

**Tercera Conferencia de Directores de Tecnología de
Información y Comunicación en Instituciones de Educación
Superior: Soluciones para la Enseñanza y la Investigación.
Laboratorio Remoto Virtual para la Enseñanza de
Administración de Redes**

Daniel Britos, Laura Vargas, Silvia Arias, Nicolás Giraudo, Guillermo Veneranda

Laboratorio de Redes y Comunicaciones de Datos, FCEFyN, Universidad Nacional de
Córdoba, Ciudad Universitaria, Argentina
dbritos@efn.uncor.edu, lvargas@efn.uncor.edu, sil_var@hotmail.com,
nicolasgiraudo@hotmail.com, guillevene@gmail.com

Resumen. El presente trabajo presenta un laboratorio remoto y virtual para la enseñanza de redes de computadoras. La implementación seleccionada plantea como alternativa a la utilización de las herramientas que simulan componentes, ya que estas solo ofrecen los comandos que el desarrollador decidió incluir. En este caso, se hacen las prácticas con sistemas operativos de Interconexión de redes originales, accediendo así a todos los comandos y parámetros posibles. Por otra parte, el acceso en forma remota a los equipos reproduce una situación real y es un modo flexible respecto a necesidades de alumnos y docentes. La originalidad del presente trabajo consiste en que el alumno con computadoras de escritorio de bajo costo pueda acceder a trabajar con redes complejas, ya que la carga de procesamiento esta en los servidores remotos.

Palabras Clave: emulación, virtualización, IOS, GNS3, redes de computadoras.

1 Introducción

Las redes académicas y gubernamentales crecen en complejidad y su administración requiere de personal idóneo altamente calificado, el trasladar a los administradores de redes a realizar cursos en centros de formación es caro e implica dejar desatendidas sus redes, por otra parte el traslado de instructores a los sitios remotos, requiere que estos sitios cuenten con laboratorios con la suficiente infraestructura para realizarlos. Dado el alto costo que tiene trabajar en laboratorios con equipos físicos y considerando que en las redes la administración se suele realizar en forma remota, se decidió implementar un laboratorio virtual y remoto de redes de computadoras. Para esto se analizaron distintas herramientas gratuitas, una vez hecha la selección de las mismas, se sometió el conjunto a pruebas a fin de evaluar su rendimiento. De este modo, se buscó complementar prácticas en equipos físicos y virtuales simultáneamente, así como permitir la interacción entre los mismos.

El estudio de las asignaturas relacionadas con redes de computadoras posee una extensa base teórica de la que se abreva, pero las competencias requeridas para

administrar redes se basan en las aptitudes desarrolladas por la práctica en laboratorios. En particular, en las carreras de ingeniería se espera que el estudiante adquiera competencias en diseño, implementación y administración de redes, por lo que tiene especial importancia la interrelación de la teoría y la práctica. La dificultad de contar y mantener actualizados los laboratorios requiere de grandes inversiones en equipamiento, los que rápidamente se vuelven obsoletos y cada vez son más costosos, sumada a la necesidad de sostener personal suficiente para mantener las prácticas.

Esta implementación se distingue de trabajos anteriores en que todo la carga de procesamiento requerido se encuentra en el servidor remoto, pudiendo el alumno acceder a administrar una red y a dispositivos de redes, con computadoras de baja prestaciones. La emulación de los dispositivos de redes router, switchers y servidores se ejecutan en un servidor remoto.

La posibilidad de implementar una herramienta para la simulación de redes de computadoras con dispositivos de red emulados dio el impulso inicial para esta investigación. Los laboratorios de redes virtuales le acercan a los ingenieros y/o administradores de redes una nueva alternativa para poner en práctica los conocimientos teóricos adquiridos y experimentar todas las funcionalidades de los Sistemas Operativos de Interconexión de redes.

En el Capítulo 2 se presenta una breve introducción a la virtualización, en el Capítulo 3 se expone la virtualización de laboratorios, en el Capítulo 4 analiza la selección de herramientas para la virtualización, en el Capítulo 5 se muestra como se van a integrar la distintas herramientas seleccionadas, en el Capítulo 6 se propone una arquitectura para la implementación del laboratorio virtual, en el Capítulo 7 se presentan las particularidades del modelo elegido, en el Capítulo 8 se realizan las pruebas de rendimiento y finalmente en el Capítulo 9 se muestran las conclusiones finales.

