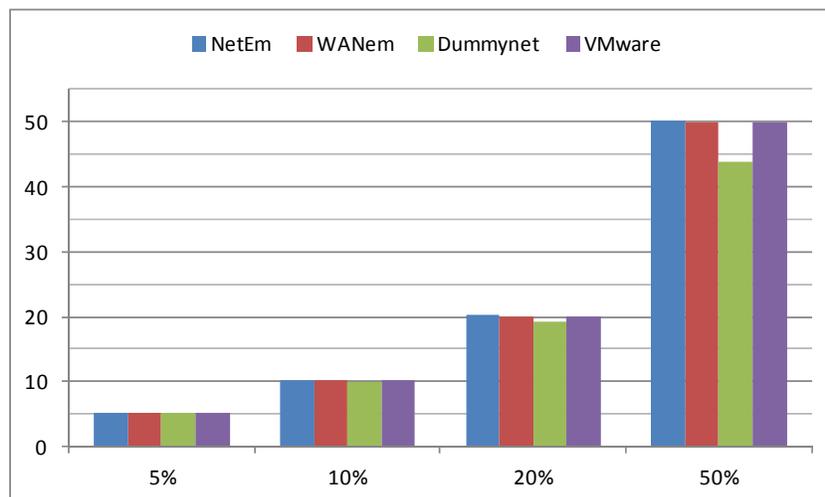


Figura 8: Experimentos de Tasa de Pérdida con IPv6

La Figura 9 muestra los resultados de los experimentos de ancho de banda para IPv6. Como con IPv4, todas las herramientas muestran resultados similares. En este caso, WANem y VMWare reportaron los valores más cercanos a los parámetros configurados, seguidos de cerca por NetEm y DummyNet.

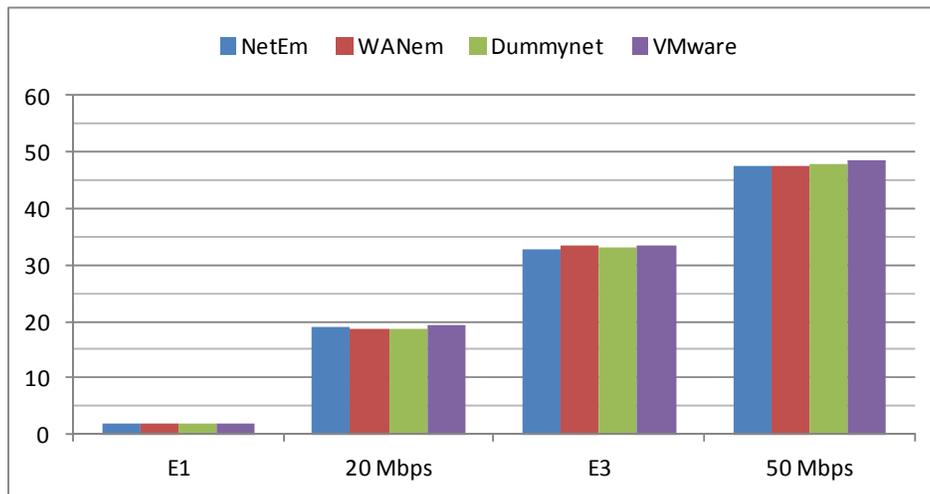


Figura 9: Experimentos de Ancho de Banda con IPv6

Finalmente, la Figura 10 muestra los resultados de los experimentos de retardo con IPv6. Como en el caso de IPv4, VMware fue excluido por su falta de soporte de emulación de este parámetro. Nuevamente los valores reportados por WANem son mayores a los configurados. Se asume que esto es derivado a la ejecución desde un *live CD*. DummyNet mostró los valores más ajustados a los configurados.

