

# Using Identity Provider and Automatic Resource Management to Improve a Remote Networking Lab

T. J. Mateo Sanguino, I. J. Fernández de Viana González, J. Espejo Fernández, and A. García Domínguez

**Abstract**— Scientific research and development on remote laboratories has spectacularly increased since the past decade due to the limitations of the traditional lab sessions (e.g., high cost of equipment, scarce flexibility, limited student development, etc.). To this extent, we describe the developments carried out to improve the management and remote access of a networking laboratory —called Cisolab— deployed at the University of Huelva (Spain). The goal of these improvements have been to automatically manage and share hardware resources among remote students, while providing a high-level security to maintain an e-learning laboratory for educational purposes. The contribution to this field consisted in integrating an institution's Identity Provider (IdP) into a Meeting Room Booking System (MRBS) to provide a safe and deterministic access to the remote equipment by authorized students. In addition, we developed an automated maintenance system including periodic backups in the cloud and scheduled reboots of the networking systems. As a result of the lab automation, we have been able to track the student work. This helped us to detect the students' strengths and weaknesses by analyzing their command history during their practice sessions. So the experimentation was aimed at presenting the first results after the interaction of a group of students with the remote lab during a semester of evaluation.

**Keywords**— Learning management system, Networking lab, Remote lab, Student performance, Teaching lab.

## I. INTRODUCCIÓN

EL elevado coste de equipos de comunicación impide, en general, la preparación de maquetas de laboratorio con topologías de red distintas para cada práctica o usuario, lo que supondría el escenario ideal para el aprendizaje de conceptos en asignaturas de redes de datos. Por otro lado, los estudiantes no pueden desarrollar sus habilidades fuera del aula y toda su experiencia se reduce, habitualmente, a las sesiones prácticas de laboratorio durante el periodo docente. Estas limitaciones han llevado a distintas universidades a concentrar sus esfuerzos en el diseño de nuevos entornos interactivos de enseñanza-aprendizaje [1].

Para dar solución a este problema, herramientas como VirtualBox, GNS3 y Dynamips han sido usadas por la Universidad Nacional de Córdoba (Argentina) para crear topologías de red complejas y emular routers sobre máquinas virtuales [2]. La principal desventaja de esta solución no radica en el número de usuarios que la pueden usar, sino en los requerimientos de procesamiento y memoria inherentes a la virtualización que hacen necesario la adquisición de servidores específicos. Otros proyectos como RECOLAB o CIBNOR-ITLP se centran en proporcionar acceso remoto a laboratorios de docencia donde se encuentran dispositivos electromecánicos a controlar [3], [4]. Para lograrlo, los usuarios establecen una conexión remota con un servidor y, desde él, hacia cada uno de los dispositivos del laboratorio gracias a un applet de Java™ y una serie de formularios HTML. Siguiendo con este mismo enfoque, VirtualLab usa applets de Java™ y código HTML para proporcionar a los estudiantes topologías de red personalizadas a través de diferentes laboratorios físicos sin que los usuarios se percaten de ello y hacerlos remotamente accesibles a diversas instituciones educativas [5]. Con un objetivo más amplio, DiscoverLab se implementó como un laboratorio para estudiantes de todo el mundo que permite a los usuarios realizar experimentos reales en diferentes áreas de conocimiento (e.g., Biología, Física, Química o Electrónica) desde ordenadores localizados en cualquier lugar usando infraestructuras de bajo coste (i.e., applets de Java™ y código HTML) [6]. Siguiendo esta misma idea, WebLab-Deusto se integró en la red internacional LabsLand para crear laboratorios altamente configurables en los que compartir dispositivos propios (e.g., FPGAs, robots o circuitos analógicos/digitales) a través de una API y un entorno de aprendizaje basado en la Web (i.e., Moodle). Su principal inconveniente es que requiere la instalación de una infraestructura software para soportar las características y ajustar su rendimiento [7]. Más recientemente, SARLAB se ha desarrollado para apoyar la enseñanza mediante el acceso a equipos físicos en nuevas áreas de conocimiento emergentes (e.g., Realidad Aumentada o Robótica). Para hacerlo posible y seguro, los usuarios se comunican con un servidor remoto a través de conexiones VPN, applets de Java™ programados con el entorno Easy Java Simulations (EJS) y un servicio M2M denominado Java™-Internet-LabVIEW™ (JIL) o Java™-Internet-Matlab® (JIM) [8]. Dentro de este mismo campo se desarrolló un laboratorio remoto dinámico basado en la aplicación Remote Network Laboratory de Cisco Systems, Inc. para proporcionar distintas topologías sobre las que realizar diversos ejercicios prácticos. Para hacerlo posible, un servidor configura automáticamente las VLAN en un switch

T.J. Mateo Sanguino, Universidad de Huelva, Dpto. Ingeniería Electrónica, Sistemas Informáticos y Automática, Ctra. Huelva-La Rábida S/N, 21819 Palos de la Frontera, Huelva, España, tomas.mateo@diesia.uhu.es

I.J. Fernández de Viana González, Universidad de Huelva, Dpto. Tecnologías de la Información, Ctra. Huelva-La Rábida S/N, 21819 Palos de la Frontera, Huelva, España, i.fviana@dti.uhu.es

J. Espejo Fernández, Universidad de Huelva, Dpto. Ingeniería Electrónica, Sistemas Informáticos y Automática, Ctra. Huelva-La Rábida S/N, 21819 Palos de la Frontera, Huelva, España, jennifer.espejo@alu.uhu.es

A. García Domínguez, Universidad de Huelva, Dpto. Tecnologías de la Información, Ctra. Huelva-La Rábida S/N, 21819 Palos de la Frontera, Huelva, España, antonio.gdominguez@alu.uhu.es

Corresponding author: T.J. Mateo Sanguino

en el que usuarios y dispositivos interactúan de acuerdo con un sistema de reservas, mejorando así la administración del laboratorio. Sin embargo, se requiere un despliegue complejo y electrónica especial para el control del laboratorio [9]. También, el Laboratorio Remoto en Automática fue desarrollado para realizar prácticas en tiempo real con dispositivos físicos y virtuales basados en JavaScript, CSS y HTML. Como principal inconveniente, carece de un diseño web adaptable [10]. Por último, NetLab se implementó con la ayuda de software propietario para asistir a los estudiantes en la obtención de certificaciones profesionales de Cisco. Como principal ventaja, soporta el aprendizaje colaborativo. Ello permite que un grupo de estudiantes trabajen simultáneamente en el mismo equipo físico y facilita que el profesor participe interactivamente en las sesiones del alumno (e.g., siendo un mero observador, haciendo recomendaciones específicas o incluso tomando el control completo). Sin embargo, su principal desventaja es el alto coste debido a que la gestión del laboratorio es realizada por una empresa externa [11].

Este artículo está organizado como se indica a continuación. Tras una breve revisión del estado del arte, la Sección 2 presenta el laboratorio remoto docente llevado a cabo, objetivo fundamental de este trabajo. Las distintas subsecciones describen su infraestructura de red incluyendo el portal Web, el sistema de monitorización, el sistema de reservas y el sistema de mantenimiento automático que hacen posible las prácticas telemáticas. La Sección 3 describe un caso de estudio con el objetivo de mostrar el tipo de sesiones prácticas remotas que se pueden realizar. La Sección 4 describe la experimentación llevada a cabo y analiza el impacto de las distintas tecnologías implementadas sobre el rendimiento del laboratorio telemático y los estudiantes. Finalmente, el trabajo concluye con los resultados y trabajos futuros.

