Avoiding Sensitive Information Leakage in Moodle

Article · September 2013

DOI: 10.20533/licej.2040.2589.2013.0190

Víctor Gayoso Martínez
Spanish National Research Council
55 PUBLICATIONS **324** CITATIONS

Araceli Queiruga-Dios
Universidad de Salamanca
115 PUBLICATIONS **561** CITATIONS

Ascensión Hernández Encinas
Universidad de Salamanca
82 PUBLICATIONS **1,173** CITATIONS

Avoiding Sensitive Information Leakage in Moodle

V. Gayoso Martínez, L. Hernández Encinas

Institute of Physical and Information Technologies (ITEFI)
Spanish National Research Council (CSIC), Madrid, Spain

A. Queiruga Dios, A. Hernández Encinas, J. Martín Vaquero

Department of Applied Mathematics
E.T.S.I.I. of Béjar
University of Salamanca, Spain

Abstract

During the last years, the use of virtual learning frameworks has increased in the academic community. On account of the requirements derived from the Bologna process, many European universities started to change their education systems to new ones based on information and communication technologies. Those systems are most times based on web environments where the security is an essential issue. In this contribution, we provide an introduction about the e-learning platform Moodle, as well as an overview of the most important attacks against this system. Then, we focus on a specific type of attack that allows illegitimate users to obtain the username and password of other users when making a course backup in some specific versions of Moodle. In order to illustrate this information we describe a real attack against a Moodle 1.9.2 installation, which should encourage Moodle administrators to update their versions or backup configurations in case they are affected by the vulnerability described in this work. We complete our contribution with a list of security recommendations that can be used to secure any Moodle installation.[1]

1. Introduction

The educational model in many European universities has been changing during the last years in accordance with the proposals of the Bologna Process, which until 2010 was focused on creating a European Higher Education Area (EHEA). Since then, its aim is to consolidate the EHEA through the cooperation between ministries, higher education institutions, and students from 47 European countries [2]. The Bologna agreement has supposed a lot of changes in higher education, more specifically in the quality assurance, the contents of the subjects, the duration of the bachelor’s degree, and the education model itself.

Universities have started the implementation of the new teaching-learning techniques, moving from a blackboard-based education to a computer-based one, and from a teaching system, where teachers were mostly lecturers, to a new teaching-learning system, where students must plan their education and training carefully. In this sense, the use of computers and online platforms becomes an essential part of the student’s routine [3]. There are several tools and platforms in the market that provide the features needed for an online education. Sometimes they are used as the sole instrument for teaching, but in other cases they are used to support courses combining online and traditional classes [4, 5]. These type of products are usually known as Course Management System (CMS), Learning Management System (LMS) or Virtual Learning Environment (VLE).

The current offer is supported by free platforms, like Moodle [6]; commercial products, such as Blackboard [7]; and course aggregators like Coursera [8], edX [9] or Udacity [10]. For example, in a report published in 2009 about the VLEs used by the Spanish universities, it was stated that 55.55% of the universities used Moodle as their corporate platform for classes in 2008, 30% of the universities used no platform, and the rest used other platforms in different small percentages.

Moodle (Modular Object-Oriented Dynamic Learning Environment) is an e-learning software platform whose first version was released in August 2002 [11]. Moodle was originally developed by Martin Dougiamas to help educators create online courses with a focus on interaction and collaborative construction of contents, and is provided freely as open source software under the GNU General Public License. As of November 2013, it had a user base of 87,062 registered sites across 239 countries [12], serving almost eight million courses. In addition to the standard features of Moodle, developers can extend its functionality by creating specific plugins [13]. Due to all its features, Moodle is probably the most popular platform for online education [14].

