

# Aplicación al Ámbito Académico de un entorno de Simulación/Emulación de Arquitecturas de Red

José Javier Serrano, Julián Fernández-Navajas, José M<sup>a</sup> Saldaña  
Grupo de Tecnologías de las Comunicaciones – Instituto de Investigación en Ingeniería de Aragón  
Dpt. IEC. EINA, Universidad de Zaragoza  
Edif. Ada Byron, 50018, Zaragoza  
{182482, navajas, jsaldana}@unizar.es

**Resumen** – En este artículo se presenta un estudio sobre el empleo de una herramienta de emulación de red para la docencia en Ingeniería Telemática. La posibilidad de integrar, en un mismo escenario, nodos reales disponibles en el laboratorio junto con dispositivos hardware comerciales emulados, presenta muchas ventajas para la formación de los alumnos y la realización de prácticas de laboratorio. En concreto, se ha estudiado el entorno *GNS3/Dynamips*, originalmente diseñado para realizar pruebas de dispositivos de *Cisco* y *Juniper* en entornos telemáticos. En primer lugar, se muestra cómo los dispositivos emulados implementan exactamente las funcionalidades de los dispositivos reales correspondientes. Se han propuesto diferentes escenarios en los que los alumnos pueden realizar pruebas con distintos servicios, integrando máquinas reales con hardware emulado. Se presentan también pruebas encaminadas a caracterizar la escalabilidad del sistema, y las condiciones en las que el emulador se comporta igual que los dispositivos reales. En concreto, se analiza el incremento en el consumo de recursos del sistema emulador (carga de CPU y consumo de memoria RAM) al aumentar el número de nodos emulados. Este consumo puede determinar la escalabilidad de un escenario de trabajo emulado. Además, haciendo uso del protocolo ICMP, se miden las latencias de red que pueden resultar críticas, con el objetivo de comparar los entornos de red reales con los entornos híbridos, en los que el hardware real interopera con el entorno emulado.

**Palabras clave** - telemática, emulación, EEES, red virtual.

## I. INTRODUCCIÓN

El paradigma del aprendizaje centrado en el alumno y el desarrollo de sus competencias nos ayudan a entender la educación como algo más que la enseñanza o la mera adquisición de conocimientos. Este paradigma exige una adaptación de la mentalidad del profesorado y la aplicación de nuevas tecnologías en el diseño de planes de estudios, la preparación de asignaturas, la planificación docente, la conducción de las clases y el seguimiento, tutorización y evaluación de los estudiantes.

Unido a esto, resulta prioritario desde el prisma docente concretar y delimitar aspectos fundamentales, como la

consecución y control de los objetivos académicos. También se requiere un reflejo y vinculación directa entre los objetivos académicos y los escenarios de trabajo reales que el alumno, tarde o temprano, deberá afrontar. Por ello, en el contexto de los planes de estudio actuales, para la consecución de estos propósitos resulta deseable poner a disposición del alumno entornos controlados y prácticos de trabajo. De esta manera, se podrán marcar sus pautas del aprendizaje y simultáneamente, será posible delimitar el compromiso entre nivel de exigencia y dificultad asociada a la consecución de los objetivos académicos. Todo ello a través del trabajo práctico continuo del alumno y la supervisión de los objetivos parciales, finalizando en la evaluación global del objetivo final.

En este trabajo se analiza el potencial de un entorno software de simulación/emulación, denominado *GNS3/Dynamips*<sup>1</sup>, para el cumplimiento de los objetivos docentes anteriores. A grandes rasgos, a través de un sencillo e intuitivo interfaz gráfico, este entorno permite implementar topologías asociadas a arquitecturas de amplio espectro tecnológico, abarcando desde los diseños más básicos, hasta tecnologías de uso preferente en la actualidad.

Este entorno está principalmente concebido para emular arquitecturas de red que integran dispositivos de *Cisco Systems* y *Juniper* en entornos corporativos, haciendo uso para ello de las imágenes de los sistemas operativos de los equipos reales. Además, incorpora un considerable abanico de dispositivos de red generales que abarcan desde *switch Ethernet*, *Frame Relay* o ATM, hasta dispositivos *Firewall*, sin olvidar la emulación de terminales Linux. Todo esto permite un amplio rango de posibilidades en el ámbito docente, resultando muy interesante la posibilidad de acotar objetivos parciales y finales de aprendizaje, según los niveles de exigencia que se quieran establecer.

Añadido a esto, se trata de un entorno fundamentalmente práctico (aunque con una carga telemática teórica evidente), dado que el diseño y las configuraciones a implementar son idénticas o muy similares a las que se encontrarán en escenarios de red reales. Por ejemplo, la interacción con los dispositivos se realiza a través de un *prompt* o intérprete de comandos idéntico al de los dispositivos reales. Por ello, la

---

<sup>1</sup> <http://www.gns3.net/dynamips/>

asimilación de los contenidos por parte del alumno puede resultar más consistente que la ofrecida por un entorno de trabajo únicamente simulado.

Pero la característica funcional más relevante y decisiva de *GNS3/Dynamips*, y que analizaremos en el presente trabajo, es que permite entornos mixtos, en los que equipos y tráfico de red reales pueden interoperar, en un mismo escenario, con equipos y tráfico de red emulados. Este hecho presenta unas interesantes ventajas prácticas para el diseño y el análisis de los entornos. Por ejemplo, en el ámbito docente, los alumnos podrán lanzar pruebas desde equipos reales, interactuando con un gran abanico de equipos de red, aunque de hecho no se disponga de ellos en el laboratorio.

Además, a través de la herramienta software *Wireshark*, integrada en el entorno, se posibilita la visualización de capturas de tráfico emulado mediante librerías *.cap*, pudiendo analizarse de un modo similar al de una red real. Este tráfico puede tener como origen y/o destino un dispositivo de red real o bien emulado, según sea el propósito del análisis. Esta característica facilita un amplio abanico de arquitecturas, tráficos de red y servicios que pueden ser estudiados desde el ámbito de la docencia en telemática, pero con un claro enfoque hacia el mundo profesional.

La Fig.1 muestra el interfaz gráfico *GNS3*, sobre el que se ha implementado una sencilla estructura de red, que integra un equipo real externo (usando el ítem de diseño *cloud* o nube), que interopera con los nodos del escenario emulado.