## II. LABORATORIO REMOTO CISCOLAB

Este proyecto surge de la iniciativa de un equipo de

profesores como parte de un proyecto educativo más amplio, que ha sido ampliamente descrito en [1], [12]-[14]. El sistema propuesto ha sido probado en un laboratorio de redes de la Universidad de Huelva (UHU), Huelva (España), cuyo objetivo final es desplegar un laboratorio de e-learning en Ingeniería Informática para realizar prácticas a distancia con hardware real. Este laboratorio remoto —disponible en [www.uhu.es/diesianetworking](http://www.uhu.es/diesianetworking)— tiene como objetivo mejorar varios aspectos de los laboratorios mencionados en la sección de Introducción. Con tal fin, en la Tabla 1 se muestran las características y capacidades de nuestro laboratorio, Ciscolab, en comparación con otros trabajos realizados en el campo de los laboratorios virtuales y/o remotos. Las principales ventajas del sistema propuesto sobre otros son: *i)* mejora de la seguridad al prescindir de applets de Java™ y delegar la autenticación, autorización y provisión de información en el proveedor de identidad institucional; *ii)* infraestructura de bajo coste debido al uso exclusivamente de software libre de código abierto; *iii)* escalabilidad para agregar nuevos dispositivos de red; *iv)* mantenimiento mejorado debido a copias de seguridad configurables y scripts automáticos para reiniciar los sistemas; *v)* gestión mejorada gracias al control de acceso, trazabilidad de los usuarios e informes sobre el estado del hardware; *vi)* mejor experiencia de usuario debido a diagramas interactivos y un diseño web adaptativo; y *vii)* capacidad para detectar las dificultades del estudiante mediante el registro histórico de las sesiones prácticas de laboratorio.

### A. Infraestructura de red

La Figura 1 muestra un esquema de toda la infraestructura de red. Consiste en una red corporativa en la UHU (izquierda) comunicada a través de Internet con el laboratorio de docencia localizado en un campus diferente (centro). Este último contiene dos racks de comunicaciones (1x router Cisco 2621XM, 1x hub Conceptronic, 2x servidores Ubuntu y 2x UPS) así como las maquetas de trabajo de los estudiantes

TABLA I  
CARACTERÍSTICAS DE LOS LABORATORIOS VIRTUALES Y REMOTOS EN INGENIERÍA INFORMÁTICA

Nombre	Requerimientos del Laboratorio	Requerimientos del Usuario	Método de Interacción	Año de la Versión
Univ. Nacional de Córdoba	Servidor potente con VirtualBox, Dynamips y dispositivos virtuales	Navegador web	GNS3 y HTML	2013
RECOLAB	Servidor y dispositivos remotos	Navegador web y soporte Java™	Applets y HTML	2005
CIBNOR-ITLP	Servidor y dispositivos remotos	Navegador web y soporte Java™	Applets y HTML	2010
Virtlab	Servidor y dispositivos remotos	Navegador web y soporte Java™	Applets y HTML	2009
DiscoverLab	Servidor y dispositivos remotos	Navegador web y soporte Java™	Applets y HTML	2007
WebLab-Deusto	Servidor propietario, API, Moodle y dispositivos remotos	Navegador web y soporte JavaScript	HTML	2016
SARLAB	EJS, servidor JIL y/o servidor JIM	Navegador web, soporte Java™ y VPN	Applets y HTML	2016
Remote Network Laboratory	Servidor, VLANs en switches y sistema UPS	Navegador web y soporte JavaScript	HTML	2008
Lab. Remoto de Automática	Dispositivos virtuales y remotos	Navegador web y soporte JavaScript	HTML y CSS	2013
NetLab	Servidor NDG y dispositivos remotos	Navegador web y soporte JavaScript	HTML	2011
Ciscolab	IdP, Moodle, MRBS, Nagios®, MK Livestatus, AnyTerm, SSHPass, Bootstrap y dispositivos remotos	Navegador web y soporte JavaScript	C++, PHP HTML5, CSS3, AJAX JavaScript y Canvas	2016

(derecha). La funcionalidad de los distintos sistemas es la siguiente: el servidor Web corporativo de la UHU hospeda el portal Web del laboratorio que conecta con el entorno de enseñanza remoto, el proveedor de identidad (IdP) se encarga de provisionar la información de usuario para validar su acceso (i.e., tanto personal como de las asignaturas matriculadas), el sistema de gestión del aprendizaje (LMS) es responsable de gestionar la inscripción automática de las asignaturas de los usuarios en el laboratorio, el servidor MRBS se encarga de reservar los recursos de red y validar el acceso de los estudiantes al laboratorio de enseñanza a través de los servicios LMS e IdP, el servidor Nagios® se encarga de monitorizar los routers y switches del laboratorio (i.e., estado del dispositivo e interfaces de conexión), y el servidor MySQL alberga la base de datos responsable de registrar las reservas de los usuarios en el servicio MRBS. Todos estos elementos están configurados para asegurar una alta disponibilidad de tipo activo/pasivo ya sea mediante clúster de servidores Web o de base de datos.

De acuerdo con la Figura 1, el proceso de acceso telemático que realizan los usuarios es el siguiente: (1) el estudiante accede desde su ordenador a la página de inicio del laboratorio en el servidor Web de la universidad, la cual permite mostrar tanto el estado de una maqueta de trabajo como reservarla; (2) el servidor Web envía las solicitudes de estado y/o reserva al servidor MRBS del laboratorio docente; (3) el servidor MRBS realiza las consultas oportunas al servidor MySQL, el cual devuelve información de la maqueta (e.g., imagen personalizada, texto y enlaces con ayuda adicional); (4) el servidor MRBS envía simultáneamente una petición al servidor Nagios®, el cual informa del estado de la maqueta; (5) el servidor MRBS recibe y reenvía los datos sobre la disponibilidad de la maqueta hacia el servidor Web; (6) el servidor Web muestra la información de estado de la maqueta tanto si está disponible como si no; (7) en el momento de realizar la reserva, el usuario se autentica mediante el IdP de la universidad, el cual comprueba el permiso del usuario; (8) el servidor IdP solicita al servidor LMS las asignaturas en las que el usuario está matriculado y verifica si estas están inscritas en la relación de asignaturas que pueden hacer uso del laboratorio; (9) el usuario puede acceder desde su navegador al dispositivo remoto y usar los terminales interactivos durante el tiempo de la reserva.

Para implementar esta funcionalidad se han utilizado las siguientes herramientas: el servidor Web está basado en la distribución Apache®. La comunicación entre este y el servidor MRBS que se detalla en los pasos (1) a (5) se ha resuelto mediante la implementación de servicios Web localizados en el servidor MRBS. Dichos servicios se invocan

con una llamada AJAX gracias al método POST soportado por HTTP y devuelven la información en formato JSON. El servicio IdP se ha implementado utilizando SimpleSAMLphp, lo que nos permite trabajar con los principales protocolos de autenticación (e.g., SAML, OpenID y OAuth2). El servicio LMS se basa en Moodle y ha sido modificado para dar soporte a la autenticación de usuarios y matriculación de asignaturas mediante el protocolo SAML 2.0 incluido en la extensión SAML. Los servidores Ubuntu han sido instalados con la versión 10.04.4 para alojar los servidores MRBS, MySQL y Nagios®. MRBS es una aplicación Web basada en GPL que fue modificada para reservar los recursos de red, validar el acceso de los estudiantes al laboratorio docente y acceder de forma remota a las maquetas a través de conexiones SSH. MySQL es la base de datos de Oracle implementada para registrar las reservas de los usuarios mediante consultas SQL desde el servidor MRBS. Nagios® es un software de código abierto en el que se instaló el complemento MK Livestatus para supervisar los routers y switches del laboratorio. Para más información, pueden consultarse los detalles técnicos en [15]-[16].