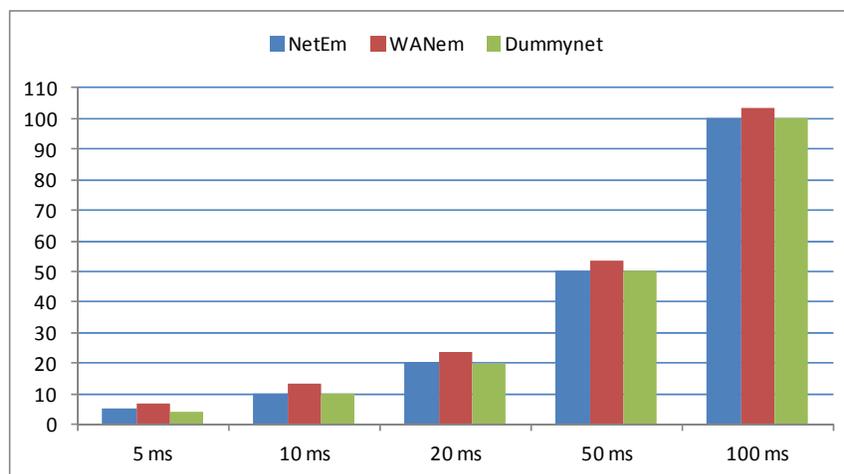


Figura 10: Experimentos de Retardo con IPv6

5. CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo se presentaron los resultados de una investigación en la que se evaluaron y analizaron varias herramientas de emulación de redes WAN. La evaluación incluyó la modificación de tres parámetros de red: tasa de pérdida de paquetes, ancho de banda, y retardo. Se realizaron experimentos usando IPv4 e IPv6 como protocolos de red. Para el estudio, se consideraron seis herramientas: NetEm, WANem, NIST Net, DummyNet, WANSim, y VMware. Se realizaron una serie de experimentos para probar la precisión de los resultados arrojados por cada herramienta. Para los experimentos, se usó D-ITG y también se desarrolló un benchmark propio para calcular el retardo de una vía (OWD, One Way Delay). Se pudo observar que WANSim no reportó resultados coherentes, posiblemente por una falla de configuración o por un problema en la herramienta. Por este motivo, WANSim fue excluido del análisis realizado. Para los experimentos con

tráfico IPv6, también se excluyó NIST Net, ya que no ofrece soporte para este protocolo. El resto de las herramientas mostró un comportamiento razonable, acercándose a los valores configurados. VMWare y NetEm reportaron los valores más cercanos a los parámetros configurados. Para valores altos de tasas de pérdida, Dummynet fue el menos preciso mientras que el resto de las herramientas mostraron un buen desempeño. Para el caso del ancho de banda, NetEm fue el más exacto, aunque las demás herramientas también mostraron buenos resultados. Finalmente, para los experimentos de retardo, WANem fue la única herramienta cuyos reportes fueron mayores a los valores esperados. Esto puede ser consecuencia de su ejecución a través de un *live CD*. De acuerdo a los resultados obtenidos, se puede decir que (con excepción de WANSim, y NIST Net en el caso de IPv6) todas las herramientas de emulación evaluadas mostraron una buena precisión en los parámetros estudiados, reportando valores cercanos a los configurados tanto para IPv4 como para IPv6; así que al momento de elegir una herramienta, la razón principal debe ser el soporte del parámetro requerido. Otros puntos importantes al momento de tomar una decisión deben ser la documentación y la actividad reciente.

Como trabajos futuros se propone usar estas herramientas para validar y evaluar algunos protocolos de red diseñados e implementados en nuestra Universidad. Este es el caso de DBRP (Delay Based Routing Protocol) [4], un protocolo basado en el algoritmo de vector-distancia con una métrica basada en el retardo, capaz de transportar información de servicios adicionales, como DNS. Como la métrica de este protocolo está basada en el retardo, las herramientas evaluadas en este trabajo pueden ser muy útiles para corroborar el correcto funcionamiento de DBRP.

REFERENCIAS

- [1] Avalone, S., Pescape, A., and Ventre, G. Distributed Internet Traffic Generator (D-ITG): Analysis and Experimentation over Heterogeneous Networks. In *Proceedings of International Conference on Network Protocols*. (Atlanta, Georgia, USA. November 2003).
- [2] Carbone, M. and Rizzo, L. Dummynet Revisited. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, SIGCOMM10*. (New Delhi, India. April 2010).
- [3] Carson, M. and Santay, D. NIST Net – A Linux-based Network Emulation Tool. *ACM Computer Communication Review*. Volume 33, Number 3. (July 2003).
- [4] Gameess, E., Marrero, P., and Gámez, D. A Delay Based Routing Protocol with Support for Common Services. In *Proceedings of the 2011 Latin America Networking Conference, LANC'11*. (Quito, Ecuador. October, 2011).
- [5] Gameess, E. and Veracoechea, C. A Comparative Analysis of Network Simulation Tools. In *Proceedings of the 2010 International Conference on Modeling, Simulation and Visualization Methods*. (Las Vegas, Nevada, USA. July 2010).
- [6] Hemminger, S. Network Emulation with NetEm. In *Proceedings of Australia's 6th National Linux Conference*. (Canberra, Australia. April 2005).
- [7] Kalita, H. and Nambiar, M. Designing WANem: A Wide Area Network Emulator tool. In *Proceedings of the Third International Conference on Communication Systems and Networks, COMSNETS*. (Bangalore, India. January 2011).
- [8] Rizzo, L. Dummynet: A Simple Approach to the Evaluation of Network Protocols. *ACM Computer Communication Review*. Volume 27, Issue 1. (January 1997).
- [9] Velásquez, K. and Gameess, E. A Survey of Network Benchmark Tools. *Machine Learning and System Engineering* (2010), pp. 465-480. Springer.
- [10] VMware. *Using VMware Workstation - VMware Workstation 8*. EN-000558-00. 2011.