Moodle can be run on Windows, Mac, and Linux operating systems. The most widely used version of Moodle is 1.9.x, and the latest stable version as of November 2013 is 2.5.3. Moodle is one of the platforms most used in the world, especially in USA, Spain, Brazil, and United Kingdom [12]. For this reason, its security is a crucial aspect which must be analysed in depth. The information stored at the Moodle platform includes elements such as user profiles, students' results and grades, examination and assessment questions, discussion forums contents, assignments submitted by the students, news and announcements, wikis, etc. This information is critical, and so it must be protected accordingly to its importance. After presenting several security aspects related to the use of Moodle, this contribution focuses in one of the most important threats, the information disclosure. This disclosure can be even more serious in education institutions, where many users access different services with the same credentials. In particular, we have analysed the possibility of obtaining the usernames and passwords of users in specific versions of Moodle which are currently installed in many educational centers and institutions. In fact, we prove that is very easy to obtain such information and get access, impersonating other users, to the system where those specific versions of Moodle are being used, unless the site administrators have taken the appropriate measures to avoid this risk. The rest of the paper is organized as follows: In section 2, we provide some information about the security of the VLEs in general and, more specifically, some potential vulnerabilities of Moodle. Section 3 describes the Moodle backup procedure. In Section 4, we have included all the steps that must be completed in order to install Moodle, either for conducting security tests or deploying the service in a real scenario. Section 5 presents a practical attack conducted against a production server. A list of additional security recommendations is included in Section 6. Finally, we summarize our conclusions in Section 7.

2. Moodle security

The proliferation of VLEs and the use of computers as part of daily classes in most levels of the educational system implied the necessity of providing communications between users and protecting the information delivered through those communication channels. VLE systems allow to share information and data among all the users, and they often manage one-to-many and many-to-many communications, providing powerful capabilities for learning [15].

E-learning can be considered as a special form of ebusiness, where its digital content has to be distributed, maintained, and updated. Moreover, these contents has to be adequately protected from unauthorized use and modification, and at the same time, they must be available to the students in a flexible way [16].

Discussions about how to protect web applications and the e-learning content from being used without permission, including the authentication system, and different types of availability, integrity, and confidentiality attacks, are presented in [17–20]. Moodle, as any other VLE, is open to attacks if vulnerabilities are found and they are not revised by the developers [21, 22]. Fortunately, some of the vulnerabilities of Moodle have already been satisfactorily corrected with the latest versions. At the Moodle website, the history of each version can be found, as well as the new features included in them [6]. A particularization to Moodle security vulnerabilities using the AICA model (whose name refers to the concepts of Availability, Integrity, Confidentiality, and Authentication) is analysed in [23]. After this analysis, the authors recommend some security and privacy protection mechanisms in order to minimize the vulnerabili-ties found. The suggestions about the setting configuration apply to Moodle administrators, but there are no recommendations in that article for end-users. Given that Moodle is managed as a corporative tool, teachers are not typically allowed to change settings or to decide upon any other change that could make the VLE secure against external or internal threats. From a broader point of view, some of the attacks against Moodle and the measures that must be taken into account in order to secure the virtual environment installation are presented in the following subsections [24].

2.1. Server security

An organization's servers provide a wide variety of services to internal and external users, and many servers also store or process sensitive information for the organization. Servers are frequently targeted by attackers because of the value of their data and services [25]. Some of the practical countermeasures to the most common attacks on servers are listed below:

- Firewall. Working closely with router programs, firewalls examine each network packet to decide to forward it toward its destination or not. A firewall also includes or works with a proxy server that makes network requests on behalf of workstation users. A firewall is often installed in a specially designated computer separate from the rest of the network so that no incoming request can get directly at private network resources.
- OS and software update. Keeping the operating system where Moodle is running up-to-date is a necessary mechanism to prevent attackers from using the most recent vulnerabilities discovered for that operating system. Other programs, though no directly related to Moodle, must be

also updated as they could create additional security gaps.

- Removing unnecessary software packages. Administrators must know exactly what is installed on their systems because otherwise it is more difficult to secure every program and service available. Because of that, administrators must remove unnecessary packages that do not comply with the established security policy.

2.2. Authentication

The term authentication refers to the means and processes used to corroborate the identity of a person, a computer terminal, a credit card, etc. Some of the threats related to the authentication phase in Moodle are described below:

- Weak passwords. As weak passwords are vulnerable to brute force and dictionary attacks, administrators are advised to control the security level of the users' passwords, so apart from regular letters, they should also include digits and special symbols. On the internet there are several online tools that allow users to generate strong random passwords, for example [26]. Another solution to this problem consists in using the bcrypt library [27], which uses the idea of adaptive hashing where the same block of code produces passwords that are hard to crack, and could be configured in setup time.
- Password change. A good security policy consists in forcing every user to change his/her password from time to time. The goal when implementing this feature is to limit the amount of time a lost, stolen, or forged credential can be used by someone else.
- Authentication roles. As it is the case of other software, Moodle has different types of users (students, teachers, administrators, etc.). The differentiation of roles prevents some type of users (e.g. students) to access resources managed by other users which are higher in the role hierarchy (e.g. teachers).
- Session hijacking. The term session hijacking refers to the exploitation of a valid computer session to gain unauthorized access to information or services in a computer system. In particular, it is tipically used to refer to the theft of a cookie that authenticates a user to a remote server. Since early versions [28], Moodle can run via https using SSL/TLS (Secure Sockets Layer/Transport Layer Security) to avoid session hijacking.