2 Virtualización

La *virtualización* es la creación de una versión virtual de algún recurso tecnológico [1]. La misma oculta las características físicas de los recursos tecnológicos a las aplicaciones haciendo que un solo recurso de hardware parezca como múltiples recursos virtuales y viceversa, que múltiples recursos físicos aparezcan un único recurso virtual.

La *virtualización de plataforma o de hardware* es la creación de una Máquina Virtual que actúa como una computadora real con un Sistema Operativo (SO). El Software ejecutado en estas Máquinas Virtuales está separado de los recursos de hardware subyacentes.

Una *Máquina Virtual (VM)* es una representación lógica en software de una computadora que ejecuta programas como una máquina real. Al desacoplar el hardware físico del SO, la virtualización de plataformas permite compartir los recursos tecnológicos entre múltiples VMs, cada una corriendo su propio SO, aumentando así la tasa de utilización del hardware subyacente [2].

Una *máquina host* es la máquina física en la cual se lleva a cabo la virtualización, y una máquina guest es la VM. Las palabras host y guest se utilizan también para distinguir el software que se ejecuta en la máquina física del software que se ejecuta en la VM.

Un *hypervisor* o *Virtual Machine Monitor (VMM)* es el software o firmware que crea una capa de abstracción entre el hardware del host y el SO del guest, dividiéndose el recurso tecnológico en uno o más entornos de ejecución. El VMM maneja, gestiona y arbitra los cuatro recursos principales de una computadora (CPU, Memoria, Almacenamiento y Conexiones de Red) distribuyendo los mismos en forma estática o dinámica entre todas las VMs definidas en el computador central. Los hypervisors se pueden clasificar en dos tipos [3]:

- Hypervisor de tipo 1 (nativo o bare-metal): se ejecuta directamente sobre el hardware del host. (Citrix XenServer, VMware ESX/ESXi y Microsoft Hyper-V).
- Hypervisor de tipo 2 (sobre host): se ejecuta dentro de un entorno de SO convencional. (KVM y VirtualBox).

Un *emulador* es un hypervisor que imita las funciones de una computadora (guest) en otra computadora diferente (host), independientemente de cómo sea el hardware físico, de modo que el comportamiento imitado se asemeja mucho al comportamiento de la computadora real. Se destaca por hacer énfasis en la reproducción del comportamiento característico y de aspectos funcionales de otro dispositivo de hardware, dejando de lado aspectos no funcionales como el tiempo de respuesta o el rendimiento.

Una *red virtual* es una red de computadoras que consiste, al menos en parte, de enlaces de red virtuales. Un enlace de red virtual no se implementa mediante una conexión física (por cable o inalámbrica) entre dos dispositivos de computación, sino que se lleva a cabo utilizando métodos de virtualización de redes.

Las principales ventajas de la virtualización de plataforma son:

- Permite la coexistencia de múltiples SO en el mismo computador, aislados unos de otros.
- La VM puede proveer un ISA (Instruction Set Architecture) diferente al de la máquina física.

3 Desarrollo del laboratorio virtual

El interés del laboratorio se centra en las redes virtuales basadas en dispositivos que conectan VMs dentro de un hypervisor y con el mundo exterior. De ahora en adelante, cuando se hable de redes virtuales se considerarán exclusivamente las basadas en dispositivos virtuales y no en protocolos, cuyas componentes son:

- Switch virtual (vSwitch)
- Adaptador Ethernet Virtual (VEA)
- Servidor DHCP y DNS virtual
- Enlace virtual

Los componentes clave son los VEAs utilizados por las VMs y los vSwitches que conectan las VMs entre sí. Los vSwitches permiten a las VMs, comunicarse entre sí

utilizando los mismos protocolos usados por los switches físicos. Un vSwitch es una simple construcción de software que generalmente está dentro del hypervisor pero a veces se presenta al SO host como una interfaz de red. Una VM puede ser configurada con uno o varios VEAs, cada uno de los cuales tiene su propia dirección IP y MAC. Como resultado, desde el punto de vista de las redes, las VMs tienen las mismas propiedades que las máquinas físicas [4].