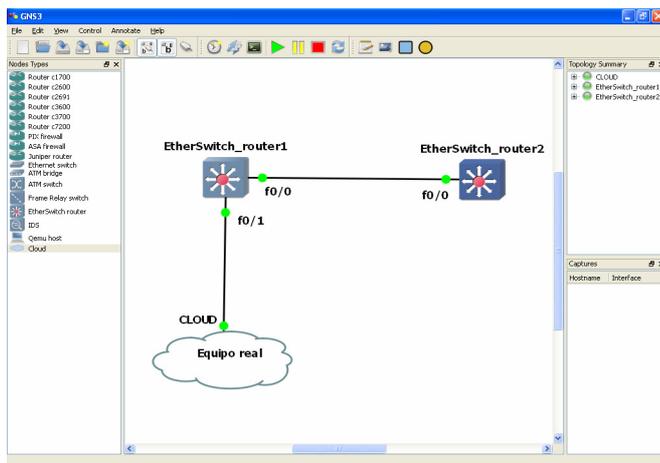


Fig. 1. Panorámica general del entorno *GNS3/Dynamips* con la integración de un equipo real en el entorno emulado.

Por tanto, en este trabajo nos centraremos en el estudio de la capacidad del entorno de emulación *GNS3/Dynamips* como una herramienta de trabajo adecuada para entornos académicos y en su interés para tareas docentes. Expondremos en detalle algunas de las pruebas que permite el sistema, mediante diferentes escenarios, centrándonos en los conceptos que los alumnos pueden aprender. También presentaremos algunas pruebas encaminadas a medir el nivel de realismo del sistema a la hora de emular dispositivos de red. Es importante que el alumno conozca los límites de la emulación, y que compruebe que existen experimentos cuyos resultados no serán idénticos a los que se obtendrían con dispositivos reales.

La estructura del resto del artículo es la siguiente: la sección II detalla los trabajos relacionados. La sección III explica brevemente la solución de emulación utilizada. La

siguiente sección detalla las diferentes arquitecturas de red que se han implementado en las pruebas. La sección V presenta las pruebas encaminadas a medir el rendimiento del sistema. Posteriormente, en la sección VI se estudia la influencia del sistema emulador en los parámetros medidos en la red y el artículo se cierra con las conclusiones.

## II. TRABAJOS RELACIONADOS

El interés por el reflejo y traslación de escenarios reales de trabajo hacia entornos virtuales va siendo cada vez mayor, y las propuestas concretas se han ido diversificando. Pueden encontrarse en la literatura especializada estudios de entornos de trabajo telemático que implementan soluciones de virtualización. Por ejemplo, [1] muestra una comparativa de plataformas virtuales, principalmente *Xen* y *VMware*. Estas pueden ser consideradas como las plataformas de virtualización más utilizadas y de mejor rendimiento en los sistemas de producción para la virtualización de servidores. En esencia, se constata comparativamente como la plataforma *Xen* (paravirtualización) presenta un mejor rendimiento que la plataforma *VMware* (virtualización completa).

Así mismo, [2] presenta algunas técnicas de virtualización existentes en el campo docente en la actualidad, incluyendo el análisis de ciertas características funcionales del entorno software *GNS3/Dynamips*. Se plantea en este caso la alternativa de uso de la plataforma XORP (eXtensible Open Router Platform) frente a *GNS3/Dynamips*, con el objetivo principal de minimizar el consumo de recursos del sistema.

Con todo ello, en el presente trabajo nuestro esfuerzo se va a centrar en el análisis del entorno simulador/emulador *GNS3/Dynamips* y en sus potenciales posibilidades.

También los autores de [3] muestran una posible aplicación directa de la utilización de este entorno software en el desarrollo práctico y virtualización de asignaturas de índole telemática, mediante la configuración de escenarios de red. Los escenarios presentados en dicho trabajo abarcan desde unos sencillos y básicos planteamientos iniciales de diseño, como la configuración de enrutamiento estático, hasta la implementación de servicios DHCP y NAT, así como la configuración de *firewall*. Se realizaron igualmente ciertos ensayos para analizar las características y potenciales prestaciones y usos del entorno software *GNS3/Dynamips*.

Estos ensayos, unidos a las referencias anteriores como botón de muestra de estudios y aplicaciones ya realizados, sugieren el empleo de este entorno de virtualización para el desarrollo y consolidación práctica de los contenidos teóricos de asignaturas del área de la Ingeniería Telemática. Unido a esto, la más que contrastada usabilidad del entorno software, facilita enormemente su uso en el ámbito docente.

## III. APLICACIÓN Y USO DE GNS3/DYNAMIPS

Los ensayos realizados en laboratorio van dirigidos, en primer lugar, a que el alumno pueda analizar el diseño y configuración de arquitecturas de red de propósito general, fundamentadas en hardware de red de *Cisco Systems*. En estas arquitecturas, los dispositivos emulados son imagen de elementos de red reales, y funcionan en base a sistemas *Cisco IOS*. Por tanto, estos sistemas operativos deberán ser previamente extraídos del hardware de red, para seguidamente incorporarse en los dispositivos emulados.

En segundo lugar, los análisis se focalizan en la potencial capacidad del entorno simulador/emulador para realizar

pruebas en entornos mixtos, integrando tráfico y equipos reales con escenarios emulados. Esta capacidad puede presentar aplicaciones prácticas interesantes para el alumno, como por ejemplo, segmentar arquitecturas de red en una “parte real” y una “parte emulada”, un recurso de diseño que solucionaría problemas logísticos a la hora de implementar topologías extensas con recursos hardware limitados.

Otra aplicación que resulta interesante, especialmente en el entorno docente, es la capacidad de inyectar tráfico real en el entorno emulado para su captura y análisis mediante la herramienta *Wireshark*. Esto permite estudiar las características del tráfico y el efecto de diferentes parámetros de red, como la latencia asociada a una determinada arquitectura de red en un servicio telemático.