### B. Portal web

El entorno Web fue completamente creado con Atom —un sofisticado editor de código gratuito— y Emmet —una herramienta para desarrolladores Web que mejora eficientemente el flujo de trabajo— a partir de una plantilla HTML5. La página web principal consta de un menú de navegación superior junto con un control deslizante de imagen que da la bienvenida al laboratorio remoto de redes. En cuanto al menú, utilizamos ventanas modales y enlaces a ‘Inicio’, ‘Cisco Networking Academy’, ‘FAQ’ desarrollado con la extensión FancyBox, ‘Quiénes Somos’ con plano de localización proporcionado por Google Maps y un formulario de contacto validado con HTML5 y reCAPTCHA de Google. En cuanto al control deslizante, usamos Unsplash, un servicio Web que ofrece imágenes de alta resolución bajo licencia Creative Commons Zero (CC0) en intervalos de tiempo muy cortos a través de una API. Debajo se muestran dinámicamente las maquetas habilitadas para la enseñanza y las disponibles en el momento de la visita (Fig. 2). El portal Web fue diseñado para soportar tanto teléfonos inteligentes (ancho < 768 píxeles), tabletas (ancho > 768 píxeles), terminales de escritorio de tamaño medio (ancho < 992 píxeles) como de tamaño grande (ancho > 1200 píxeles). El contenido en las vistas móviles aparece contraído gracias a la extensión Collapse de JavaScript integrado en Bootstrap, mostrando así el sitio Web y los glifos (i.e., fuentes de iconos) verticalmente. A medida que aumenta el ancho de la ventana

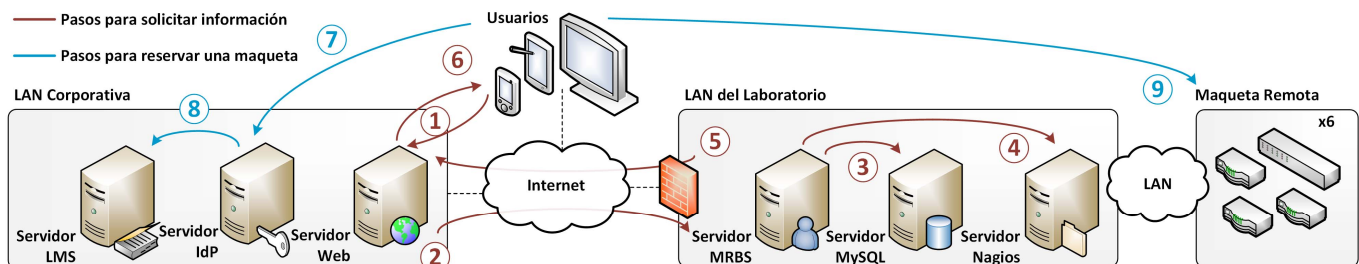


Figura 1. Metodología de acceso remoto al laboratorio de redes.

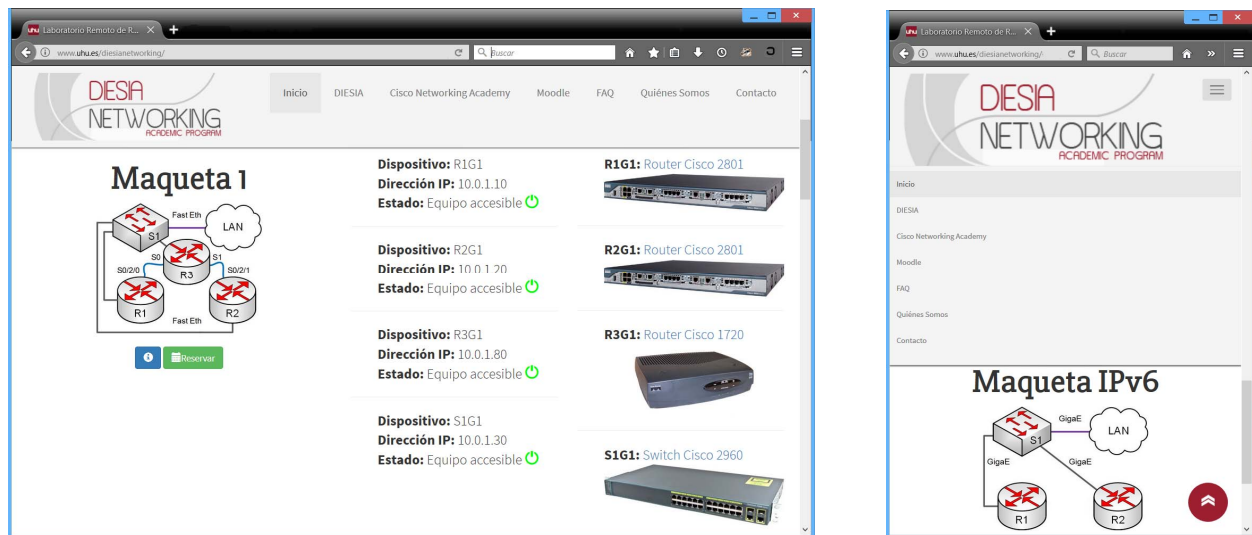


Figura 2. Portal web de Cicolab: detalle del estado de la maqueta (izquierda) y de la vista móvil (derecha).

de visualización, el menú se expande hacia un diseño horizontal que pierde la contracción.

Por otro lado, el portal Web se indexó en Google, Bing y Yahoo! mediante técnicas SEO como estrategia publicitaria para dar mayor visibilidad a la academia local DIESIA de la UHU. Dicha academia de enseñanza está involucrada de forma gratuita en el programa Cisco Networking Academy desde 2007. La implementación SEO requirió almacenar en el servidor Web un archivo 'robots.txt' y un archivo XML del mapa del sitio con directivas y direcciones URL para cada uno de los motores de búsqueda web de Internet. Finalmente, el registro del sitio web se completó a través de herramientas de búsqueda disponibles para desarrolladores Web denominadas webmasters.

### C. Monitorización de los recursos hardware

El laboratorio de enseñanza consta de un puesto de trabajo de profesor y 6 maquetas para estudiantes. Cada maqueta consta de hasta dos routers Cisco 2801, un router Cisco 1720, un switch Cisco Catalyst 2960 y dos PCs equipados con dos tarjetas Gigabit Ethernet conectadas a la LAN y al hardware de las maquetas. De esta manera, una tarjeta de red se utiliza para acceder de forma remota al laboratorio mientras la segunda se utiliza para administrar el hardware durante las sesiones prácticas del aula.

Para mostrar en tiempo real el estado de los dispositivos de red y sus interfaces usamos Nagios® Core™ 4.0.8, la extensión Nagios 2.0.3, el complemento MK Livestatus 1.2.4 y el script 'check\_cisco.pl'. El complemento MK Livestatus de M. Kettner hace uso de la API Nagios Event Broker para abrir un socket mediante el cual los hosts, servicios u otras fuentes de datos son recuperados a demanda (Fig. 2). Los requisitos mínimos de los dispositivos del laboratorio para ser monitorizados y administrados desde el servidor MRBS fueron: a) estar en la misma LAN que el servidor Nagios®, b) tener habilitado SNMP en la configuración, y c) tener activado SSH. Por otra parte, se configuró un servicio DHCP en el laboratorio docente para asignar direcciones IP privadas estáticas a los dispositivos de red basándose en sus direcciones MAC. A diferencia de los routers, los switches necesitaron

tener configurada una VLAN en sus puertos. Además, en los dispositivos de red se requirió la generación de claves públicas y privadas para proporcionar acceso seguro mediante SSH habilitando la función de clave criptográfica RSA. Como ejemplo de la flexibilidad del sistema, hemos incluido también en el laboratorio otros dispositivos Cisco como los routers 1841 y 3741 con software IOS a partir de la versión 12.3 advsecurityk9-mz, SNMP v1.0 y RSA entre 360-2048 bits con SSH v1.5.