2.3. Protection against internet bots

Internet bots are software applications that run automated tasks applied to any content publicly available on the Internet. They usually perform tasks that are repetitive and trivial at a much higher speed than any human being can do.

- Protecting Moodle from unwanted search bots. Web search engines such as Google employ internet bots that obtain information about web sites. Moodle includes a configuration option that allows some search engines to enter the Moodle web site as guests, so administrators are encouraged not to habilitate this option if they want to prevent search engines to access the Moodle installation in these conditions.
- Protection against spam bots. As user profiles includes e-mail accounts, spam bots can try to gain access to Moodle using the guest role and retrieve information such as valid e-mail addresses from other user's profiles. Administrators that want to avoid this risk must configure the Moodle installation so guest users are not allowed to obtain the profile of other users.
- Protection against brute force attacks. Even though Moodle does not limit the number of incorrect login attempts before suspending and account (as this feature could be used as a denial of service attack), it allows to configure the number of login failures for a specific user (or from a specific IP address) that trigger the sending of a warning e-mail to the administrator.
  Another solution to avoid brute force attacks during the registration and/or authentication phase is to use captchas in order to make the procedure stronger, removing the possibility of a brute force attack. The captcha support was added to the email-based self-registration process in version 1.9.1 [29].

2.4. Data storage

In confidentiality attacks, the main purpose of an attacker is not data modification, but data access and dissemination. This flaw is based on the fact that sensitive information does not have an appropriate encryption. In fact, VLE systems rarely use properly cryptographic functions or strong algorithms to protect data and credentials. Regarding this threat, a new solution, named VAST, was proposed in 2005. VAST uses large, structured files to improve the secure storage of secret data, and preserves both the confidentiality and integrity of the data [30]. The VAST storage system uses extremely large (e.g., terabyte-sized) files to store secret information. A secret is broken into shares distributed over a large file, so that no single portion of the file holds recoverable information. Obviously, the main disadvantage of this system is the high cost in data storage that it implies.

3. Moodle backup

Data backup and recovery are essential parts of any professional service operation, as they permit to recover information from a loss originated by either human error or infrastructure failure. The backup procedure implemented in Moodle allows to save in a compressed file all the relevant information pertaining to a certain course. This is an important feature, as teachers might need to move their courses to another server, backup the courses before an upgrade, or just protect the course contents against potential errors. Backups are typically managed by administrators or by the teachers themselves, who can select the specific elements that will be included in the backup bundle by using a web menu. Figure 1 displays as an example the information selected for backup in one of our courses. The output of the backup process is a compressed file that, among other elements, includes the file moodle.xml (or moodle_backup.xml, depending on the version). Though there is no official Moodle documentation about moodle.xml, we will explain the structure and contents of this XML file. The first level of information of the XML file is formed by the elements INFO, ROLES, and COURSE, whose meaning is described next:

