# Securizando la comunicación de un repositorio digital de contenido académico con Moodle

Francisco Javier Díaz[1], Alejandra Schiavoni[1], Ana Paola Amadeo[1], Duilio Ray[1]

[1] Laboratorio de Investigación en Nuevas Tecnologías Informáticas - LINTI
Facultad de Informática – Universidad Nacional de La Plata
La Plata, Buenos Aires, Argentina
jdiaz@unlp.edu.ar, {ales,pamadeo}@info.unlp.edu.ar, dray@linti.unlp.edu.ar

**Resumen.** En la Facultad de Informática de la Universidad Nacional de La Plata se vienen usando un conjunto de plataformas que permiten sistematizar los procesos inherentes a la gestión académica. En el LINTI, Laboratorio de Investigación en Nuevas Tecnologías Informáticas, se está desarrollando un proyecto que consiste en la integración de un repositorio con diferentes herramientas y plataformas, como el LMS Moodle, un sistema de gestión de bibliotecas, sistemas de almacenamiento en la nube y redes sociales. La flexibilidad de uso de plataformas basadas en software libre permite la personalización, adaptación e incorporación de nuevos módulos en forma totalmente abierta. Sin embargo, al realizar una modificación es imprescindible tener en cuenta los aspectos de seguridad que pueden poner en riesgo la integridad de la plataforma. El presente artículo describe el análisis realizado a un módulo implementado para la comunicación entre Moodle y DSpace, con el objetivo de detectar posibles vulnerabilidades. El trabajo se realizó con el asesoramiento de expertos en seguridad informática del CERT-UNLP y se tuvieron en cuenta los aspectos de seguridad propios del LMS y del repositorio. Luego de analizar la implementación, se corrigieron los errores encontrados siguiendo las normas prescriptas.

## 1 Introducción

Las principales universidades de varios países han incorporado a su oferta educativa programas apoyados por e-learning para la formación profesional, haciendo uso de las TIC, que permiten flexibilidad en tiempo y espacio, para integrar a más gente en los procesos de enseñanza-aprendizaje. Estos programas consumen y generan contenidos en formato digital que pueden ser aprovechados por otros programas, sistemas u organizaciones con objetivos comunes. La incorporación de nuevas plataformas sirve para complementar la clase presencial, ampliando los límites y espacios temporales, promoviendo el aprendizaje ubicuo, que representa un nuevo paradigma educativo [1].
Con este fin, el sector educativo apuesta a la difusión y reutilización de objetos abiertos como elemento clave para la interoperabilidad y la concentración de recursos de forma estándar, compartida y organizada. Pensar en recursos educativos y en las licencias CC permite introducir el concepto de Recursos Educativos Abiertos (REA),

término que fue utilizado por primera vez por la UNESCO en el año 2002 y desde entonces, varias instituciones educativas de todo el mundo acuerdan con esta iniciativa. Por esta razón, el hecho de integrar los distintos sistemas de información intervinientes en el campo de la educación a distancia resulta de suma importancia y todo un desafío. Siguiendo esta línea, desde hace varios años se está desarrollando un proyecto en el LINTI que apunta a integrar distintas plataformas y herramientas que se utilizan dentro del ámbito académico, así como también la comunicación con las redes sociales ampliamente usadas hoy en día. El proyecto incluye no sólo la creación de un repositorio para albergar el contenido como recursos abiertos, sino también la implementación de módulos que extienden su funcionalidad básica permitiendo la comunicación con otras herramientas, la interacción con servicios externos y la integración con las redes sociales.

Para de la integración con el LMS Moodle, se implementó un módulo, cuyo objetivo fue facilitar la publicación del material generado por los alumnos a través de las entregas de los trabajos prácticos y trabajos finales. La posibilidad de contar con este método de publicación alienta a docentes a publicar dichos trabajos en uno o varios repositorios externos, ya que no necesitan conocer la interfaz y forma de acceso a ellos [2].