#### IV. CONFIGURACIÓN DE ARQUITECTURAS DE RED

En base a las consideraciones anteriores, en esta sección describiremos los escenarios que se han implementado con *GNS3/Dynamips*. La idea es que, mediante dos ejemplos de arquitecturas de red, podamos realizar un análisis funcional del entorno emulador. En ambos casos, se realiza la interconexión de dos nodos de red por medio de dos tecnologías distintas. Esta configuración se realiza tanto en un entorno real como en un entorno mixto (real-emulado), para poder comparar así los resultados emulados y los obtenidos con equipos reales. De esta forma, cada nodo emulado representa una determinada ubicación física, correspondiente a una bancada del laboratorio de prácticas, y es imagen de un dispositivo de red real *Cisco Systems*. Para ello, sobre el dispositivo emulado se ha incorporado el sistema *Cisco IOS* extraído de un dispositivo real, según la metodología expuesta en [4].

##### A. Arquitectura de red WAN ppp

En este caso, se interconectan los nodos emulados *Nodo 1* y *Nodo 2* por medio de un enlace serie WAN *point-to-point* de 1,544 Mbps. Los nodos emulados son dispositivos multilayer de la familia *Cisco 3700 series*, que usan el sistema *Cisco IOS*. La Fig. 2 muestra la topología de esta arquitectura en el entorno mixto, así como las características más relevantes del diseño. Los nodos emulados *Auxiliar 1* y *Auxiliar 2* se utilizarán para funciones complementarias en las pruebas.

A la red emulada se conectan también dos equipos reales externos (PC portátiles) siguiendo las indicaciones de [4]. Su conexión se realiza por medio de un enlace *Fast Ethernet*. Las características de estos equipos no son relevantes en el diseño general de la arquitectura, puesto que sólo se utilizan para inyectar o recibir el tráfico de las pruebas. Del mismo modo, se incorporan en el esquema emulado mediante enlaces *Fast Ethernet* dos *host* virtuales (*QEMU2* y *QEMU3*), que ejecutan el sistema Linux *Microcore*. El entorno de emulación permite también la definición de diferentes VLAN. Haciendo uso de esta funcionalidad, consideramos los equipos reales externos adscritos administrativamente a *VLAN2*, y los *host* virtuales a *VLAN3*.

Por tanto, las configuraciones implementadas en los nodos emulados mediante comandos *Cisco IOS* serán idénticas a las implementadas en el escenario real del laboratorio. Cuando se realizan las configuraciones, el *prompt* o intérprete de comandos es el mismo que si estuviésemos configurando el dispositivo real a través de su puerto consola.

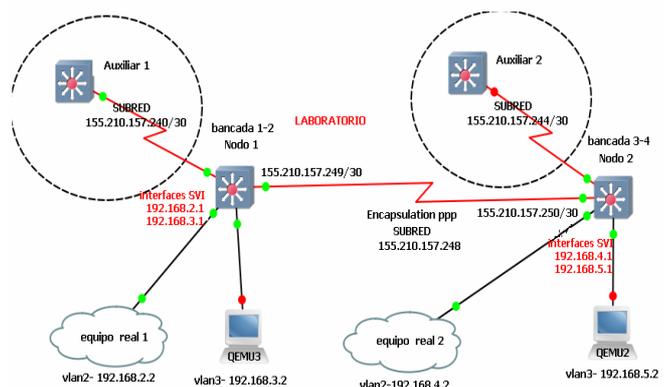


Fig. 2. Topología y características particulares de diseño de la arquitectura de red WAN ppp de estudio.

Cabe destacar que en el entorno emulado, dentro de las características y parámetros de configuración, podemos configurar el *prompt* para que su aspecto visual resulte más familiar para el alumno. En nuestro caso, se ha utilizado la herramienta software *putty*. La Fig. 3 muestra un ejemplo de las configuraciones implementadas en el nodo emulado 1 (*Nodo 1*) de la arquitectura, en este caso la configuración del enlace serie WAN ppp. En este escenario se procede a conferir ciertas características de diseño que permitirían implementar *InterVLAN routing* (conmutación a nivel 3) entre diferentes VLAN, a nivel local en los nodos emulados. Así mismo, no hay que olvidar que la comunicación entre VLAN en nodos diferentes se realiza a nivel de red.

```
Dynamips(0): R1, Console port
switch_router1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch_router1(config)#interface serial 2/0
switch_router1(config-if)#ip address 155.210.157.249 255.255.255.252
switch_router1(config-if)#no shutdown
switch_router1(config-if)#encapsulation ppp
switch_router1(config-if)#exit
switch_router1(config)#
```

Fig. 3. Configuración del enlace serie WAN ppp en el nodo emulado 1.

Para ello, se configura en nuestra arquitectura el protocolo de enrutamiento dinámico RIPv2, como muestra la Fig. 4. Seguidamente, la Fig. 5 muestra un ejemplo de cómo se implementarían las configuraciones en el nodo emulado 1, a través de los comandos *Cisco IOS* oportunos, mostrando así la asignación física de interfaces a cada VLAN para nuestro ejemplo práctico concreto (Fig. 5a), así como la configuración de los interfaces virtuales SVI, necesarios para implementar *InterVLAN routing* (Fig. 5b).

```
Dynamips(1): R1, Console port
switch_router1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch_router1(config)#router rip
switch_router1(config-router)#version 2
switch_router1(config-router)#network 155.210.0.0
switch_router1(config-router)#network 192.168.3.0
switch_router1(config-router)#network 192.168.2.0
switch_router1(config-router)#exit
switch_router1(config)#
```

Fig. 4. Configuración del protocolo de enrutamiento dinámico RIP v2 en el escenario emulado.