Para cada VM, el hypervisor puede crear varios VEAs, seleccionando en cada uno, tanto el tipo de hardware que será virtualizado como el modo de virtualización en el cual estará operando en relación con la red física del host.

4 Selección de herramientas para la virtualización.

En una primera etapa, se analizaron varias herramientas para simular redes de computadoras.

Tabla 1. Herramientas para virtualización de redes de computadoras

Herramienta	Packet Tracert	Boson NetSim	VNX	GNS3	Netkit	Marionet	CORE
Licencia	Privativo	Privativo	GPL	GPLv2	GPL	GPLv2	BSD
Última versión	5.3.3 Feb, 2012	8.0 Nov, 2011	2.0 Mar, 2012	0.8.4 Feb, 2013	2.8 May, 2011	0.6 Feb, 2011	4.3 Mar, 2012
Sistemas Operativos soportados	Windows Linux	Windows	Linux	Windows Linux Mac OS X	Linux	Linux	Linux FreeBSD
Soporta los TP	Si	Si	No	Si	Si	Si	Si
Fácil de utilizar	Si	Si	No	Si	No	Si	Si
IOS Cisco	No	No	No	Si	No	No	No
GUI	Si	Si	Si	Si	No	Si	Si
Conexión con equipos reales	No	No	Si	Si	Si	Si	Si

Luego del análisis realizado sobre las herramientas disponibles (Tabla 1), se decide utilizar el programa GNS3 como GUI y Dynamips como emulador ya que cumplen con las siguientes condiciones:

- Ambos son de distribución gratuita.
- GNS3 posee una interfaz gráfica simple y fácil de utilizar.
- Dynamips emula IOS Cisco reales. Puede ejecutarse en hosts independientes y permitir balanceo de carga.
- Pueden ejecutarse en un SO potente y robusto como lo es Linux.
- Permiten la conexión con dispositivos reales.
- Ambos se encuentran en pleno desarrollo y con constantes mejoras.

Respecto a la virtualización de computadoras, existe una amplia oferta de software. VirtualBox es la herramienta que mejor se adapta a los requerimientos planteados. Cabe destacar que VirtualBox puede ser instalado en un servidor dedicado, tiene la capacidad de clonar VMs, está integrado a GNS3 y mediante phpVirtualBox se puede acceder a las distintas VMs a través de un navegador web, requerimiento indispensable para que el cliente pueda acceder mediante un navegador a las VM.

Tabla 1. Herramientas para virtualización de computadoras.

Herramienta	VirtualBox	Vmware Server	ESXi	XenServer	QEMU
Hypervisor	Tipo 2	Tipo 2	Tipo 1	Tipo 1	Tipo 2
Entorno subyacente en el servidor	Windows Linux	Windows Linux	Hardware	Hardware	Windows Linux Mac
Licencia	VirtualBox PUJEL	Gratuita	Gratuita	Gratuita	GNU GPL
Estado del proyecto	Activo	Discont.	Activo	Activo	Activo
Soporte de software	Si	No	Si	Si	Si
Admin. remota	CLI	WUI	GUI	CLI	CLI
	WUI			GUI	WUI
Entorno subyacente en el cliente	Windows Linux	Browser	Windows	Windows Linux	Windows Linux
	Browser				Browser
Integración con GNS3	Si	No	Si	Si	Si
Interfaz de red virtual accesible	Si	Si	No	No	Si
Permisos y propiedad de las VMs	No	No	Si	Si	No
Comp. Con hardware	Alta	Alta	Baja	Baja	Alta
Manipulación de imágenes de disco	Alta	Alta	Baja	Baja	Alta
Asistente para clonación	Si	No	No	No	No
Snapshots	Si	Si	Si	Si	Si

Se elige la herramienta VirtualBox, aun cuando no se trata de un software de distribución libre y de que phpVirtualBox no maneja el concepto de propiedad de una VM. VMware Server fue en algún momento el gran candidato pero como el proyecto se ha discontinuado se lo descartó.