La adopción de plataformas basadas en software libre como DSpace y Moodle permite la personalización, adaptación de la interfaz y de las funciones provistas y la incorporación de nuevos módulos, en forma flexible y totalmente abierta. Sin embargo, al modificar una herramienta es imprescindible tener en cuenta en forma estricta aspectos de seguridad, que garanticen que no aparezcan vulnerabilidades que puedan poner en riesgo la integridad de la información.

En este artículo se describen en primera instancia las características y organización de las plataformas involucradas: LMS Moodle y el repositorio basado en DSpace. A continuación, se menciona el trabajo del CERT-UNLP, (Computer Emergency Response Team) que funciona en el ámbito de la Universidad Nacional de La Plata y está integrado por un grupo de investigadores que trabaja en temas de seguridad desde hace más de 5 años. Luego, se describe el módulo desarrollado, se detallan los errores detectados y la forma en que fueron solucionados para garantizar la seguridad integral de la plataforma.

## 2 Descripción general de las plataformas involucradas

La interconexión de sistemas es algo habitual en los servicios ofrecidos actualmente, sobre todo en aplicaciones basadas en Internet y ha tenido amplia repercusión en el ámbito académico.

Una de las consecuencias más evidentes de la influencia de las TIC en la educación superior está en la forma en la que se generan y en los medios por los que se transmiten los contenidos educativos, que en su mayor parte son en formato digital. Esta evolución en la forma de manipular la información por el público en general y por los alumnos durante el proceso de aprendizaje, motivó el estudio de la vinculación con el repositorio de las distintas herramientas que contienen material educativo.

### 2.1 Repositorio digital

La construcción de un repositorio digital, con material que sirve de complemento a la Biblioteca de la Facultad, incluye el análisis de la información a almacenar junto con los detalles de catalogación de la misma.

El repositorio se está implementando con la plataforma DSpace, que es de código abierto que provee herramientas para la administración de colecciones digitales, y comúnmente es usada para gestionar repositorios institucionales. En un principio, para la implementación del repositorio, los recursos a almacenar comprenderán proyectos realizados por los alumnos, durante el dictado de asignaturas de las diferentes carreras. Estos trabajos inicialmente se encuentran almacenados en la plataforma virtual de aprendizaje, a través de la cual son entregados y evaluados por el docente.

El proyecto comenzó con la integración del repositorio con la plataforma virtual de aprendizaje Moodle, de manera de poder establecer una comunicación bidireccional entre ambas plataformas. En primera instancia se llevó a cabo la comunicación para poder consultar y transferir elementos desde DSpace e incluirlos dentro del contenido de un curso [3]. En una segunda etapa, se implementó un módulo específico que permite publicar en forma semi-automática material generado por los alumnos. En las siguientes secciones se describe en forma general las características funcionales del módulo implementado y se detallan los ajustes que se debieron realizar para garantizar los aspectos de seguridad.

En etapas siguientes, se incorporaron las funciones de comunicación con servicios y herramientas existentes en Internet con el fin de potenciar las prestaciones propias del repositorio. Se realizó la comunicación con los servicios de gestión de archivos en la Nube, DropBox y Google Drive, que permiten guardar archivos en forma directa y con la red social Facebook, para compartir y recomendar un recurso determinado. La integración de esta funcionalidad dentro de DSpace intenta aprovechar estas prácticas ya habituales para que el contenido logre una mayor difusión [4].

### 2.2 Plataforma virtual de aprendizaje

La Facultad viene trabajando con la plataforma virtual Moodle para la gestión de cursos a través de la Web, desde hace aproximadamente 12 años como complemento de las clases presenciales de los cursos de las carreras de grado, para cursos de postgrado y de la Secretaría de Extensión [5]. La plataforma incluye más de 10000 usuarios registrados en alrededor de 170 cursos. La Fig. 1 muestra la evolución del número de cursos ofrecidos año a año y por categoría. Es importante mencionar que los cursos quedan activos de un año al otro, quedando vigentes durante 4 o 5 a ños sobre todo para los cursos relacionados con las materias de grado de la Facultad.