Una vez implementadas las configuraciones de red, el alumno puede realizar, por ejemplo, un sencillo *test* de la trayectoria del tráfico seguido en la comunicación de equipos pertenecientes a VLAN diferentes en el mismo nodo emulado, como indica la Fig.6. Así se puede comprobar que la arquitectura emulada funciona conforme a nuestros propósitos de diseño, según las configuraciones implementadas, presentando un comportamiento idéntico al de los dispositivos reales.

```
Dynamips(1): R1, Console port
switch_router1(config)#interface range fastEthernet 1/0 -7
switch_router1(config-if-range)#switchport mode access
switch_router1(config-if-range)#switchport access vlan 2
switch_router1(config-if-range)#exit
switch_router1(config)#interface range fastEthernet 1/8 -15
switch_router1(config-if-range)#switchport mode access
switch_router1(config-if-range)#switchport access vlan 3
switch_router1(config-if-range)#exit
switch_router1(config)#
```

(a)

```
Dynamips(1): R1, Console port
switch_router1(config)#interface vlan 2
switch_router1(config-if)#ip address 192.168.2.1 255.255.255.0
switch_router1(config-if)#no shutdown
switch_router1(config-if)#exit
switch_router1(config)#interface vlan 3
switch_router1(config-if)#ip address 192.168.3.1 255.255.255.0
switch_router1(config-if)#no shutdown
switch_router1(config-if)#exit
switch_router1(config)#
```

(b)

Fig. 5. Configuraciones VLAN en el nodo emulado 1 de la arquitectura WAN ppp.

```
127.0.0.1 - PuTTY
tc@QEMU3:~$ traceroute 192.168.2.2
traceroute to 192.168.2.2 (192.168.2.2), 30 hops max, 38 byte packets
 1 192.168.3.1 (192.168.3.1) 35.312 ms 23.111 ms 0.117 ms
 2 192.168.2.2 (192.168.2.2) 22.160 ms 6.133 ms 6.745 ms
tc@QEMU3:~$
```

Fig. 6. Trayectoria seguida desde el *host* virtual Linux QEMU3 → equipo real 1, en el nodo emulado 1 (conmutación a nivel 3)

### B. Arquitectura de red LAN Ethernet /VTP

En esta arquitectura se configura un entorno conmutado LAN *Ethernet* a nivel 2 y 3, según muestra la Fig. 7. En ella, interconectamos los nodos emulados 1 y 2 por medio de un enlace *Fast Ethernet* que actuará como *trunk-link* (enlace troncal). Por éste circularán, convenientemente identificadas, tramas etiquetadas (*tagged*) y no etiquetadas (*untagged frames*). Al igual que antes, los nodos *Auxiliar 1* y *Auxiliar 2* se incluyen en el esquema. Los equipos reales externos y los *host* virtuales Linux (*QEMU2* y *QEMU3*) se adscriben a *VLAN2* y *VLAN3*, respectivamente. Al tratarse de un entorno conmutado a nivel 2, no hay intercambio de información a nivel de red entre dispositivos emulados, por lo que no se configuran direcciones IP en los interfaces *Fast Ethernet* asignados al enlace troncal. En consecuencia, no se configura ningún protocolo de enrutamiento dinámico, como sí se hizo en la arquitectura anterior. Recordemos que los nodos emulados son dispositivos *multilayer*, imagen del hardware de red del laboratorio.

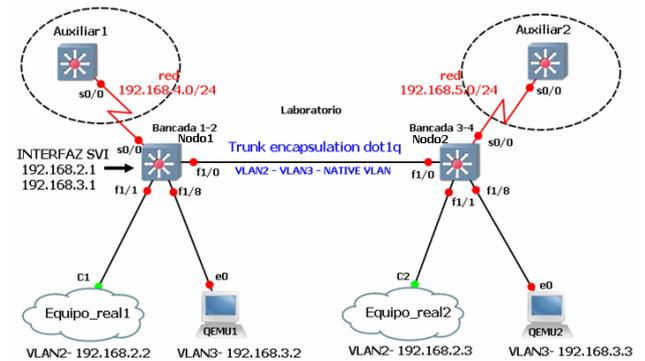


Fig. 7. Topología y características particulares de diseño de la arquitectura de red LAN *Ethernet*/VTP de estudio.

Siendo así, únicamente se confieren características a nivel 3 al nodo emulado 1. Más concretamente, a los interfaces SVI configurados en dicho nodo que nos permitirán implementar *InterVLAN routing* (conmutación a nivel 3 entre VLAN diferentes).

Además, sobre este escenario, se configura el protocolo VTP, propietario de *Cisco Systems*, que permite realizar las tareas de configuración y gestión VLAN de un modo más sencillo. Más información sobre este protocolo y su funcionalidad puede encontrarse en [5]. Aunque la arquitectura implementada aquí es sencilla, la funcionalidad de este protocolo está especialmente indicada en topologías conmutadas más extensas. Así, se configura el rol de servidor VTP en el *nodo 1*. Al *nodo 2* se le asigna la funcionalidad de cliente VTP.

Un ejemplo de estas configuraciones se muestra en la Fig. 8, capturada directamente de *GNS3/Dynamips*. En ella vemos la configuración en el nodo emulado 1 del enlace troncal, en concreto la asignación física particular de diseño de los interfaces *Fast Ethernet* a cada VLAN, así como la configuración del dominio y funcionalidad VTP. Como vemos, el alumno puede, mediante comandos *Cisco IOS*, configurar los parámetros del protocolo del mismo modo que se implementan en el correspondiente hardware de red real.

```
Dynamips(3): R1, Console port
switch_router1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch_router1(config)#interface fastethernet 1/0
switch_router1(config-if)#switchport mode trunk
switch_router1(config-if)#switchport trunk encapsulation dot1q
switch_router1(config-if)#exit
switch_router1(config)#interface range fastethernet 1/1 - 8
switch_router1(config-if-range)#switchport mode access
switch_router1(config-if-range)#switchport access vlan 2
switch_router1(config-if-range)#exit
switch_router1(config)#interface range fastethernet 1/9 -15
switch_router1(config-if-range)#switchport mode access
switch_router1(config-if-range)#switchport access vlan 3
switch_router1(config-if-range)#exit
switch_router1(config)#exit
switch_router1#vlan
*Mar 1 00:11:43.419: %SYS-5-CONFIG_I: Configured from console by c
onsole
switch_router1#vlan database
switch_router1(vlan)#vtp domain proyecto
Changing VTP domain name from NULL to proyecto
switch_router1(vlan)#vtp server
```

Fig. 8. Configuración implementada en el nodo 1 de la arquitectura LAN *Ethernet*/VTP.