5 Integración de las herramientas

En un paso posterior, se integraron las herramientas. La emulación de routers Cisco se realiza con el programa Dynamips mientras que para virtualizar los dispositivos finales se utiliza VirtualBox. Además, VirtualBox, hypervisor tipo 2, puede virtualizar routers para luego ser utilizados por GNS3 [6] como componentes de una topología virtual. Como interfaz web para la administración remota de las VMs se usa phpVirtualBox.

VBoxWrapper es un módulo servidor escrito en Python y usado para controlar VMs. En una topología de red creada con GNS3, una VM debe tener configurado su adaptador de red virtual en modo adaptador genérico para que pueda comunicarse con otro dispositivo [7].

PhpVirtualBox es un proyecto independiente que consiste en una WUI (Web User Interface) para administrar y monitorear las VMs por medio de un navegador web [8]. Cuando un usuario accede a phpVirtualBox, obtiene control total sobre todas las VMs dentro del entorno de VirtualBox. PhpVirtualBox realiza todas las acciones administrativas a través de vboxwebsrv, un servidor SOAP (Simple Object Access Protocol) distribuido junto con VirtualBox [9].

El servicio web es brindado por un archivo independiente llamado vboxwebsrv que actúa como un servidor HTTP. Vboxwebsrv es un programa de consola en modo texto que viene incluido con el paquete binario estándar de VirtualBox. Además, acepta conexiones SOAP y las procesa de manera remota o desde la misma máquina. Una vez que el servicio web se ha iniciado, actúa como un front end para el VirtualBox del usuario, que está ejecutando vboxwebsrv [10].

6 Arquitectura propuesta

Respecto al funcionamiento conjunto, la forma más fácil de trabajar con estas herramientas es ejecutándolas en la misma computadora. Previamente a agregar un componente en GNS3 se debe configurar su imagen IOS. Desde el momento en que se agrega el primer router a la topología, GNS3 ejecuta Dynamips automáticamente en segundo plano y queda a la escucha en el puerto correspondiente. Luego de que un router ha sido encendido es posible abrir su consola. De igual forma ocurre con el programa VBoxWrapper cuando lo que se agrega es una VM. Para que una VM pueda ser añadida hay que pre-configurarla en GNS3. Al encender una VM se abre su consola de VirtualBox.

La emulación de routers necesita gran cantidad de recursos de procesamiento y memoria. Muchas veces esos requerimientos no son alcanzados con el hardware de una computadora. GNS3 ofrece la posibilidad de configurar hypervisors externos para

delegar esta carga de trabajo. Un hypervisor externo es una instancia de Dynamips ejecutándose en otra computadora. Al momento de crear la topología se puede seleccionar explícitamente en cual hypervisor es emulado cada router. Pero también se puede configurar a GNS3 para que realice automáticamente un balanceo de carga entre los hypervisors.

La virtualización de computadoras también requiere una gran cantidad de recursos computacionales. Distribuir esta carga de trabajo requiere que GNS3 se pueda comunicar con una instancia de VirtualBox corriendo en otra computadora. La comunicación se logra ejecutando VBoxWrapper en el servidor de virtualización y configurando GNS3 con la dirección IP del mismo. Una vez realizado esto, será posible prender y apagar las VMs remotamente desde GNS3. Para poder iniciar una sesión remota en estos guest es necesario instalar y configurar phpVirtualBox. Este último requiere que VBoxWebSrv esté ejecutándose para establecer una comunicación con VirtualBox. También se pueden configurar varios servidores de virtualización y dividir la carga de trabajo manualmente entre ellos.