Un curso básico en Moodle está formado por recursos y actividades. A través de los recursos se incorporan contenidos como archivos en distintos formatos, páginas Web, un directorio, objetos de aprendizaje, entre otros. Por su parte las actividades crean una interacción entre alumnos y docentes, como cuestionarios, foros, entrega de tareas, etc. La conexión con las redes sociales también es un módulo desarrollado por

este grupo de trabajo, módulo Twitter [6], que se está utilizando cada vez más para comunicar alumnos y docentes.

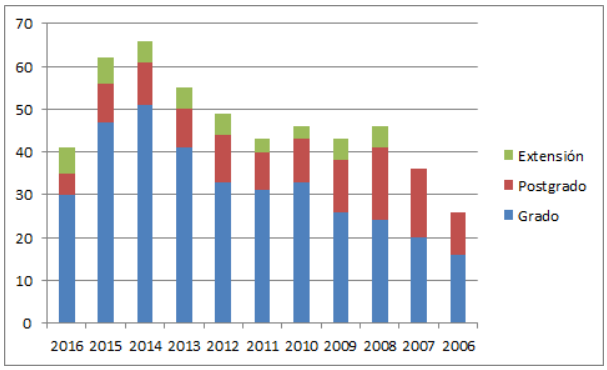


**Fig. 1.** Evolución del número de cursos por año

Dentro de las actividades, las Tareas es uno de los módulos más utilizados por la naturaleza práctica que caracterizan a las materias que integran el plan de estudios de la carrera. A través de las tareas los estudiantes suben sus trabajos que luego son evaluados por los docentes, quienes califican y plasman sus devoluciones a través de Moodle. Las devoluciones y la calificación son enviadas a cada alumno a través de su correo electrónico y sólo pueden ser vistas por él y los docentes a cargo de la materia. Esta forma de entrega resulta muy útil tanto para los alumnos como para los docentes, ya que los trabajos pueden ser subidos en forma on-line desde su hogar y en cualquier momento dentro del plazo establecido.

Como se mencionó anteriormente, algunos trabajos se desarrollan con un objetivo concreto y para ser utilizados en determinadas instituciones que los requieren. Analizando la cantidad de entregas totales realizadas por la plataforma en los últimos años, más de 2700, si fueran descartadas sería un desperdicio significativo, ya que podrían tener un muy alto potencial de reuso y re significación por la cátedra o por docentes de la misma u otras instituciones educativas.

## 3 Normas de seguridad y funcionamiento del CERT-UNLP

En el CeSPI funciona CERT-UNLP, el centro académico cuyo ámbito de aplicación es la Red de la Universidad Nacional de La Plata [7]. El mismo fue creado en el marco de la política de calidad del CeSPI, con el propósito de prevenir, detectar, analizar, investigar, registrar los incidentes de seguridad que son reportados. A partir de la creación de CERT-UNLP [8], los incidentes de seguridad se gestionan de una forma ordenada y sistemática acelerando la rapidez y eficiencia de las respuestas con el fin de minimizar la pérdida de la información y la calidad e interrupción del servicio. Dentro de los servicios que un CERT puede brindar, además del tratamiento de incidentes de seguridad, podemos detallar servicios reactivos: avisos y alertas; tratamiento

y análisis forenses y servicios proactivos: auditoría de red, auditoría de sistemas/aplicaciones, monitoreo, detección de intrusiones, evaluación de seguridad de redes y Pentest de aplicaciones Web (proceso certificado ISO9001:2008 desde el año 2012). Asimismo se prestan servicios de calidad en la seguridad como son la consultoría y la concientización en Seguridad de la Información.