Como ejemplo del realismo de las pruebas, y de la capacidad del sistema para comportarse de manera idéntica a los equipos reales, mostramos ahora (Fig. 9) el análisis de la trayectoria del tráfico en la comunicación de equipos adscritos a una misma VLAN, pero en nodos diferentes (traza

verde, conmutación a nivel 2 vía enlace troncal), así como la trayectoria usada para comunicar equipos pertenecientes a VLAN diferentes, pero en el mismo nodo emulado (conmutación a nivel 3 vía enlace troncal, traza azul).

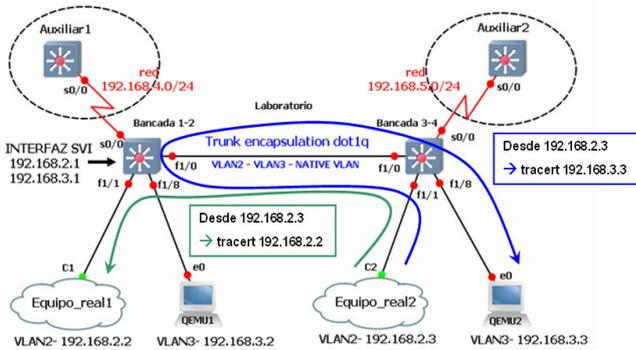


Fig. 9. Análisis de la trayectoria del tráfico de red para distintos tipo de comunicación en la arquitectura LAN Ethernet/VTP.

Los resultados de estos ensayos, consecuentes con las configuraciones implementadas, se observan en la Fig. 10. Vemos que la conmutación a nivel 2 entre equipos pertenecientes a la misma VLAN sólo implica un salto, mientras que la conmutación a nivel 3 conlleva dos: el primero para alcanzar el interfaz virtual SVI en el nodo emulado 1, y el segundo para alcanzar el *host* final desde dicho interfaz virtual. En ambos casos se hace uso del enlace troncal. De esta manera, el entorno emulado permite en este caso ayudar al alumno a comprender la diferencia entre la conmutación a nivel 2 y nivel 3, y la funcionalidad de las VLAN.

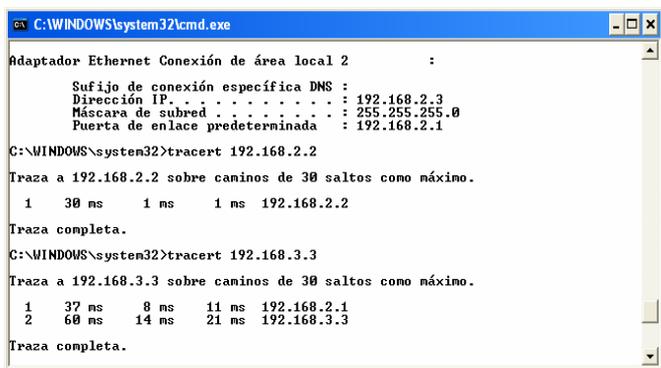


Fig. 10. Resultados de los análisis del rigor funcional de GNS3/Dynamips.

**C. Ejemplo práctico avanzado de interoperación entre los entornos de red real y emulado**

Se plantea, en base a las arquitecturas anteriores, la implementación de un servicio telemático cliente/servidor FTP para la transmisión pasiva y anónima [6] de un fichero de 250 KB. Como muestra la Fig. 11 (por ser la topología similar en las dos pruebas, sólo se muestra la arquitectura LAN Ethernet/VTP), tanto el cliente como el servidor FTP son equipos reales y externos al entorno emulado de GNS3/Dynamips, e interoperan con él. El cliente y el servidor se encuentran conectados a nodos emulados diferentes.

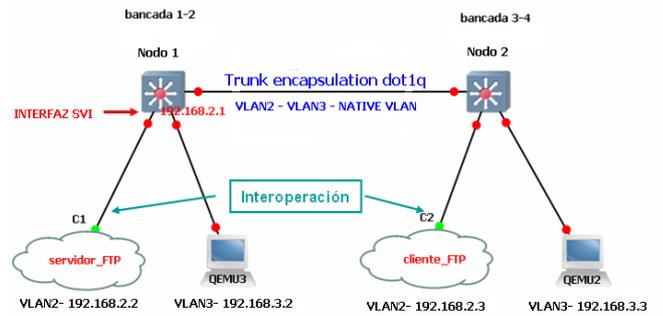


Fig. 11. Esquema de la implementación de un servicio FTP cliente/servidor usando la interoperación del entorno real con el entorno emulado.

Con el objetivo de hacer comprender al alumno la importancia del tamaño de la trama en la transmisión de un fichero, la transmisión se puede realizar con diferentes longitudes de trama Ethernet: 64, 512, 1.024 y 1.514 bytes. En ambas arquitecturas de estudio, la conexión y la transmisión del fichero se puede realizar correctamente, salvo para la longitud de trama de 64 bytes. Esta última es demasiado pequeña como para contener toda la información de las cabeceras añadidas por los protocolos necesarios para el funcionamiento de FTP.

De los resultados obtenidos (Tabla I) se deduce que en este escenario el valor de 1.024 bytes obtiene el menor tiempo de transmisión. El valor de trama de 1.514 bytes parece evidenciar una cierta congestión en el enlace serie ppp, posiblemente relacionada con la diferencia de ancho de banda entre *Fast Ethernet* (en la conexión entre el equipo real y el nodo emulado), y el enlace serie. Para analizar estos resultados se ha realizado la captura del tráfico asociado a la transmisión en el enlace WAN ppp, que el alumno puede visualizar y analizar con *Wireshark*.

TABLA I  
RESULTADOS OBTENIDOS EN LA TRANSMISIÓN FTP PASIVA Y ANÓNIMA CLIENTE/SERVIDOR PARA LA ARQUITECTURA WAN PPP.

	trama 64 bytes	trama 512 bytes	trama 1.024 bytes	trama 1.514 bytes
<i>Longitud trama ACK</i>	-	60 bytes	60 bytes	60 bytes
<i>Nº tramas totales en la transmisión (información y control)</i>	-	986	477	361
<i>Nº tramas de información FTP_DATA</i>	-	560 + 1 trama de 419 bytes	257 + 1 trama de 139 bytes	171 + 1 trama de 1.129 bytes
<i>Nº tramas de información FTP_DATA retransmitidas</i>	-	9	10	9
<i>Δ tiempo total en la transmisión (información)</i>	-	13,703 seg.	9,114 seg.	11,36 seg.