### D. Sistema de reservas del laboratorio

Para dar soporte al mayor número de navegadores posibles, incluyendo las versiones 6 a 8 de Internet Explorer, usamos MRBS 1.5.0 con jQuery 1.12.4. Para mejorar el entorno visual y adaptarlo a las resoluciones de diversas pantallas —manteniendo a la vez compatibilidad— optamos por usar Bootstrap con jQuery 1.8.2, las extensiones DataTables y ColReorder 1.3.2 así como la utilidad ThemRoller para la gestión de los estilos web. Para establecer la comunicación entre el portal Web y el sistema de reservas usamos php-curl, una librería PHP que usa Curl para hacer peticiones HTTP de origen cruzado con CORS (i.e., compartir recursos entre diferentes dominios). Además, usamos consultas AJAX dentro de código JavaScript para invocar diversos servicios Web ejecutados en el servidor MRBS (Fig. 3a y 3b). Entre ellos, conocer la disponibilidad de las maquetas a través de una base de datos MySQL actualizada por Nagios®. Dicha información se codifica posteriormente en una matriz JSON que es procesada por la llamada AJAX para mostrarla en el portal Web.

Por otro lado, AnyTerm 1.1.29 fue la aplicación de terminal interactivo usada para controlar remotamente los dispositivos del laboratorio docente (Fig. 3c). El objetivo fue emular el entorno de trabajo que los estudiantes habitualmente disponen cuando están en el laboratorio frente a una consola real (e.g., HyperTerminal). Además de realizar modificaciones en los ficheros CSS, JavaScript y PHP de MRBS para incrustar AnyTerm, su código fuente fue modificado en varios de sus aspectos originales mediante la librería Boost C++ 1.41.0 para

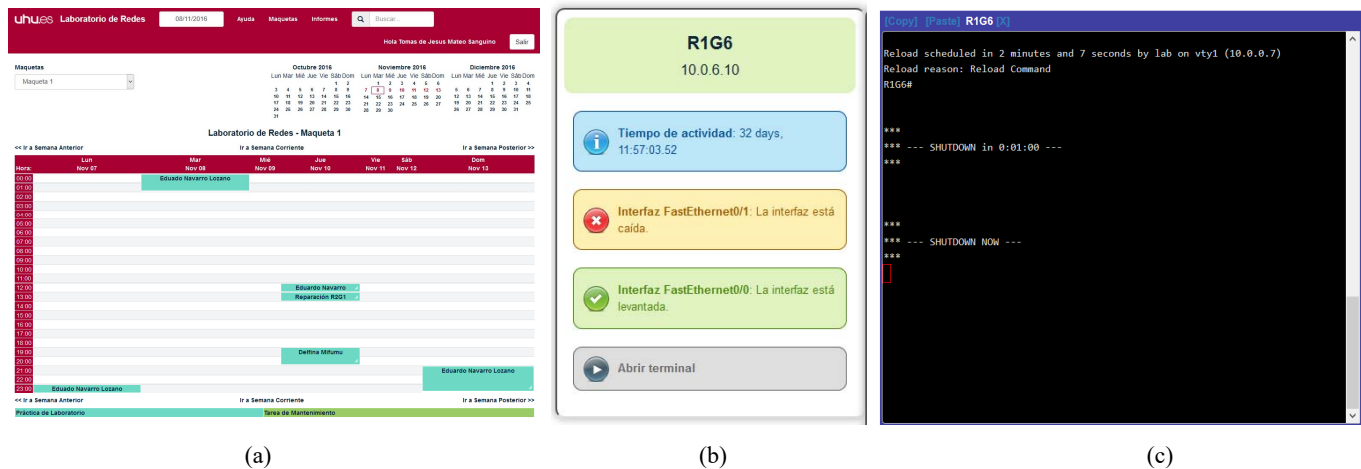


Figura 3. MRBS (a), estado de las maquetas del laboratorio (b) y terminal remoto de AnyTerm (c).

proporcionar funciones adicionales de seguridad, capacidad de controlar los parámetros de las reservas y facilitar la parametrización de la aplicación. En primer lugar, añadimos SSHPass para establecer conexiones seguras con los dispositivos de red sin tener que especificar la contraseña de usuario para el modo de ejecución privilegiado de comandos. Esto es especialmente útil para dispositivos Cisco antiguos que disponen de versiones de IOS no compatibles con SSH v2. En segundo lugar, añadimos MySQL Connector para comunicar AnyTerm con la BBDD y verificar las reservas de usuario gestionadas por MRBS (i.e., ID de reserva, clave de un solo uso, ID de maqueta y nombre, identificación del dispositivo y dirección IP). Gracias a esta característica, al abrir un terminal se comprueban los datos de la reserva que, junto a un proceso killer, se encarga de finalizar la sesión cuando expira el periodo de reserva. En tercer lugar, creamos un archivo de configuración para incluir los parámetros predeterminados y evitar la compilación del código fuente de AnyTerm con cada cambio futuro.

Una de las modificaciones más útiles realizadas en AnyTerm sirvió para registrar los comandos ejecutados por los usuarios a través de la interfaz de línea de comandos (CLI) de los dispositivos (Fig. 4). Así, cuando un usuario ejecuta una sentencia, esta se almacena automáticamente en un archivo de valores separados por comas (CSV). Este archivo también almacena el correo electrónico del usuario, UID de reserva, hora de inicio, nombre de router o switch, fecha y hora en la que fue ejecutado cada comando. Ello permite realizar un seguimiento exhaustivo del trabajo del alumno durante su reserva. Aunque esta información histórica no es visible para los estudiantes —pues queda restringida aunque los datos pertenezcan a su propia reserva— puede ser directamente visualizada por los profesores al acceder a la reserva del estudiante. Esta capacidad nos ofrece un control adicional sobre las tareas que lleva a cabo el alumno.

Por otra parte, utilizamos el componente Canvas de HTML5 para diseñar páginas web fáciles de usar y mejorar la experiencia de usuario. HTML5 representa un lenguaje multiplataforma estandarizado en 2014 por el World Wide Web Consortium (W3C) que reemplaza la obsoleta tecnología Flash para navegadores. En nuestro caso sirvió para representar las maquetas del laboratorio y crear pequeñas

Comando	Equipo	Fecha y hora
show ip int brief	R1G1est	2014-10-06 00:34:04
conf term	R1G1est	2014-10-06 00:37:56
exit	R1G1est	2014-10-06 00:37:57
show ip int brief	R1G1est	2014-10-06 00:47:20

Figura 4. Ejemplo del historial de comandos de usuario durante una sesión práctica con un router Cisco 1720.

aplicaciones Web dentro de MRBS. Mediante Canvas los profesores pueden simular dispositivos virtuales asociados a los puestos de trabajo y añadir áreas interactivas en ellos a través de un panel de herramientas (i.e., dibujar, editar, guardar o borrar texto, puntos, líneas o polígonos). De esta manera, cada vez que un usuario accede a su reserva puede hacer clic sobre una de las zonas interactivas de la imagen y abrir intuitivamente el terminal de consola remoto del dispositivo de red (Fig. 5). Para ello fue necesario combinar Canvas con AJAX, JavaScript, PHP y JSON.