A fin de cumplir con el objetivo mencionado, los miembros de CERT-UNLP participan de la comunidad de LAC-CSIRTs, la cual reúne CERTs de América Latina y el Caribe y de las capacitaciones y encuentros que organiza OWASP. En lo que respecta a la Seguridad Informática, el CeSPI desde el año 2007 brinda certificados digitales para e-ciencia a través de una PKI registrada en TACAR, reconocida por TAGPMA e IGTF [9]. Respecto a los Pentests de aplicaciones web, el CERT-UNLP realiza pruebas de penetración sobre aplicaciones Web desarrolladas en la organización. Estos chequeos, que permiten identificar problemas de seguridad, están basados en recomendaciones de OWASP y se realizan en la fase de prueba [10].

En los chequeos habituales, es posible agrupar las vulnerabilidades que se estudian y el Pentest analiza básicamente: *protección de sesiones de usuario*: problemas en el manejo de sesiones, fallas de inyección en formularios de autenticación, problemas en el manejo del cambio de contraseñas, etc.; *protección de información en Base de datos*: inyecciones SQL, problemas en el almacenamiento criptográfico, problemas de encriptación de las comunicaciones; *protección de los usuarios y sus transacciones:* CSRF, fallas de inyección XSS, problemas de encriptación de las comunicaciones, problemas de Phishing, problemas de DOS, etc.; *protección de la Aplicación*: referencia insegura de objetos, configuración de seguridad inadecuada, revelación innecesaria de información, inclusión de archivos local y remota, etc.; *protección del Servidor*: puertos abiertos y protocolos habilitados innecesarios, análisis de servicios y versiones utilizadas, existencia de política de firewall, problemas de DOS.

## 4 Características del módulo de comunicación con Moodle

La interacción entre DSpace y Moodle se llevó a cab o utilizando el protocolo SWORD, que permite el depósito de forma remota hacia un repositorio o sistema de información. DSpace incorpora el protocolo SWORD de dos formas: como un servidor compatible, disponible desde la versión 1.8 en adelante; o como cliente, para hacer que DSpace deposite ítems en otros sistemas que acepten este protocolo. En el lado de Moodle, se implementó un módulo que utiliza una librería cliente provista por SWORD. El módulo creado se encarga de preparar el recurso a transferir al repositorio incorporándole los metadatos en un formato estándar según información de contexto dentro de la plataforma. El diseño de la funcionalidad del módulo se basó en una metodología que permitió definir la estructura del recurso a publicar y los criterios de clasificación automática.

La posibilidad de contar con este método de publicación alienta a los docentes que gestionan sus cursos en la plataforma educativa, a publicar los trabajos entregados por sus alumnos en uno o varios repositorios externos, ya que no necesitan conocer la interfaz y forma de acceso a ellos. Para realizar esta tarea primero se le presenta al docente un formulario donde se listan la estructura del repositorio, de forma tal de

permitirle seleccionar en qué unidad se depositará el envío. Para obtener estas colecciones se usa el protocolo REST. Luego el docente selecciona los trabajos enviar, introduce su nombre de usuario y contraseña propios del repositorio, y realiza el envío, para esta comunicación se utiliza el protocolo de interoperabilidad SWORD.

# 5 Testeos y corrección de vulnerabilidades del módulo

## 5.1 Aspectos de seguridad de Moodle

Moodle provee herramientas para manejar la seguridad del módulo, esto es, asociar permisos a los diferentes usuarios del módulo (docentes, docentes del curso, estudiantes, etc.), y luego controlar los mismos. Si un usuario pertenece a un grupo va a poseer todos los permisos asociados a ese grupo. Para configurar los permisos se deben definir las *capabilities*. Estos permisos nos permiten definir acciones y los niveles de seguridad requeridos para cada acción. Para validar que un usuario tenga los permisos asociados se debe utilizar el método *has_capability()*, que recibe como segundo parámetro el contexto del módulo del curso, esto se debe a que los permisos definidos corresponden justamente a la instancia del módulo en un curso determinado, como se observa en la Fig. 2.

```
29
30 require_once(dirname(dirname(dirname(__FILE__))).'/config.php');
31 require_once(dirname(__FILE__).'/lib.php');
32
33 require_login();
34 $cmid = required_param('id', PARAM_INT);
35 if (!$cm = get_coursemodule_from_id('sword', $cmid)) {
36     error("Course module ID was incorrect");
37 }
38 $context = context_module::instance($cm->id);
39
40 if(has_capability('mod/sword:view',$context)){
41
```