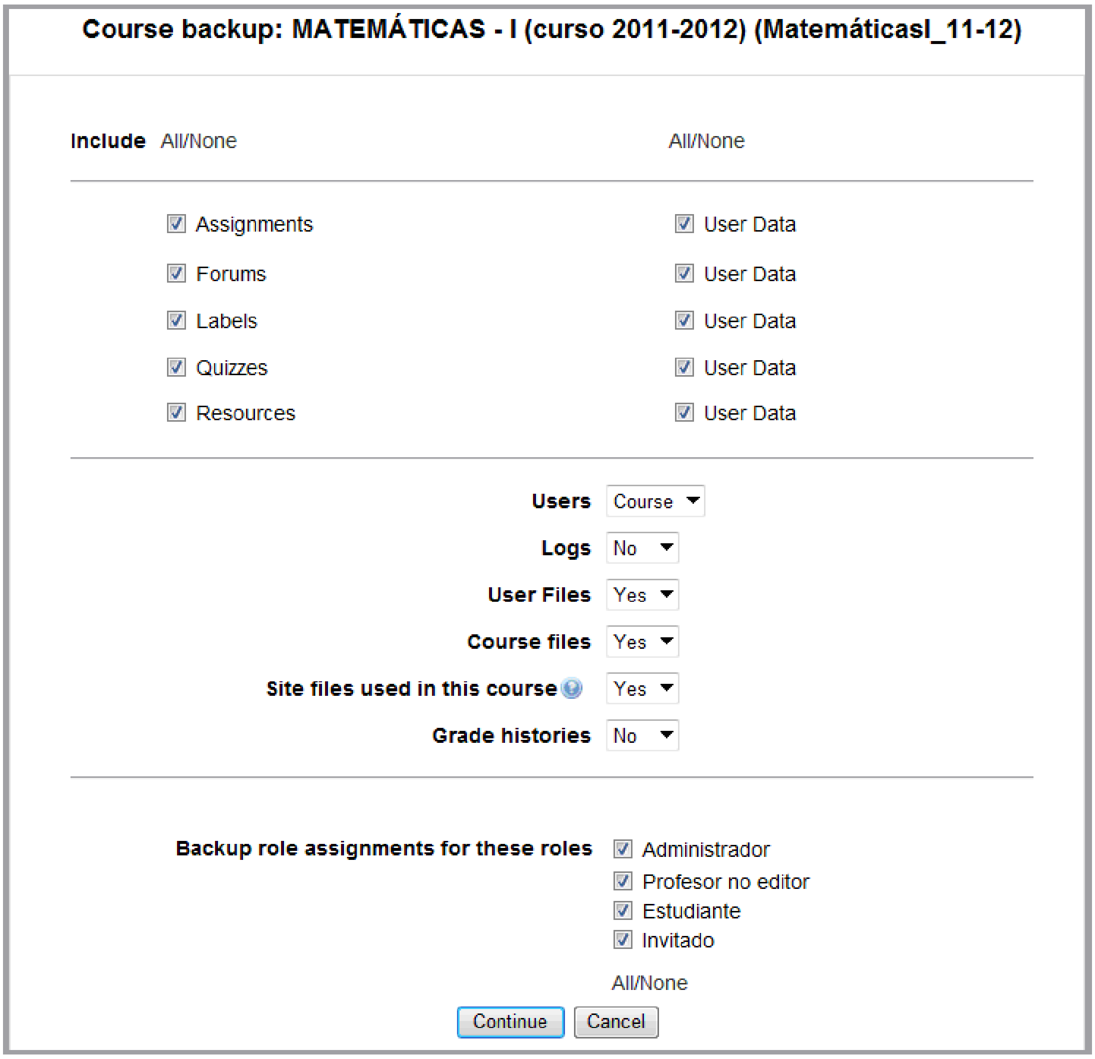
- INFO includes the name of the backup file (NAME); the Moodle version (MOODLE_VERSION); the Moodle release (MOODLE_RELEASE); the backup tool version (BACKUP_VERSION) and its release information (BACKUP_RELEASE); the URL of the site (ORIGINAL_WWWROOT); the storage method for the backup (ZIP_METHOD); the date of the backup (DATE); and general information about a set of elements included in the course such as assignments, quizzes, forums, lessons, resources, surveys, wikis, etc. (DETAILS), each one presented as a MOD element.
- ROLES is composed of several ROLE items. Each ROLE element gathers the information about the capabilities possessed by users of a certain type (e.g. teachers, students, etc.).
- COURSE is where the actual information about the course is located, and it is comprised of several elements: HEADER includes general information about the course (e.g. official name, starting date, etc.); BLOCKS informs about the different modules that can be accessed in the course web page; SECTION presents the information about several elements of the course (questionnaires, etc.); USERS includes important data about the users of the course (e-mail address, etc.); QUESTION_CATEGORIES includes information about the questions developed by the teacher that must be answered by the students; LOGS records the details of every user access (user ID, IP of the computer from which the user connected, etc.); GRADEBOOK stores the students' grades, and finally, MODULES includes information such as the resources (files uploaded by the teacher, etc.), and the posts published at the forums.

Within USERS, information about each user of the course is stored inside a USER element. Among this information, we can find the following items:

- USERNAME: The name that uniquely identifies any user in a particular Moodle deployment.
- PASSWORD: The hash of the user's access code computed with the MD5 algorithm.
- IDNUMBER: An identification number provided to the system for each user, for example, the national identity number, a corporate identification number, etc.
- FIRSTNAME: Given name of the user as registered in Moodle.
- LASTNAME: Family name of the user as registered in Moodle.
- EMAIL: User's e-mail address.