En el caso de la arquitectura LAN Ethernet/VTP, la transmisión FTP del fichero se realiza entre equipos reales adscritos a una misma VLAN, pero conectados a nodos emulados distintos. Por consiguiente, la transmisión del

archivo implica en este caso que la conmutación se haga a nivel 2 mediante el enlace troncal (*trunk link*). Por ello, y como muestra la Tabla II, los tiempos de transmisión son considerablemente más bajos que en la arquitectura anterior.

Del mismo modo, en este caso sí puede considerarse como longitud óptima de trama *Ethernet* 802.3 el valor de 1.514 bytes. Debido a la información añadida por el protocolo IEEE 802.1Q (también llamado *dot1q*), este tamaño se corresponde con un valor de trama Ethernet 802.1Q de 1.518 bytes en el enlace troncal. Estos 4 bytes añaden la información necesaria para identificar *tagged* y *untagged frames* (tramas etiquetadas y no etiquetadas). El protocolo 802.1Q no encapsula la trama original, sino que únicamente añade 4 bytes al encabezado *Ethernet*. El valor del campo *EtherType* cambia a 0x8100, para así señalar el cambio en el formato de la trama. Además, 802.1Q obliga a recalcular el campo FCS [5].

TABLA II  
RESULTADOS OBTENIDOS EN LA TRANSMISIÓN FTP PASIVA Y ANÓNIMA  
CLIENTE/SERVIDOR PARA LA ARQUITECTURA LAN ETHERNET/VTP

	trama 64 bytes	trama 512 bytes	trama 1.024 bytes	trama 1.514 bytes
<i>Longitud trama ACK</i>	-	66 bytes	66 bytes	66 bytes
<i>Nº tramas totales en la transmisión (información y control)</i>	-	966	47	357
<i>Nº tramas de información FTP_DATA</i>	-	561 + 1 trama de 305 bytes	259 + 1 trama de 919 bytes	172 + 1 trama de 453 bytes
<i>Nº tramas de información FTP_DATA retransmitidas</i>	-	24	21	7
<i>Δ tiempo total en la transmisión (información)</i>	-	1,3 seg.	1,03 seg.	0,43 seg.

## V. ANÁLISIS DEL RENDIMIENTO DEL SISTEMA

Una característica común a todas las plataformas software de emulación es que el consumo de recursos del sistema emulador puede llegar a ser muy elevado, sobre todo en términos de carga de CPU y memoria RAM. Este hecho puede llegar a condicionar la escalabilidad de las arquitecturas a emular.

Para disminuir el consumo de memoria RAM, *GNS3/Dynamips* dispone de varios optimizadores, entre los que destaca *ghost ios*. Esta utilidad mapea una región de memoria compartida por los dispositivos emulados que ejecutan el mismo sistema *Cisco IOS*, en lugar de copiar el sistema tantas veces como dispositivos se estén emulando. De esta manera se evita que cada dispositivo necesite su propia región de memoria. Por lo tanto, este recurso es óptimo si todos los dispositivos de nuestra arquitectura son similares. Con el uso de *ghost ios*, el consumo de memoria RAM deja de ser crítico para la emulación, siempre que el PC emulador disponga de la suficiente memoria RAM (el mínimo aconsejable son 2 GB).

No ocurre lo mismo con la carga de CPU, que se convierte en el principal límite en la emulación de topologías extensas. A pesar de configurar inicialmente el parámetro *idle PC* [4] al iniciar el diseño de una arquitectura, y así optimizar el rendimiento del procesador en la emulación, la carga de este último aumenta considerablemente conforme se incrementa el número de nodos emulados.

Los ensayos realizados en el laboratorio sobre un PC de gama tecnológica media (procesador de un solo núcleo) emulando las arquitecturas previamente consideradas, se resumen en la Tabla III. Para analizar el consumo de recursos, se consideran activos los nodos *Auxiliar 1* y *2*, que se configuran mediante comandos *Cisco IOS*. Cabe destacar que estos nodos emulados no tienen ningún propósito telemático especial de diseño: su objetivo es solamente evaluar la repercusión del número de nodos emulados activos sobre el rendimiento y el consumo de recursos del sistema emulador. Así podremos evaluar, de un modo aproximado, la potencial escalabilidad del diseño de la arquitectura emulada.

Se ha observado que los resultados se estabilizan tras un pequeño transitorio inicial, una vez iniciamos los equipos que participan en el escenario. Este tiempo es necesario para el inicio del sistema operativo *Cisco IOS* y las configuraciones asociadas a las arquitecturas de red. En los resultados de la Tabla III, las expresiones “XX→XX” deben interpretarse como el descenso en el consumo de recursos en el PC emulador, cuando los recursos optimizadores (principalmente *ghost ios*) se activan.

TABLA III  
CONSUMO DE RECURSOS DEL SISTEMA EMULADOR OBTENIDO

<i>Nodos activos</i>	<i>Carga de CPU (offmode)</i>	<i>RAM en uso en MB (offmode)</i>	<i>Carga media de CPU (pseudo- activo)</i>	<i>RAM en uso en MB (pseudo- activo)</i>
<i>1 nodo</i>	1% max.	318	1-2 %	780 → 540
<i>2 nodos</i>	1% max.	318	4 % → 2-3%	940 → 720
<i>2 nodos + 1 QEMU</i>	1% max.	318	16 % → 14%	950 → 750
<i>2 nodos + 2 QEMU</i>	1% max.	318	17 % → 15%	960 → 760
<i>2 nodos + 1 Aux (no QEMU)</i>	1% max.	318	6 % → 4-5%	970 → 760
<i>2 nodos + 2 Aux (no QEMU)</i>	1% max.	318	11 % → 10%	980 → 790

No debe confundirse el descenso en la carga de CPU en este caso con el descenso debido a la configuración del parámetro *idle PC* (parámetro a configurar en los nodos emulados al iniciar el diseño de una arquitectura). El descenso es atribuible a que el sistema se estabiliza tras varios minutos, una vez se han cargado completamente los sistemas *Cisco IOS* y las configuraciones de los dispositivos, y posteriormente los recursos optimizadores de memoria RAM están activos y se ha configurado la región de memoria compartida para los dispositivos. El estado *offmode* es el estado del PC antes de iniciar el software emulador.