**Fig. 2** Se envía información de contexto para garantizar los permisos adecuados

## 5.2 Testeos y correcciones realizados en el módulo

Una vez finalizado el desarrollo del módulo, se solicitó una evaluación de seguridad al CERT-UNLP. El Pentest realizado permitió encontrar diversos errores. A continuación se detallan estos errores de seguridad, sus implicaciones y los cambios realizados con el fin de solucionar estos problemas:

1. *Ausencia de control de acceso a funciones*: este problema de seguridad permite que usuarios accedan a funcionalidad no autorizada para su perfil. Por ejemplo permite que usuarios autorizados accedan a funciones privadas, o también que usuarios anónimos accedan a funcionalidad propia de usuarios registrados.
Para evaluar si un sitio tiene esa vulnerabilidad se debe revisar la interfaz de cada posible usuario y verificar que no se muestren enlaces a funciones no autorizadas.

También se debe comprobar que se encuentren todos los chequeos de autenticación y autorización del lado del servidor. Estas verificaciones son fundamentales, ya que aunque el usuario no tenga un enlace directo a la funcionalidad, puede tratar de acceder a la misma a través de su URL.
En el caso del módulo, no se encontraban enlaces a funcionalidad no correspondiente al perfil de cada usuario, pero el problema que se presentaba era que no se validaba que el usuario estuviera autorizado a realizar la operación correspondiente en el servidor. Para solucionar esto se agregó código que verifica que el usuario tenga los permisos necesarios, por ejemplo las siguientes líneas de código muestran cómo se valida que un usuario tenga permiso para ejecutar una sección de código:

```
require_course_login($course);
require_login();
if(has_capability(
 'mod/sword:view',context_user::instance($USER->id)))
```

A su vez en los archivos que no son accedidos directamente en el flujo normal de un usuario, se agregan restricciones de seguridad para que no puedan ser operados directamente en caso de un ataque:

```
defined('MOODLE_INTERNAL') || die();
```

2. *Secuencia de Comandos en Sitios Cruzados (XSS):* Esta vulnerabilidad se presenta siempre que no se sanitizen todas las entradas de datos al sistema, para asegurar que éstas no tengan código malicioso. El problema de no realizar la sanitización es que existe la posibilidad que los datos sean llevados a la vista del usuario, donde se ejecute el código malicioso lo cual podría, por ejemplo, robar información de sesión del mismo. Para solucionar esta vulnerabilidad se deben tener en cuenta varias cuestiones. Antes que nada se debe asegurar que todos los datos ingresados al sistema, por sistemas externos o por usuarios, sean correctamente sanitizados. La sanitización consiste en obviar los caracteres especiales, con esto se logra evitar que se ejecute código presente en los datos. En vez de ejecutarse, el código será presentado al usuario como texto plano. En el caso particular del repositorio, se recibe un JSON con datos los cuales se presentan al usuario. En la versión previa a las modificaciones realizadas, la entrada de datos no era sanitizada produciendo la vulnerabilidad descrita. Para solucionar esto se utiliza la función *htmlspecialchars()*, que escapa los caracteres especiales, con lo cual cualquier código malicioso es presentado como texto simple, como se muestra en la Fig. 3. Además de usar la función ya mencionada, se utilizó la función *strip_tags()*,como vemos a continuación, de forma tal que el JSON recibido sea sanitizado de una forma en la cual, una vez realizada esta acción no tuviera código de ningún tipo ni tampoco tags HTML.

```
require_once($CFG->libdir.'/weblib.php');
function sec_print($s) {
 return htmlspecialchars(strip_tags($s), ENT_QUOTES);
}
function sec_print_array($arr){
  foreach ($arr as &$act) {
    if(!is_array($act)){
      $act = sec_print($act);
    }
    else{ sec_print_array($act);}
  }
  return $arr;
}
```