It is very important to remark that passwords are stored in Moodle as hashes processed by the MD5 algorithm, a 128-bit hash function designed by Ron Rivest in 1992 and published as a RFC (Request for Comments) document by the IETF (Internet Engineering Task Force) [31]. We recall that hash functions do not encrypt data as they use no keys; in other words, they only determine a summary (digest) of the data [32]. Before Moodle v1.9.7 (November 2009), hashed passwords were automatically stored in the backup files of the courses. Starting in version 1.9.7, this feature was disabled, so in the latest versions hashed passwords are not stored as part of the backup process. In fact, if a course is restored to a new site, users will need to reset their password the first time they connect. The problem with MD5 is that it has been proved that it is a weak algorithm. Several attacks have demonstrated that nowadays it is not a secure choice for a hashing algorithm [33–35]. On the internet there exist several websites that can retrieve the original text or message related to a given MD5 hash, either directly as an online service (e.g. [36], see Figure 2), or as a downloadable binary application (e.g. hashcat [37], a free tool to recover plain text strings for a variety of hashing methods).

Almost 20% of Moodle versions registered so far in education institutions are 1.8.x or previous [12]. Though no statistics are available about the exact version of Moodle that has been installed at the registered sites, taking into account that the latest 1.9.x version was 1.9.18, and that the Moodle



Course backup: MATEMÁTICAS - I (curso 2011-2012) (MatemáticasI_11-12)

Include All/None	All/None
☑ Assignments	☑ User Data
☑ Forums	☑ User Data
☑ Labels	☑ User Data
☑ Quizzes	☑ User Data
☑ Resources	☑ User Data

Users Course

Logs No

User Files Yes

Course files Yes

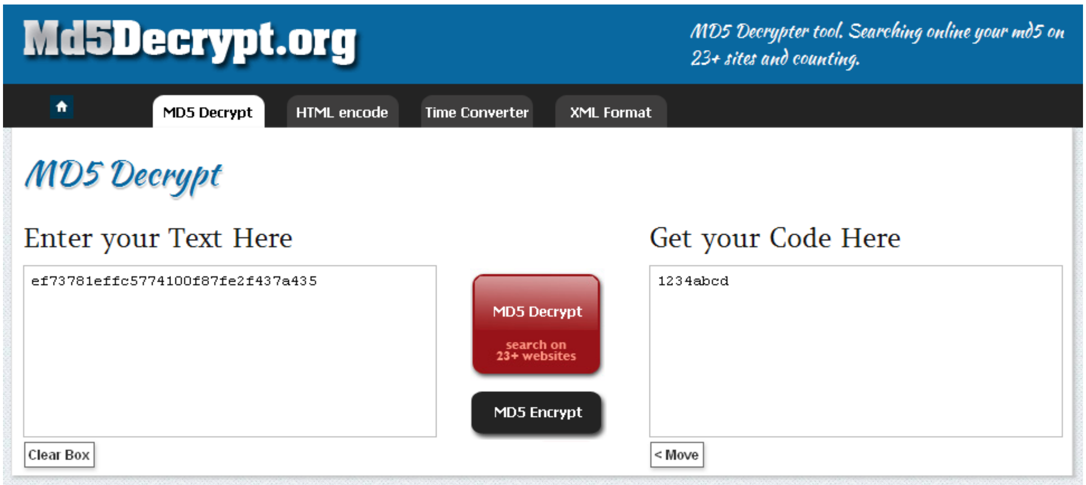
Site files used in this course Yes

Grade histories No

Backup role assignments for these roles ☑ Administrador
☑ Profesor no editor
☑ Estudiante
☑ Invitado

All/None

Figure 1. Moodle backup options



Md5Decrypt.org
MD5 Decrypter tool. Searching online your md5 on 23+ sites and counting.
MD5 Decrypt
HTML encode
Time Converter
XML Format
MD5 Decrypt
MD5 Decrypt search on 23+ websites
MD5 Encrypt
Clear Box
< Move

Enter your Text Here

```
ef73781effc5774100f87fe2f437a435
```

Get your Code Here

```
1234abcd
```

Figure 2. Online MD5 recovery tool

versions vulnerable to this problem are versions previous to 1.9.7, it is reasonable to expect that a proportion of 1.9.x sites are vulnerable, though with the available data it is not possible to provide an exact figure. In summary, with the data publicly available at [12], it can be stated that the percentage of affected sites could range from 20% to almost 80%. Figure 3 shows the distribution of all Moodle registrations by version as of November 2013.