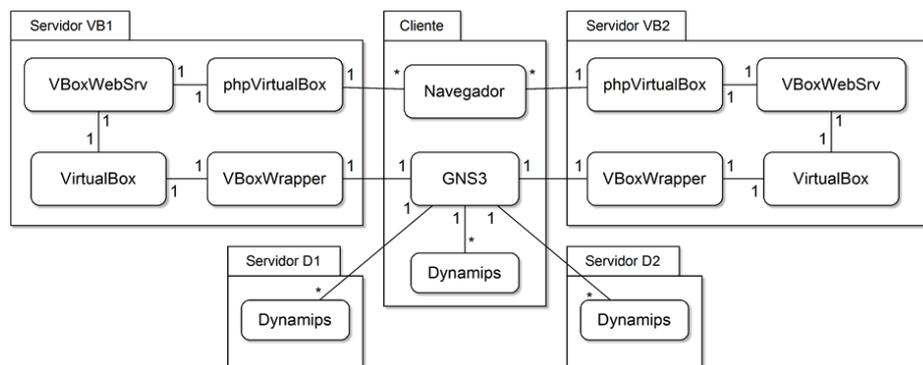


Figura 1 - Dos servidores VirtualBox y dos servidores Dynamips externos

En la Figura 1 se observa la arquitectura planteada.

7 Particularidades del modelo elegido

GNS3 permite conectar dispositivos virtuales con equipos reales. La conexión se realiza utilizando nodos tipo nube que pueden comunicar redes virtuales con redes físicas. Es necesario ejecutar GNS3 como root para configurar correctamente la nube, además se debe emular en forma local al dispositivo.

Se puede concluir que la utilización conjunta de las herramientas elegidas permite virtualizar redes complejas de computadoras. Además, la arquitectura formada por estas herramientas es altamente escalable, por lo tanto, los clientes pueden utilizar varios servidores externos como se observa en la Figura 2.

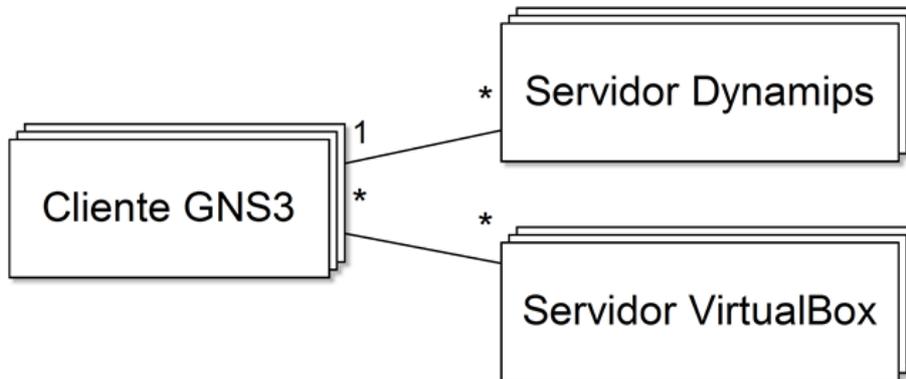


Figura 2 - Clientes y Servidores.

GNS3 no tiene las funcionalidades necesarias para permitir a varios clientes trabajar sobre una misma topología de red. Sin embargo, si la topología se divide en partes y cada una es simulada localmente en un cliente GNS3 distinto, entonces, es posible conectar las partes y trabajar de manera distribuida sobre la misma topología. WinPcap en Windows y libpcap en Linux son las herramientas que permiten a GNS3 capturar, inyectar, analizar y construir paquetes en la red física del host, como se muestra en la Figura 3.

Con VirtualBox es posible crear VMs con SOs Linux adaptados para funcionar como routers virtuales.

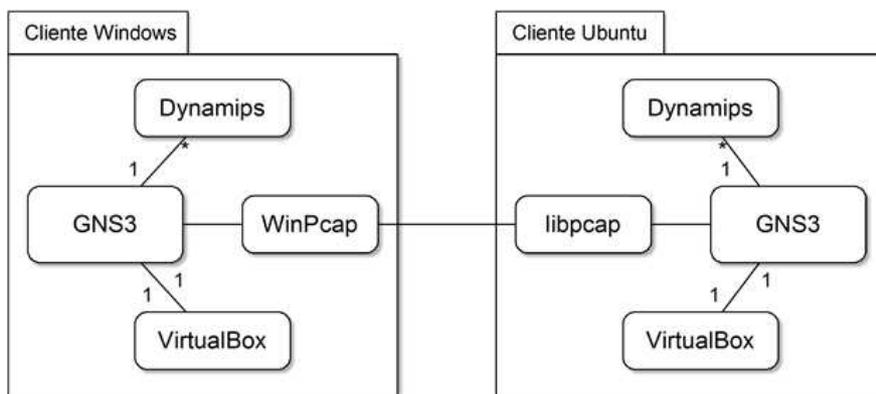


Figura 3 - Comunicación entre clientes GNS3

8 Pruebas de rendimiento

En una segunda etapa, se realizaron pruebas para evaluar el rendimiento. El Prototipo es el servidor encargado de emular los routers. Debido a que este servicio es escalable horizontalmente, se configuraron dos hosts para que lleven a cabo las tareas de emulación. En las Tablas 3 y 4 se describen respectivamente las características de hardware del Servidor 1 y Servidor 2.