**Fig. 3** Cadena JSON sanitizado

3. *Redirecciones y reenvíos no validados:* esta vulnerabilidad consiste en que el sistema recibe una URL del usuario y no valida que la misma lleve a un sitio confiable. Para eliminar esta vulnerabilidad se debe:

   a. Inspeccionar el código para verificar que sólo se puedan realizar reenvíos o redirecciones a sitios confiables.

   b. Validar que los sitios supuestamente confiables no realizan ninguna redirección a un sitio no confiable. Para revisar esto se debe validar que al enviar un requerimiento HTTP a este sitio, no se reciba como respuesta un mensaje de HTTP que indique un reenvío (éstos son los que tienen código entre 300 y 307, el más usado para reenvíos es el que tiene código 302).

   En el caso particular del módulo no se validaba que las direcciones utilizadas correspondieran a sitios confiables. Inicialmente la configuración que determina hacia qué sitio se intenta realizar un depósito se basaba en una URL ingresada por el usuario en un campo de texto. Se modificó el formulario para que el usuario, en vez de ingresar la URL manualmente, deba seleccionar entre un conjunto de direcciones correspondientes a sitios remotos ya verificados. Esta modificación además dificulta la posibilidad de un ataque XSS.

4. *Almacenamiento de contraseñas sin hashear:* El almacenamiento de contraseñas de una forma segura es un punto clave para la seguridad de un sistema informático. Si un atacante logra tener acceso a una base de datos y las claves no se encuentran hasheadas, el atacante podrá conocer la contraseña de cada usuario lo cual compromete la seguridad del sistema. Para solucionar este problema lo que corresponde es hashear las contraseñas antes de almacenarlas en la base de datos. En el caso particular del módulo se deben tener en cuenta otras consideraciones, ya que lo que se almacenan no son contraseñas de Moodle, sino que son contraseñas del repositorio con el cual el módulo se comunica. En base a esto se deben tener algunas consideraciones extras, por ejemplo en el caso de almacenar contraseñas hasheadas el algoritmo de encriptación debería ser el mismo en ambas plataformas. Además el repositorio debe estar configurado para validar correctamente el acceso al recibir contraseñas ya encriptadas. Esto significa un alto nivel de acoplamiento entre el módulo y el repositorio, lo cual supone un problema si se realizan cambios en alguno de ellos. En base a las consideraciones mencionadas se optó por no almacenar las

contraseñas y pedirle al usuario que la introduzca junto con su nombre de usuario en el repositorio cada vez que realiza un envío, tal como se muestra en la Fig. 4.