#### E. Sistema de mantenimiento automático

Por lo general, los técnicos de laboratorio realizan manualmente el mantenimiento de los equipos Cisco del laboratorio de docencia una vez por semana. Con el objetivo de mejorar la gestión automática del laboratorio de redes, creamos una tarea de mantenimiento automatizada utilizando un tipo especial de reserva que reinicia los sistemas en una fecha, intervalo o tiempo dado. Para ello usamos Cron como planificador de tareas y shell scripts personalizados. Así, cada vez que el personal técnico realiza una reserva especial de mantenimiento para una maqueta del laboratorio se incluye una entrada en el archivo crontab del servidor MRBS. Cuando llega el momento, el gestor de procesos de la máquina solicita al servidor Nagios® el estado de los dispositivos de la maqueta utilizando la extensión MK Livestatus. Si la maqueta en cuestión está presente, se accede al IOS de cada dispositivo Cisco —ya sea router o switch— a través de SSHPass. Después se realiza en la memoria Flash una copia de respaldo de la configuración de inicio por defecto del equipo y se envían varios mensajes de aviso a través de la consola de terminal antes de reiniciar el equipo (Fig. 3). Posteriormente,





Figura 5. Componente Canvas (derecha) usado para definir zonas de interacción sobre las maquetas (izquierda).

se eliminan las entradas antiguas del archivo crontab del servidor mediante una sencilla aplicación propia desarrollada en C++ denominada CronMonitor.

Por otro lado, los servidores Ubuntu en los que desplegamos los servicios de MRBS, MySQL y Nagios® del laboratorio de redes fueron equipados con scripts capaces de intercambiar periódicamente copias de seguridad en la nube. Entre varias alternativas como Dropbox o Google Drive, elegimos el servicio de Box debido a su mejor seguridad, política de privacidad y espacio disponible por cuenta. Box ofrece hasta 10 GB de almacenamiento gratuito con 250 MB de tamaño máximo por archivo. Además cifra el contenido en sus centros de datos —ofreciendo así las medidas de seguridad que necesitamos— y es menos intrusivo para los usuarios que otros servicios que proporcionan un uso gratuito de sus servidores. Con este objetivo desarrollamos dos shell scripts, uno para realizar copias de seguridad en la nube y otro para restaurarlas en el laboratorio. Mientras el primero es capaz de realizar copias completas, diferenciales o incrementales de archivos comprimidos y carpetas de los servidores del laboratorio docente, el segundo permite restaurarlas a través de un menú de usuario desarrollado. Dicha tarea se realiza de forma eficiente, para lo cual se eliminan cíclicamente los datos más antiguos con idea de optimizar el espacio disponible en la nube. Los scripts fueron programados para utilizar el protocolo WebDAV de Box a través de las funciones Davfs2 y Curl de RPMforge. A modo de ejemplo, un test consistente en subir siete archivos de 50 MB y un archivo de 41,4 MB para un total de 391,4 MB requirió ~49 minutos a través de un enlace con ancho de banda ascendente de 1,2 Mbps.

## II. CASO DE ESTUDIO

Un ejemplo del tipo de sesiones prácticas remotas que llevaron a cabo los estudiantes en el laboratorio remoto fue la configuración de protocolos de enrutamiento dinámico. La topología de red utilizada para llevar a cabo esta práctica consistió en dos routers Cisco 2801 y un switch Cisco Catalyst 2960 con ayuda de un router Cisco 2621XM situado en uno de los armarios de comunicación del laboratorio docente. La conexión entre los dos routers se realizó a través de un cable Serial y las interfaces WIC. La conexión desde los dos routers al switch se realizó mediante un cable UTP y las interfaces Fast Ethernet. El switch se conectó al router del armario de comunicaciones a través de la LAN del laboratorio, actuando este como puerta de enlace.

El objetivo educativo de la práctica fue probar la conectividad entre dispositivos con la herramienta ping, determinar qué subredes conocía cada router, reflexionar sobre

qué protocolos de enrutamiento se habilitaron en cada dispositivo, revisar las métricas y distancias administrativas de las rutas en las tablas de encaminamiento, cuáles de ellas prevalecían en dichas tablas y por qué. Para llevarlo a cabo, la práctica fue dividida en cuatro tareas: 1) configurar los parámetros básicos de los dispositivos: examinar y guardar la configuración del router, configurar las direcciones de router en las interfaces FE0/1 y S0/0 según la tabla de direccionamiento dada, y chequear la tabla de enrutamiento de cada router; 2) configurar el protocolo RIPv2: activarlo en ambos routers con el comando 'router rip' e inspeccionar las tablas de enrutamiento; 3) configurar el protocolo EIGRP: activarlo en ambos routers con el comando 'router eigrp 1', inspeccionar las tablas de encaminamiento, comprobar los routers vecinos con el comando 'show ip eigrp neighbors' y comprobar las tablas de topología con el comando 'show ip eigrp topology'; 4) configurar el protocolo OSPF: activarlo en ambos routers con el comando 'router ospf 1', inspeccionar las tablas de encaminamiento, comprobar los routers vecinos con el comando 'show ip ospf neighbor' y comprobar las tablas de topología con el comando 'show ip ospf database'.

## III. EXPERIMENTACIÓN

Con objeto de medir las bondades del laboratorio remoto se han llevado a cabo varias pruebas consistentes en analizar el esfuerzo necesario para desplegarlo, el impacto de las distintas herramientas utilizadas sobre la red de datos, la asistencia de los estudiantes y su desarrollo práctico. La experiencia fue llevada a cabo por 24 estudiantes de diferentes niveles del Grado de Ingeniería Informática de la UHU entre el 02 de noviembre de 2016 y el 18 de enero de 2017, con un total de 36 sesiones prácticas.

### A. Análisis sobre el despliegue y gestión de la infraestructura del laboratorio

Con el objetivo de cuantificar la dificultad de emplear las diversas herramientas utilizadas por los autores en el proyecto, la Tabla II muestra una relación de las tareas efectuadas. La primera columna indica el identificador de la tarea, la siguiente columna resume el objetivo, la tercera columna indica si la tarea puede ser llevada a cabo de forma genérica o si es específica del proyecto (i.e., ad hoc) y la última columna representa la carga de trabajo en horas. Como se describe, el conjunto de tareas requirió un total de 183,5 horas de desarrollo (74,11% genéricas y 25,88% específicas) donde aquellas que requirieron más esfuerzo fueron orientadas a modificar el terminal interactivo de AnyTerm para añadir nuevas funcionalidades (48 horas) y el desarrollo del editor de

TABLA II  
RESUMEN DE TAREAS GENÉRICAS Y ESPECÍFICAS PARA IMPLEMENTAR EL LABORATORIO CISCOLAB

Tarea	Descripción	Propósito	Duración
1	Desplegar instancia de SimpleSAMLphp en modo proveedor de servicios	Específico	1,5 horas
2	Desarrollar módulo authZ para SimpleSAMLphp	Genérico	8 horas
3	Desarrollar módulo SimpleSAMLphp para MRBS	Genérico	8 horas
4	Desarrollar tema Bootstrap para MRBS	Genérico	16 horas
5	Modificar código MRBS para soportar AnyTerm y recibir estado de router/switch	Específico	10 horas
6	Desplegar Nagios® e instalar módulos de soporte para Cisco	Específico	3 horas
7	Añadir nuevas funciones a AnyTerm (i.e., soporte MySQL, archivo de configuración, login no interactivo, acceso planificado, registro de comandos)	Genérico	48 horas
8	Desplegar AnyTerm	Específico	1 hora
9	Desarrollar editor Canvas para MRBS	Genérico	40 horas
10	Programar scripts para realizar copias de seguridad (copia completa, incremental y diferencial)	Genérico	16 horas
11	Desarrollar servicio Web para mostrar estado de router/switch	Específico	2 horas
12	Desarrollar portal Web de Ciscolab	Específico	30 horas

Canvas para MRBS (40 horas). Como se aprecia, dichas tareas están en consonancia con el desarrollo típico de páginas web como la de Ciscolab (30 horas).