Una vez iniciado *GNS3/Dynamips*, el estado *pseudo-activo* se refiere al tiempo en que los nodos sólo intercambian información de los protocolos a nivel 2 y 3. Igualmente es apreciable cómo el uso de *host* virtuales Linux QEMU eleva significativamente la carga de CPU, respecto a escenarios que no los incluyen (para ello, comparamos las filas 2 y 3 de la Tabla III). Sin embargo, conforme incorporamos más *host* virtuales QEMU, el consumo crece, pero en mucha menor medida (resultado comparativo de las filas 3 y 4).

Finalmente, y centrando la atención únicamente en los nodos emulados *switch-router* de nuestras arquitecturas anteriores (quedan por tanto excluidas las filas 3 y 4 de la Tabla III), podemos “aproximar” la tendencia del consumo de CPU a partir de dos nodos emulados con una progresión geométrica de razón dos (incidimos en la connotación de cálculo aproximado), que nos permitiría implementar arquitecturas de hasta 7 nodos emulados. Esta cota máxima de potencial *escalabilidad* implicaría una carga de CPU entre el 70% y el 90%. Acerca de este consumo, no debe olvidarse la circunstancia de que nuestro sistema emulador es un PC de gama tecnológica media. Sobre PC emuladores de gama tecnológica alta es esperable una mayor capacidad de incluir máquinas emuladas.

VI. ANÁLISIS DE PARÁMETROS DE RED EN LA INTEROPERACIÓN ENTRE LOS ENTORNOS REAL Y EMULADO

En la sección anterior se ha contrastado mediante pruebas de laboratorio que la funcionalidad a nivel 2 y 3 del emulador es rigurosa. Vemos confirmado, por tanto, que los nodos emulados dentro de *GNS3/Dynamips* pueden interoperar con equipos reales en la red, a través de interfaces *Ethernet* disponibles en el PC emulador [4]. Este hecho da la posibilidad de que el alumno pueda utilizar hardware emulado sin la necesidad de disponer realmente del equipo.

En esa sección analizaremos la capacidad del entorno para emular con realismo ciertos parámetros de red que pueden ser críticos en la prestación de servicios telemáticos. Aunque la funcionalidad del hardware emulado sea idéntica al del real, existirán de hecho algunas diferencias en el comportamiento, como por ejemplo, valores diferentes en los retardos añadidos por los dispositivos. No olvidemos que existen servicios en los que el retardo es crítico (por ejemplo, Voz sobre IP, VoIP), y para los que la ITU-T define unos valores máximos [7].

Estas particularidades deben ser adecuadamente interpretadas por el alumno, que deberá distinguir los retardos asociados al hardware real de los que se deriven de utilizar un entorno emulado. Se han realizado en el laboratorio, y utilizando las arquitecturas previamente presentadas, dos tipos de análisis para medir las diferencias entre un entorno real y uno mixto incluyendo máquinas reales y emuladas. En ambos se han implementado peticiones de eco desde el *equipo real 1* de los diseños (ver Fig. 2 y Fig.7), evaluándose el retardo de ida y vuelta (*Round Trip Time, RTT*) a través del protocolo ICMP (instrucción *ping*).

Los análisis se han realizado para diferentes cargas de trabajo del PC emulador, utilizando para ello los nodos *Auxiliar 1* y 2 de las arquitecturas. En consecuencia, el valor del *Round Trip Time* reflejará la latencia mínima a considerar en el escenario real-emulado de trabajo, ya que el protocolo ICMP no implementa control de errores en la transmisión.

Para el primero de los análisis, la petición de eco tiene como destino el interfaz virtual SVI configurado en el *nodo emulado 1* (recordemos que este interfaz sirve para implementar *InterVLAN routing*). El segundo de los análisis tiene como destino el *equipo real 2*, que interopera con el *nodo emulado 2*. En este segundo análisis deberemos “atravesar”, por lo tanto, la arquitectura emulada, y recibir la petición de eco en el *equipo real 2*. Estos ensayos se reflejan en la Fig. 12, para la arquitectura LAN *Ethernet/VTP* (el esquema sería similar para la arquitectura WAN ppp). Así mismo, los análisis se han realizado para las longitudes de trama Ethernet consideradas en apartados anteriores: 64, 512, 1.024 y 1.514 bytes.

Los valores de la latencia para el primero de los experimentos son muy elevados, y similares para las dos arquitecturas. Estos valores se muestran en la Tabla IV.

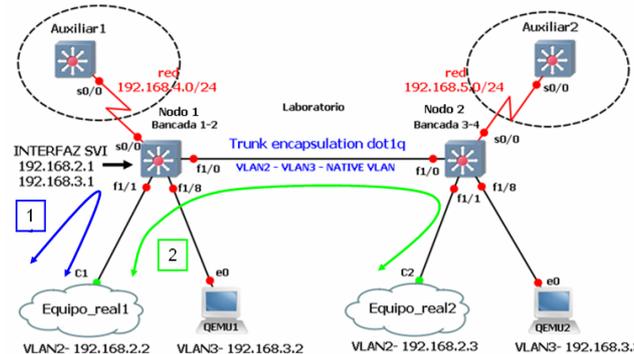


Fig. 12. Detalle de los ensayos implementados en el análisis de la latencia de red en la interoperación entre los entornos de red real y emulado.

TABLA IV  
VALORES OBTENIDOS EN EL PRIMER EXPERIMENTO DE LA LATENCIA DE RED

RTT promedio (milisegundos)	1 nodo activo	2 nodos activos	3 nodos activos	4 nodos activos
Trama de 64 bytes	32 → 30	40 → 34	47 → 40	61 → 49
Trama de 512 bytes	34 → 31	41 → 35	48 → 42	62 → 51
Trama de 1.024 bytes	37 → 32	44 → 37	51 → 45	67 → 54
Trama de 1.514 bytes	39 → 33	45 → 39	52 → 46	67 → 55

Para el segundo de los experimentos, en el caso de la arquitectura WAN ppp los resultados son igualmente elevados, debido a que existe una comunicación a nivel 3 entre los equipos reales a través de la arquitectura emulada.