Tabla 1 - Hardware Servidor 1

Componente	Cantidad	Modelo
Procesador	1	AMD Phenom II X4 955 3200 MHz
Memoria	2	G.Skill RipjawsX DDR3-1600 PC3-12800 2 GB
Motherboard	1	GA-880GM-UD2H
Disco Rígido	1	250 GB

Tabla 2 - Hardware Servidor 2

Componente	Cantidad	Modelo
Procesador	1	Intel Core 2 Duo E6750 2660 MHz
Memoria	2	G.Skill PI Black Edition DDR2-800 PC2-6400 4 GB
Motherboard	1	Intel DP35DP
Disco Rígido	1	500 GB

Se resalta la escalabilidad horizontal que tiene la herramienta porque permite proyectar el conjunto de servidores necesarios para emular los 90 routers requeridos. La memoria RAM es más fácil de escalar verticalmente en cambio el poder de procesamiento es más fácil de escalar horizontalmente.

Se planteó una prueba de carga consistente en someter al prototipo Servidor 1 a una carga de trabajo equivalente a una clase práctica en el Laboratorio de Redes y Comunicaciones de Datos para 7 grupos de alumnos.

Como resultado de todas las pruebas de estrés realizadas se pudo concluir que la terminación del proceso Dynamips, por motivo de una falla interna o por la llegada de una señal externa, detiene la simulación en curso y no puede restablecerse, aunque la topología puede volver a usarse. Además, un aumento considerable del tráfico de red que atraviesa los routers, detiene la simulación y corrompe la topología de manera que no puede volver a utilizarse. Por otra parte, la pérdida de conectividad entre el servidor y los clientes no detiene la simulación, una vez restablecida dicha conectividad, los dispositivos pueden volver a comunicarse entre sí.

En base a las pruebas realizadas sobre el prototipo se determinó que con 12 GB de RAM se puede suplir la necesidad de memoria que implica la emulación de 90

routers. Las pruebas también reflejan que el Servidor 1 puede soportar la carga de procesamiento de emular 30 routers. Por lo tanto, para emular 90 routers se necesita un poder de cómputo equivalente al de 3 servidores similares. GNS3 y VirtualBox permiten crear topologías complejas de redes de computadoras sin necesidad de contar con equipos de interconexión ni gran cantidad de recursos computacionales.

Respecto a las aplicaciones, se pudo observar el alto grado de escalabilidad que tienen las herramientas seleccionadas. Montando 3 servidores similares al prototipo, cada uno con 4GB de RAM, es posible emular 90 routers.

No obstante, no es necesario comprar 3 computadoras, en una primera fase pueden utilizarse equipos en existencia de menor potencia hasta alcanzar los requerimientos. Luego de evaluar el grado de aceptación de la herramienta, en una etapa más avanzada o cuando se considere conveniente, se recomienda adquirir un equipo servidor de características similares a las mostradas en la Tabla 5. Los precios están expresados en moneda Argentina.

Tabla 3 - Hardware del servidor recomendado

Componente	Cantidad	Modelo	Precio
Procesador	2	AMD Opteron 4280 Valencia 2.8 GHz	\$ 2.475
Memoria	4	4 GB ECC Registered DDR3 1333 Mhz	\$ 693
Motherboard	1	Supermicro H8DCT-HLN4F-B two Socket C32	\$ 1.742
Disco Rígido	1	Western Digital 1 TB 7200 RPM 64 MB Cache SATA 6.0 Gb/s	\$ 500
Total			\$ 5.410