#### B. Impacto de las herramientas en el rendimiento de la red

Para estudiar el impacto de los diferentes servicios utilizados en la solución de los autores, se realizaron dos pruebas sobre el rendimiento de la red. El objetivo fue medir cuantitativamente el acceso Web de los usuarios a los recursos del laboratorio (i.e., cómo perciben el tiempo de acceso) y el tráfico del sistema de mantenimiento automático sobre la red del laboratorio (i.e., cómo el laboratorio docente soporta la carga de tráfico). La Figura 6 muestra los diferentes ensayos agrupados por colores, donde el verde representa las conexiones entre usuarios y servidores en la red corporativa, el naranja representa las conexiones entre usuarios y servidores en el laboratorio docente, y el color azul representa las conexiones originadas por los servicios de gestión automática dentro y desde el laboratorio. Los tamaños de datos y tiempos promedios fueron obtenidos considerando una serie de 10 registros por evento.

En cuanto a los estudiantes, supervisamos sus sesiones Web utilizando la herramienta para desarrolladores Web de Mozilla Firefox siguiendo la metodología de acceso descrita en la Figura 1. Para ello capturamos el tamaño de los mensajes correspondientes a contenido web no almacenado en caché. También descartamos los tiempos de espera de los usuarios para hacer clic en los enlaces. Los pasos dados consistieron en acceder el servidor Web de Ciscolab (1,44 Mbytes en 1,68s), solicitar el estado de las maquetas del laboratorio al servidor Nagios® (2,62 Mbytes en 3,27s), autenticarse contra el servidor IdP institucional (0,51 MB en 1,78s) y acceder al servidor MRBS del laboratorio (0,59 MB en 2,48s). La operación hasta llegar al terminal remoto interactivo requirió 12 clics en total sumando 5,18 MB de datos transferidos en un proceso que duró 9,21s. En concreto, el ~45% de los datos correspondió a archivos de imágenes (e.g., sliders ornamentales y figuras con topologías de las maquetas), el ~25% a archivos de tipo 'js', el ~20% a archivos de tipo 'css' (e.g., hojas de estilo) y el ~10% a ficheros de tipo 'html'. En conclusión, los usuarios percibieron tiempos de acceso y

transferencias de datos compatibles con otras aplicaciones Web similares.

En cuanto al sistema de mantenimiento automático, monitorizamos el tráfico de datos del laboratorio usando Wireshark®, una herramienta de código abierto para el análisis, desarrollo de protocolos y resolución de problemas de red. En primer lugar capturamos el tráfico SNMP, el cual generó un promedio de 135,19 KB cada 10s por defecto. Con respecto a la sincronización de la información de reserva entre las bases de datos de los servidores Ubuntu, transferimos 0,39 MB de datos sin comprimir en 25,21s. En cuanto a las copias de seguridad en la nube, realizamos un duplicado completo para mostrar el caso más desfavorable. Este consistió en realizar una copia de la base de datos MySQL y de la información de las reservas de MRBS obteniendo un total de 36,97 MB en 79,24s. Cabe decir que dicha copia requiere subir una secuencia de archivos comprimidos una sola vez por semana, mientras que las copias diferenciales e incrementales requieren subir mucha menos cantidad de datos una vez y cinco veces por semana, respectivamente. Como conclusión de esta batería de pruebas, se determinó que la mayor carga de tráfico se originó desde el sistema de mantenimiento automático del laboratorio a la WAN, mientras que el tráfico dentro de la LAN tuvo un impacto mínimo sobre la red del laboratorio docente. Por otra parte, encontramos que la comunicación entre los servicios Web —ya estuvieran

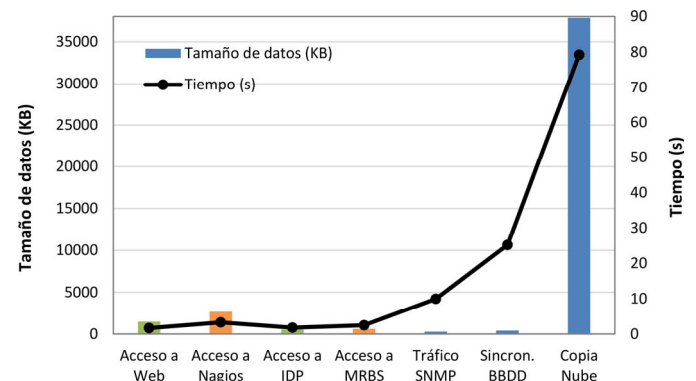


Figura 6. Análisis del rendimiento de la red: tamaño de datos y tiempo promedio de envío.

ubicados en la red corporativa o en el laboratorio docente—requieren mucho menos tiempo en comparación con la comunicación de los sistemas de mantenimiento automático del laboratorio. Esto sugiere un mejor tiempo de acceso a los servicios Web en general y, por tanto, menos impacto sobre el trabajo de los estudiantes.

### C. Análisis sobre la asistencia al laboratorio remoto

En la Figura 7 se muestra un gráfico sobre la asistencia de los estudiantes al laboratorio remoto, donde el tamaño de los círculos es proporcional al tiempo de trabajo que emplearon. La experiencia acumulada por los alumnos resultó en un total de 1663 minutos ( $\sim 27,71$  horas) con una media de  $46,2 \pm 25,7$  minutos por sesión. Teniendo en cuenta que cada práctica presencial del laboratorio tiene una duración de 90 minutos, encontramos que el tiempo de trabajo que los estudiantes dedicaron a sus prácticas remotas se redujo al 51,3%. Esto sugiere que se consiguió una mejor utilización de los recursos de red como resultado de tener y usar equipos de laboratorio remotos. Por otra parte, encontramos que el 19,2% del calendario de trabajo se desplazó al fin de semana, preferiblemente sábado. Este desplazamiento supuso un total de 533 minutos de uso (i.e., 32,05%). Además, encontramos que se empleó un total de 294 minutos (i.e., 17,67%) en realizar las prácticas fuera de la jornada de trabajo habitual (i.e., después de las 21:00 horas). Todo lo anterior sugiere que los estudiantes cambiaron sus hábitos de trabajo fuera de la jornada laboral y el horario estándar, aportando el laboratorio remoto mayor flexibilidad en comparación con el laboratorio tradicional. Como principal inconveniente, observamos que los usuarios reservaron el laboratorio un total de 4380 minutos ( $\sim 73$  horas) con una media de  $121,7 \pm 36,9$  minutos por sesión. Esto representa un ratio de aprovechamiento del 37,96% respecto al uso real que los estudiantes hicieron de los recursos. Dicho resultado sugiere un uso deficiente del sistema de reservas, lo que podría haber bloqueado a otros compañeros potenciales para llevar a cabo sus prácticas. Como solución, hemos equipado el laboratorio remoto con varias maquetas de trabajo —suficientes en principio para el quorum de clase— pudiendo los profesores modificar o anular las reservas de los

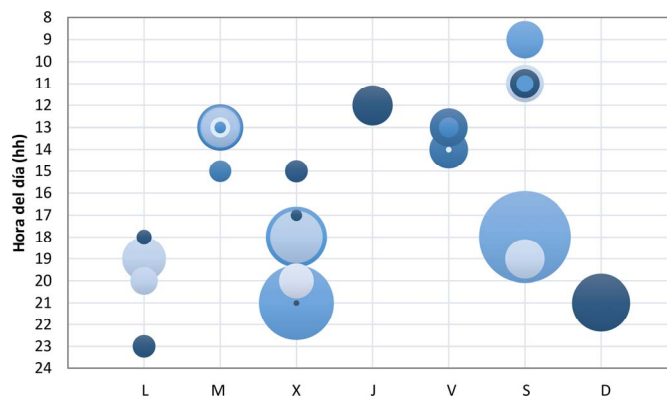


Figura 7. Análisis del horario utilizado por los estudiantes durante las prácticas remotas.

estudiantes en caso de abuso de los recursos.