Y aunque estos valores cumplen de hecho las recomendaciones de retardo máximo de la ITU-T [7], no son directamente trasladables a un escenario de red real similar, ya que nuestra conexión *extremo-a-extremo* sólo incluye dos nodos emulados.

En el caso de la arquitectura LAN *Ethernet/VTP* los resultados son más acordes a lo esperable en un entorno real análogo, como muestra la Tabla V. En este caso, la petición de eco sólo requiere conmutación a nivel 2 entre los equipos

reales, por estar adscritos a la misma VLAN en nodos emulados diferentes.

TABLA V  
VALORES DE LATENCIA OBTENIDOS EN EL SEGUNDO EXPERIMENTO PARA LA ARQUITECTURA DE RED LAN ETHERNET/VTP.

RTT promedio (milisegundos)	1nodo activo	2 nodos activos	3 nodos activos	4 nodos activos
Trama de 64 bytes	-	2 → 1-2	3 → 2	3 → 2
Trama de 512 bytes	-	2 → 1-2	3 → 2	4 → 3
Trama de 1.024 bytes	-	2 → 1-2	3 → 2	5 → 4
Trama de 1.514 bytes	-	3 → 2-3	4 → 3	6 → 5

A partir de los resultados obtenidos, podemos interpretar que se da un comportamiento correcto de la latencia a nivel 2, pero un comportamiento poco realista cuando se ejecutan en *GNS3/Dynamips* procesos de conmutación a nivel 3. Estos límites en el realismo del entorno emulador deberán ser tenidos en cuenta por el alumno, sabiendo que la conmutación a nivel 3 añade unos retardos adicionales, que estarán en función de la capacidad de proceso de la máquina en la que se ejecute el entorno.

Tal como se mencionó con anterioridad, focalizando el uso de *GNS3/Dynamips* como herramienta académica, el comportamiento de la latencia no resulta decisivo ni condiciona el análisis de arquitecturas de red y la funcionalidad de los protocolos a nivel 2 y 3. Pero puede resultar interesante, bajo el prisma docente, aprovechar los ensayos anteriores como base para incidir en la importancia del análisis, comportamiento y restricciones de la latencia y de otros parámetros de red.

## VII. CONCLUSIONES

En este trabajo se han presentado una serie de análisis sobre el uso de entornos de pruebas mixtos, integrando máquinas reales y emuladas, en el ámbito de la docencia en Ingeniería Telemática. En concreto, se han realizado diferentes pruebas con el entorno software *GNS3/Dynamips*, como una herramienta adecuada en el ámbito académico, con un claro enfoque hacia el mundo profesional.

Se ha contrastado la adecuada usabilidad del software, a través de su intuitivo interfaz gráfico. A pesar de que actualmente este emulador se circunscribe a entornos corporativos con hardware de *Cisco Systems* y *Juniper*, esto no es inconveniente para diseñar y analizar arquitecturas de red más generales. Las configuraciones de estos dispositivos emulados se realizan de forma idéntica a como se implementarían mediante el puerto consola del hardware real correspondiente. Esta ventaja permite al alumno realizar pruebas con escenarios de red reales sin necesidad de disponer de los dispositivos físicos, cuya disponibilidad puede ser limitada. Igualmente, esta circunstancia es una ventaja manifiesta a la hora de plantear posibles escenarios de estudio.

Añadidas a las ventajas anteriores, la capacidad del entorno emulador de interoperar con escenarios de red reales amplía enormemente las posibilidades docentes para el

análisis de escenarios y servicios telemáticos, teniendo en cuenta que las capturas de tráfico emuladas se pueden analizar mediante *Wireshark*.

Se han presentado pruebas, mostrando que la funcionalidad a nivel 2 y 3 del emulador es correcta y rigurosa con análogos escenarios de red reales. También se ha detectado que el sistema añade retardos diferentes de los que se obtendrían en un entorno real, especialmente si el número de nodos es elevado y se usa conmutación a nivel 3.

Como trabajo futuro se considera el uso de recursos optimizadores más complejos de *GNS3/Dynamips*, que pueden aumentar su escalabilidad. Tal es el caso del recurso *Hypervisors*, que permiten repartir la carga de los procesos de emulación entre varias máquinas, aumentando así las posibilidades de diseño y la escalabilidad de las arquitecturas de estudio.

## AGRADECIMIENTOS

Este trabajo ha sido parcialmente subvencionado por el Fondo Social Europeo en colaboración con el Gobierno de Aragón, el proyecto CPUFLIPI (MICINN TIN2010-17298) del Gobierno de España así como por la Cátedra Telefónica-Universidad de Zaragoza.

## REFERENCIAS

- [1] S. Gimeno Martínez. "Evaluación de plataformas virtuales: Estudio comparativo". Tesis de Máster. Universidad Politécnica de Valencia. Escuela Técnica Superior de Informática Aplicada. Valencia, Diciembre de 2008.
- [2] J. Martínez, J.J. Ortega, J.A. Fernández. "Laboratorios virtuales de redes: sí, inténtelo en casa". JENUI 2011. Sevilla. Julio de 2010.
- [3] J. M. Ribadeneyra Sicilia. "Prácticas de redes en entorno real en laboratorios de propósito general". XVI Jornadas de Enseñanza Universitaria de la Informática. Santiago de Compostela: Universidade de Santiago de Compostela. Escola Técnica Superior d'Enxerñaría, p. 155-160, 2010.
- [4] Graphical Network Simulation – GNS3- Documentation (n.d). Recuperado en octubre de 2012 de <http://www.gns3.net/documentation>
- [5] Cisco Systems, Inc. (2006). *Cisco Networking Academy program, CCNP: Building Multilayer Switching Networks v.5.0*, San Jose, USA: Americas Headquarters.
- [6] File Transfer Protocol (n.d.). En *Wikipedia*. Recuperado en Diciembre de 2012 de [http://es.wikipedia.org/wiki/File\\_Transfer\\_Protocol](http://es.wikipedia.org/wiki/File_Transfer_Protocol)
- [7] Unión Internacional de Telecomunicaciones (n.d.). *Normalización UIT-T*. Recuperado en Febrero de 2013 de <http://www.itu.int/ITU-T/sitemap/index.asp>