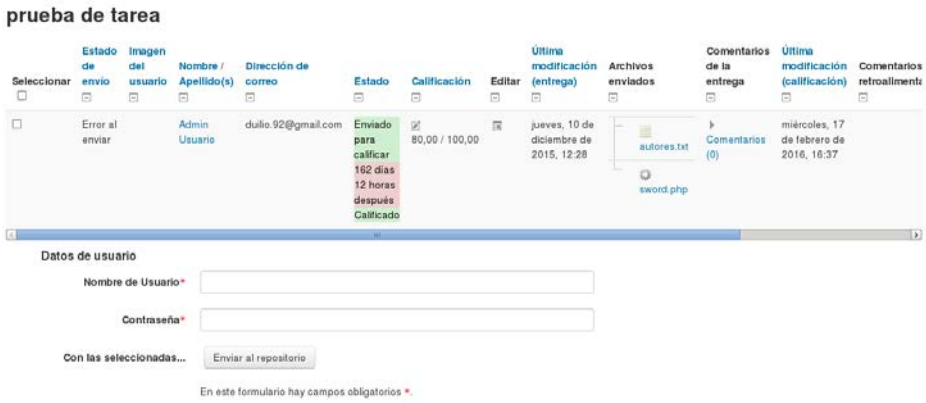


**Fig. 4** Página de envío de Tareas de Moodle al repositorio DSpace.

5. *Comunicación insegura con el repositorio:* Al establecer comunicaciones en la web se debe tener en cuenta consideraciones de seguridad consecuentes a la información que se envía y recibe. En este sentido existen diversos protocolos los cuales afrontan los posibles ataques de seguridad en distinta medida. HTTP es vulnerable a ataques man-in-the-middle y eavesdropping. Un ataque de man-in-the-middle consiste en que un tercero intercepte, lea y modifique los mensajes entre dos partes sin que éstas se enteren. Por otro lado un ataque de eavesdropping consiste simplemente en que un atacante tenga la posibilidad de leer los paquetes enviados entre las dos partes. HTTPS está diseñado para resistir esos ataques, Para hacer esto, provee una comunicación más segura usando SSL para encriptar y desencriptar mensajes, con esto se hace mucho más resistente a los ataques mencionados anteriormente. SSL provee seguridad en dos cuestiones fundamentales: en primer lugar garantiza que la comunicación se está realizando con el servidor correcto, esto se verifica revisando un certificado que éste envía. En segundo lugar se asegura la confidencialidad de los datos dado que el servidor y el cliente intercambian mensajes encriptados, de esta manera en caso en que un atacante pudiera leerlos, inicialmente no podría entender su contenido ya que no conoce como desencriptarlos. Aunque es mucho más seguro que HTTP, un ataque man-in-the-middle sigue siendo posible en HTTPS. Esto se debe a que para iniciar una comunicación segura, el cliente y el servidor deben intercambiar claves, estas claves serán utilizadas para desencriptar los mensajes.
La comunicación con el repositorio se hacía inicialmente usando el protocolo de interoperabilidad SWORD sobre el protocolo de comunicación HTTP. Tomando en cuenta la información sensible que se envía en estas comunicaciones (usuario y contraseña del repositorio) se decidió reemplazar el protocolo HTTP por el protocolo HTTPS, mejorando la seguridad de la comunicación del módulo con el repositorio.

Estas adecuaciones se llevaron a cabo en contacto con un representante del CERT-UNLP asignado al caso, para evacuar dudas y verificar procedimientos, en forma conjunta y efectiva.

## 6 Conclusiones

La integración entre plataformas se viene llevando a cabo desde hace tiempo en el marco de un proyecto global que se desarrolla en el LINTI. En una primera instancia a través de la comunicación del LMS Moodle con sistemas de gestión académica, redes sociales y repositorios digitales. En una segunda instancia, se avanzó en extender la funcionalidad del repositorio de manera de permitir su comunicación con otras herramientas muy utilizadas hoy en día. Las extensiones incorporadas tienen como objetivo ampliar el espacio propio del repositorio con el fin de integrarlo a plataformas de la nueva generación que van cambiando los hábitos de uso de la información. Tanto el repositorio como Moodle están basados en una plataforma de software libre, que permiten su personalización y adaptación a las necesidades de un proyecto concreto. Sin embargo, la adopción de herramientas abiertas y su modificación implica realizar un registro riguroso de las normas de seguridad para evitar vulnerabilidades y de esta manera poner en riesgo la información. El análisis se llevó a cabo en conjunto con el equipo de trabajo del CERT-UNLP, quienes evalúan las implementaciones que se realizan en el ámbito de la universidad y presentan un reporte exhaustivo de los errores que detectan. Este trabajo condujo la implementación del módulo de comunicación, hacia una versión ajustada que cumple con las normas de seguridad vigentes. Las correcciones realizadas permitieron evaluar exhaustivamente algunos aspectos claves, tales como los relacionados con la encriptación de claves y la organización de las distintas componentes del módulo para evitar que puedan ser accedidos en forma insegura. Por otro lado, en el repositorio se comenzó a usar SSL, norma impuesta según los requisitos del CERT, lo que llevó a realizar un cambio en la forma de comunicación. Los procesos de análisis, evaluación y corrección nos permitieron adaptar el módulo implementado a una versión segura utilizando tecnologías modernas que otorgan un mayor nivel de seguridad. Las modificaciones realizadas resultaban imprescindibles para la incorporación del módulo de comunicación en una plataforma de uso masivo como es el LMS Moodle que se encuentra en producción en la Facultad de Informática. La interacción con el equipo del CERT-UNLP nos posibilitó conocer y tener en cuenta determinados aspectos de seguridad que resultan cruciales en toda implementación y así poder establecer pautas y fijar metodologías de trabajo a seguir en los desarrollos que involucren las plataformas utilizadas en el proyecto de integración.