9 Conclusiones

Las herramientas GNS3 y VirtualBox permiten crear topologías complejas de redes de computadoras sin necesidad de contar con equipos de interconexión ni gran cantidad de recursos computacionales. A continuación se listan las conclusiones detalladas con respecto a las aplicaciones utilizadas:

- GNS3 es una herramienta que está en continuo desarrollo, agregando nuevas funcionalidades, corrigiendo bugs e integrándose con nuevas herramientas. Esto obligó a cambiar el rumbo del Proyecto Integrador en varias oportunidades, pero también sirvió para evitar los problemas de integración encontrados con determinados hypervisors.
- GNS3 permite a varios usuarios trabajar sobre la misma topología.
- VirtualBox es una herramienta potente que permite experimentar con los SOs actuales más utilizados. Además, está integrado a GNS3.
- PhpVirtualBox no está diseñado para funcionar en un entorno de hosting donde se necesita el concepto de propiedad de una VM.

- Los routers emulados con Dynamips no soportan un alto tráfico de red. En un ambiente educativo posibilita experimentar las funcionalidades de las IOS permitiendo a los estudiantes afianzar sus conocimientos.

Con este trabajo se lograron cumplir los objetivos propuestos:

Objetivo General

Implementar un Laboratorio Virtual para hacer prácticas de redes de computadoras con acceso al mismo en forma remota con máquinas de bajas prestaciones y que posibilite la interacción directa entre los ingenieros y/o administradores de redes y los IOS de los equipos de interconexión de redes.

Objetivos Específicos

- Brindar una solución al problema del costo creciente de laboratorios de redes de computadoras.
- Crear, configurar y dar acceso a Máquinas Virtuales con Sistemas Operativos de escritorio.
- Configurar un servicio web que permita el acceso a las Máquinas Virtuales a través de un navegador.
- Diseñar topologías de laboratorios de redes virtuales de computadoras.

Trabajos Futuros

A partir del 2013, los alumnos podrán utilizar la arquitectura de laboratorios remotos planteada. Se evaluará in situ el rendimiento y se comparará con los resultados obtenidos en la primera etapa.

En cuanto a la investigación, se espera agregar a phpvirtualbox los conceptos de: propiedad de una VM, privilegios, roles de usuarios y grupos de usuarios y modificar el código de GNS3 para permitir a varios usuarios trabajar sobre la misma topología.

Se comparará una herramienta con fines educativos contra otra, diseñada para trabajar en un entorno de producción.

Agradecimientos

Este trabajo ha sido financiado parcialmente por el Proyecto de Investigación de la SeCyT de la Universidad Nacional de Córdoba.

Referencias

1. Turban, E; King, D; Lee, J; Viehland, D. *Chapter 19: Building E-Commerce Applications and Infrastructure*. Electronic Commerce A Managerial Perspective, 5ta Edición. Prentice-Hall, pp. 27. 2008.
2. *Virtualization in education*. IBM Global Education. <http://www.07.ibm.com/solutions/in/education/download/Virtualization%20in%20Education.pdf>, 2007.

3. Goldberg, R.P. *Architectural Principles for Virtual Computer Systems*. Harvard University [http:// flint.cs.yale.edu/cs428/ doc/goldberg.pdf](http://flint.cs.yale.edu/cs428/doc/goldberg.pdf). pp. 22-26, 1973.
4. VMware *Virtual Networking Concepts*. [http:// www.vmware.com/files/pdf/virtual_networking_concepts.pdf](http://www.vmware.com/files/pdf/virtual_networking_concepts.pdf), 2007.
5. *Chapter 6. Virtual networking. Oracle VM VirtualBox User Manual*. Virtualbox.org. <http://www.virtualbox.org/manual/ch06.html>, 2012.
6. Graphical Network Simulator 3 <http://www.gns3.net/>, 2012.
7. *Welcome to VirtualBox.org!*. <https://www.virtualbox.org/>, 2012.
8. *VirtualBox Web Console*. <http://code.google.com/p/vboxweb/> , 2012.
9. *phpVirtualBox* <http://code.google.com/p/phpvirtualbox/> , 2012.
10. Oracle VM VirtualBox: Programming Guide and Reference. Version 4.1.18, <http://download.virtualbox.org/virtualbox/SDKRef.pdf>, 2012.