### D. Análisis del trabajo de los estudiantes

Los routers y switches de Cisco se basan en su mayoría en el firmware IOS, un sistema operativo modal con estructura jerárquica que proporciona un conjunto fijo de comandos que admite varias entradas posibles para la misma sentencia. Esto significa que el conjunto de comandos disponible está determinado por el modo y nivel de privilegios del usuario, lo cual es fácil de inspeccionar. Con el fin de analizar las fortalezas y debilidades de los estudiantes, se recogieron los archivos \*.csv de sus sesiones prácticas y se extrajeron las sentencias empleadas. Posteriormente analizamos con NodeXL un total de  $\sim 2330$  entradas en la línea de comandos obteniendo una media de  $64,75 \pm 33,92$  registros por sesión. La Figura 8 muestra un gráfico de Harel-Koren que relaciona los comandos erróneos (texto rojo), comandos en modo incorrecto (flecha naranja) y comandos de ayuda (texto azul) con el modo de funcionamiento correcto del router (texto verde, texto negro y flecha gris). Del análisis del trabajo de los estudiantes se encontró un total de 72 entradas erróneas (3,09%) debido principalmente a errores tipográficos y, en menor medida, a comandos no existentes. Esto sugiere que usar un CLI en una ventana de terminal puede llegar a dificultar en ocasiones la práctica de laboratorio (e.g., debido a

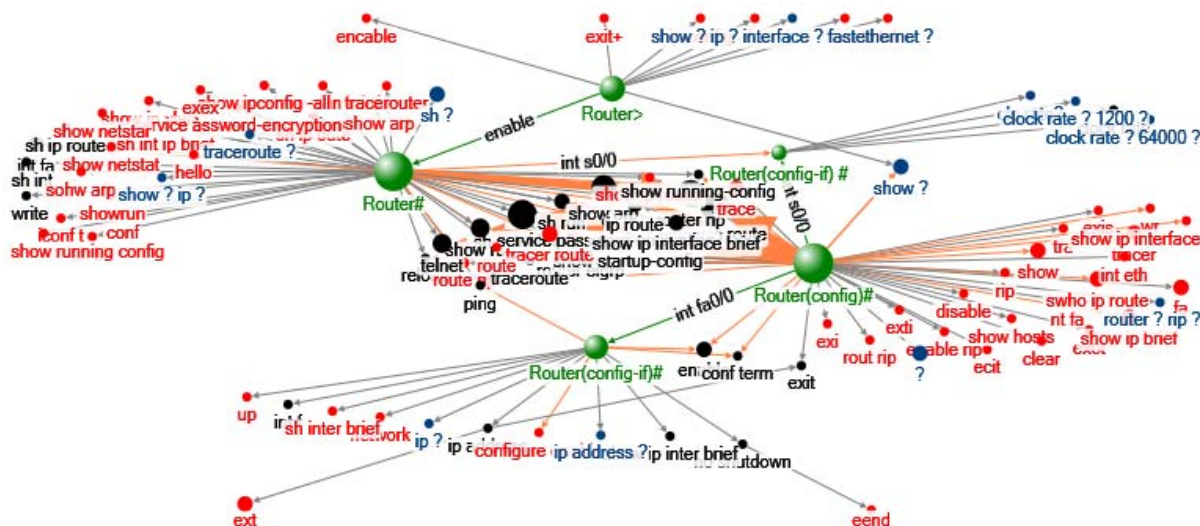


Figura 8. Análisis del trabajo de los estudiantes sobre los dispositivos de red de Cisco.



la escritura rápida de comandos o a la confusión con sentencias comúnmente utilizadas en otros sistemas operativos como Unix). Además, encontramos un total de 77 comandos ejecutados erróneamente en un nivel de privilegio diferente al esperado (3,30%). Esto significa que los estudiantes confunden en ocasiones la estructura jerárquica en árbol del firmware IOS. A pesar de los errores, solo encontramos un total de 17 entradas relacionadas con el comando de ayuda o su signo equivalente '?'. Esto sugiere que los estudiantes generalmente prefieren confiar en sus conocimientos en vez de usar la ayuda del sistema IOS, reduciendo levemente así su rendimiento general.

#### IV. CONCLUSIONES Y TRABAJOS FUTUROS

Generalmente, el aprendizaje práctico en laboratorios tradicionales docentes está supeditado a la disponibilidad y topología de un cierto número de dispositivos hardware —normalmente de alto coste— que, por lo general, está ligado además a las escasas sesiones presenciales disponibles. Estos inconvenientes nos llevaron a desarrollar un proyecto educativo universitario —denominado Ciscolab— en el que implementar un laboratorio de redes de computadores. El objetivo fue proporcionar acceso remoto a los recursos, mejorar la experiencia del usuario y detectar el desempeño del estudiante. Con esta idea se implementó una solución basada en servidores Web, MRBS, MySQL, IdP y Nagios®, entre otros. El resultado fue una infraestructura de bajo coste —fundamentada en software libre— que mejora la seguridad del acceso, gestión de cuentas, monitorización de recursos, procedimientos de respaldo, tareas de mantenimiento, interactividad y diseño.

Las pruebas sirvieron para demostrar que el esfuerzo que supone implementar la solución propuesta por los autores está en consonancia con el desarrollo de proyectos Web similares. En cuanto al impacto sobre el rendimiento de la red, los experimentos realizados sugieren que el tráfico y tiempo de acceso de los usuarios está en la línea de un entorno Web. Por otro lado, la experiencia llevada a cabo por un total de 24 estudiantes del Grado de Ingeniería Informática de la UHU durante un cuatrimestre mostró un mejor uso y flexibilidad de los recursos remotos en comparación con las prácticas presenciales de laboratorio. El análisis de un total de 36 sesiones prácticas nos permitió conocer el trabajo de los estudiantes, detectar sus hábitos y averiguar sus debilidades al trabajar con dispositivos de red. Ello supone un punto de partida para resolver sus dificultades e incidir en la mejora futura de su rendimiento.

Dentro de los trabajos futuros, desde el punto de vista técnico estamos focalizando nuestros esfuerzos en mejorar la disponibilidad, escalabilidad y facilidad de mantenimiento del laboratorio. Si bien los sistemas están configurados actualmente para proporcionar alta disponibilidad y escalabilidad, queremos dar soporte 24x7x365 y reducir la complejidad. Por otro lado, algunas de las herramientas que se han usado para lograrlo son específicas por lo que, en ocasiones, el mantenimiento del laboratorio puede llegar a ser costoso. Para solucionarlo, las tareas se están centrando en mejorar tres aspectos: *i)* creación de un clúster MySQL activo/activo mediante Galera y MariaDB MaxScale para albergar todas las bases de datos del laboratorio. Este clúster

lo monitorizaríamos con un software específico denominado ClusterControl que, entre otras funcionalidades, aporta un rearmado automático de los nodos; *ii)* creación de un clúster de servidores Web con balanceadores de carga, sistema de ficheros compartidos y replicados mediante GlusterFS para hospedar la aplicación de reservas; *iii)* simplificar la monitorización de maquetas del laboratorio mediante PandoraFMS, un software basado en Web más sencillo de instalar y configurar que Nagios® incluso en alta disponibilidad. Aunque este último es considerado un estándar de facto en la monitorización Open Source, como contrapartida necesita la instalación de varias extensiones para alcanzar una configuración eficiente y funcional. Esta se realiza además, en la mayoría de los casos, mediante ficheros de texto. Aunque PandoraFMS no tiene la misma variedad de extensiones que Nagios®, contiene las necesarias para monitorizar los distintos elementos que conforman nuestro laboratorio.