## Agradecimientos

Queremos agradecer a todo el equipo de CERT-UNLP que realizó el análisis y testeo del módulo implementado y nos asistió en forma permanente, orientándonos en las correcciones necesarias.

## Referencias


1. B. Cope y M. Kalantzis. Traducción Emilio Quintana. Aprendizaje Ubicuo. University of Illinois Press, 2009. 264 pp-
2. J. Díaz, A. Schiavoni, A. Osorio, P. Amadeo, E. Charnelli, "Integrating a Learning Management System with a Student Assignments Digital Repository. A Case Study", IADIS 2013, IADIS Multi Conference, Computer Science and Information Systems, e-Learning 2013, Praga, República Checa, 22 - 26 Julio, 2013.
3. J. Díaz, A. Schiavoni, P. Amadeo, E. Charnelli, "Diseño y construcción de objetos de aprendizaje. Su integración en repositorios y plataformas virtuales de aprendizaje", WICC 2012, XIV Workshop de Investigadores en Ciencias de la Computación, Posadas, Misiones, 26 y 27 de Abril, 2012.
4. J. Díaz, A. Schiavoni, P. Amadeo, E. Charnelli, J. Schulz, A. Humar. "Integrando un Repositorio Digital de Objetos de Aprendizaje con Servicios que Promuevan su Uso y Mantenimiento". LACLO 2014, IX Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje, Pág. 523-529, ISSN 1982-1611, Volumen 5, Open Access, Manizales, Colombia, 20-24 Octubre, 2014.
5. Cursos de Grado: https://catedras.info.unlp.edu.ar, Cursos de Postgrado: http://postgrado.linti.unlp.edu.ar y Cursos de Extensión: http://cursos.linti.unlp.edu.ar
6. Twitter Activity Module, https://github.com/mcharnelli/moodle-module_twitter
7. CERT-UNLP, Centro de Respuestas de Incidentes de Seguridad (CSIRT) Académico de la UNLP, http://www.cespi.unlp.edu.ar/cert
8. J. Díaz, P. Venosa, E. Lanfranco, N. Macia. "Definición e Implementación de un Centro de Atención de Incidentes (CERT) para un Á mbito Universitario". Anales de CACIC 2009, Universidad Nacional de Jujuy, octubre 5 al 9 de 2009 - ISBN 978-897-24068-4-1.
9. M. Dova, C. Grunfeld, F. Monticelli, M. Tripiana, A. Veiga, V. Ambrosi, A. Barbieri, J. Díaz, M. Luengo, N. Macia, L. Molinari, P. Venosa, M. Zabaljáuregui. "Progress of Grid Technology in Argentina: Lessons Learned from EELA". Anales de la III conferencia de EELA, Catania, pp.225-232. ISBN: 978-84-7834-565-6 http://www.eu-eela.org/3_conference/index.html
10. Open Web Application Security Project (OWASP): Top Ten Project: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project, Cheat Sheet Series: https://www.owasp.org/index.php/Cheat_Sheets, Testing Project: https://www.owasp.org/index.php/Category:OWASP_Testing_Project