Desde el punto de vista docente, los trabajos siendo llevados a cabo están orientados a evaluar y mejorar el rendimiento académico de los alumnos mediante las siguientes acciones: *i)* administrar un cuestionario de opinión y recoger la valoración de los alumnos sobre distintos aspectos del laboratorio remoto tales como el aprendizaje, motivación, usabilidad y viabilidad para su puesta en práctica en el contexto universitario; *ii)* realizar un análisis factorial de correspondencias (AFC) para estudiar la relación entre los aspectos anteriores con el objetivo de determinar las ventajas y desventajas del laboratorio remoto en base a la opinión de los usuarios; *iii)* automatizar la extracción de datos sobre el trabajo del alumno en el laboratorio empleando técnicas de inteligencia en el negocio para analizar y mejorar el rendimiento académico teórico-práctico (e.g., mediante MicroStrategy).

#### AGRADECIMIENTOS

Expresamos nuestro agradecimiento al Dpto. de Ingeniería Electrónica, Sistemas Informáticos y Automática de la UHU. Especialmente a R. González Gómez, M.R. González Pino, E. Cortés Ancos y D.A. López García por su valioso apoyo y colaboración.

#### REFERENCIAS

- [1] J.M. Andújar Márquez, and T.J. Mateo Sanguino, "Design of Virtual and/or Remote Laboratories. A Practical Case," *Revista Iberoamericana de Automática e Informática Industrial*, 7(1), pp. 64-72, 2010.
- [2] D. Britos, "Laboratorio Virtual y Remoto para la Enseñanza de Diseño y Administración de Redes de Computadoras," *XV Workshop de Investigadores en Ciencias de la Computación*. Paraná, Argentina, pp. 1121-1125, 2013.
- [3] L.M. Jiménez, et al., "Laboratorios Remotos para las Prácticas de Ingeniería de Sistemas y Automática en la Universidad Miguel Hernández," *Proc. Red Temática de Docencia en Control Mediante Web*. Alicante, Spain, pp. 7-13, 2005.
- [4] J. Sandoval, et al., "Desarrollo de laboratorios virtuales-remotos en México," *Ciencia, Tecnología e Innovación para el Desarrollo de México*, 2(46), 2010.
- [5] P. Grygárek, et al., "The Semi-automatic Evaluation of Students' Configurations in VIRTLAB Distributed Virtual Networking Laboratory," *Proc. 6th International Conference on Emerging eLearning Technologies and Applications*, Stara Lesna, Slovakia, 2009.
- [6] S. Grober, et al., "Experimenting from a Distance Remotely Controlled Laboratory (RCL)," *European Journal of Physics*, 28(3), pp. 1-27, 2007.

- [7] P. Orduña, et al., "LabsLand: A sharing economy platform to promote educational remote laboratories maintainability, sustainability and adoption," *Proc. 46<sup>th</sup> Annual Frontiers in Education Conf.*, 2016.
- [8] I. González, et al., "Novel Networked Remote Laboratory Architecture for Open Connectivity Based on PLC-OPC-LabVIEW-EJS Integration. Application in Remote Fuzzy Control and Sensors Data Acquisition," *Sensors*, 16(11), pp. 1-24, 2016.
- [9] M. Lloret, et al., "A Remote Network Laboratory to Improve University Classes," *Proc. 5th WSEAS / IASME Intern. Conf. Engineering Education*, Heraklion, Greece, 2008.
- [10] J.J. Fuertes, et al., "A Virtual Laboratory of D.C. Motors for Learning Control Theory," *Intern. J. Electrical Engineering Education*, 50, pp. 172-187, 2013.
- [11] A.P. Chávez, et al., "Mejora al Proceso de Enseñanza-Aprendizaje Mediante el Acceso Remoto a Laboratorios de Redes. Soluciones de TI para Optimizar Recursos en la Academia," *Proc. Red de Directores de Tecnologías de Información y Comunicación de las Universidades Latinoamericanas*, City of Panama, Panamá, 2011.
- [12] T.J. Mateo Sanguino, et al., "OpenGnSys: a Novel System toward Centralized Deployment and Management of Computer Laboratories," *Computers & Education*, 75, pp. 30-43, 2014.
- [13] D. Pardo Garrido, et al., "OpenGnSys: un Sistema de Gestión Centralizada y Despliegue de Sistemas Operativos en el Aula," *Proc. 8<sup>th</sup> Iberian Conf. Information Systems and Technologies*, Lisbon, Portugal, 1, pp. 295-301, 2013.
- [14] T.J. Mateo Sanguino, et al., "The Role of Telematic Practices in Computer Engineering: a Low-cost Remote Power Control in a Network Lab," *Intern. J. Online Engineering*, 8(2), pp. 15-22, 2012.
- [15] A. García Domínguez, "Prácticas Telemáticas Mediante Sistema de Reservas y Proveedor de Identidad," *Bs. Thesis*, University of Huelva, 2014.
- [16] J. Espejo Fernández, "Mejora de la Gestión y Acceso Remoto del Laboratorio de Redes," *Bs. Thesis*, University of Huelva, 2016.

Sus intereses de investigación actuales incluyen la educación en ingeniería.



**Jennifer Espejo Fernández** recibió el Grado en Ingeniería Informática por la Universidad de Huelva, España en 2016. Actualmente trabaja como consultora en diseño e integración de sistemas en Sopra Steria, Valencia, España.



**Antonio García Domínguez** recibió el Grado en Ingeniería Informática por la Universidad de Huelva, España en 2014. Actualmente es coordinador de redes sociales en Atresmedia y profesor en el Máster de Periodismo Digital de la Universidad Antonio de Nebrija, Madrid, España.



**Tomás de J. Mateo Sanguino** recibió en 2010 el doctorado en Ingeniería Electrónica por la Universidad de Huelva (UHU), España. Ha sido Ingeniero Electrónico y Máster en Enseñanza Universitaria en 2001 y 2008, respectivamente. Desde 1998 a 2004 obtuvo una beca en el Instituto Nacional

de Tecnología Aeroespacial (INTA) y trabajó como ingeniero contratado en el Consejo Superior de Investigaciones Científicas (CSIC). Desde 2004 ha trabajado como profesor colaborador y contratado doctor en el Dpto. de Ingeniería Electrónica, Sistemas Informáticos y Automática de la UHU. Actualmente trabaja también como instructor en el programa CCNA de Cisco Networking Academy. Ha participado como investigador en 20 proyectos, siendo autor y coautor de más de 75 publicaciones en diferentes revistas y congresos. Sus intereses de investigación actuales incluyen la educación de ingeniería.



**Iñaki Fernández de Viana y González** recibió en el año 2015 el título de doctor en Ciencias de la computación por la Universidad de Huelva (UHU), España. Desde 2003 ha sido profesor en el Dpto. de Tecnologías de la Información de la UHU y director del Servicio de Enseñanza Virtual de la UHU. Su investigación principal se ha centrado en modelos de

integración de información empresarial, meta-heurísticos y bio-inspirados. Cuenta con más de 20 publicaciones en revistas, conferencias, talleres y capítulos en libros editados.