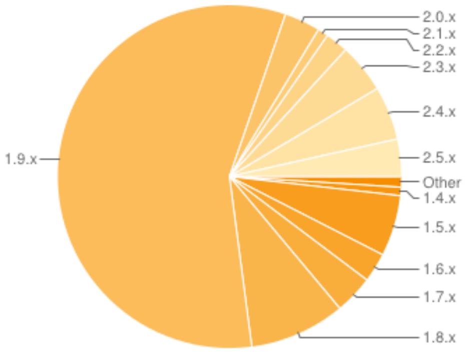


Figure 3. Moodle registrations by version (date: November 2013)

4. Testing environment

Before using the Moodle vulnerability mentioned in the next section in a real environment, we decided to try it in a testing site controlled by us. As a summary of our investigation, in this section we will show how to install and configure Moodle 1.9.6 (the last vulnerable version) in a Linux computer acting as the system admin. The first phase consists in downloading, installing, and running XAMPP, an open source cross-platform solution which includes, among other elements, the Apache web server and a MySQL database manager [38]. The steps that must be performed in this phase are the following:

1. Download the file xampp-linux-1.8.3-1-installer.run from http://sourceforge.net/projects/xampp/files/latest/download.
2. Modify the execution rights with the console command chmod 755 xampp-linux-1.8.3-1-installer.run.
3. Install the XAMPP package in the location /opt/lampp using the command ./xampp-linux-1.8.3-1-installer.run.
4. Start the XAMPP services with the command /opt/lampp/lampp start, and check its status with the command /opt/lampp/lamppstatus.
5. If everything is correct, after entering the address http://localhost at the web browser, the user will get the XAMPP welcome page, where he can choose the language for administering the web server.

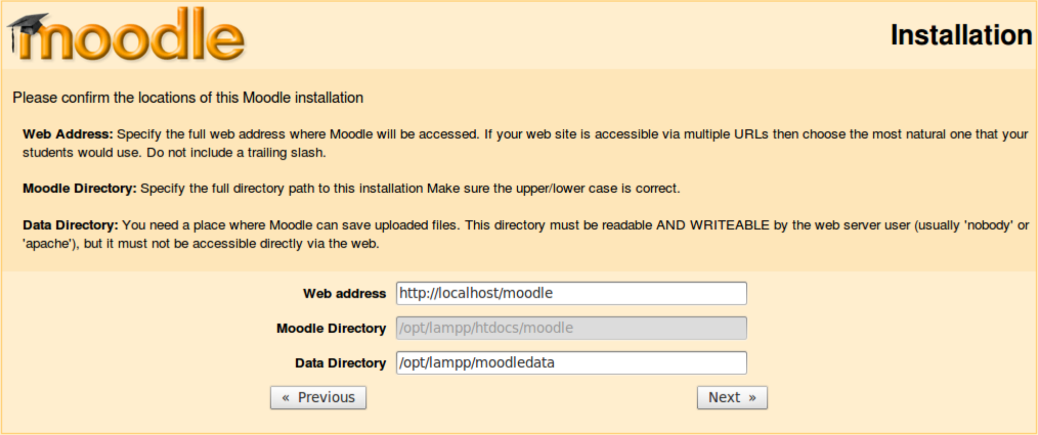
The next phase consists in downloading Moodle 1.9.6 and copying it to the proper location. In detail, the steps that must be performed are these:

1. Download the file moodle-1.9.6.zip from the Moodle repository using the address http://download.moodle.org/download.php/stable19/moodle-1.9.6.zip.
2. Extract the contents of the zip file into the directory /opt/lampp/htdocs, access the directory /opt/lampp/htdocs/moodle, and execute the command chmod 777 *.
3. Create the subdirectory moodledata in the location /opt/lampp/ and change its permissions with the command chmod 777/opt/lampp/moodledata.

Next, it is necessary to create a new database so it can be used later by Moodle. For this task, we need to perform the following steps:

1. Enter the address http://localhost/xampp at the web browser and select the phpMyAdmin link placed at the Tools section.
2. In the new page, select the Databases element and create a new database with the name moodledb, selecting the option utf8_unicode_ci as collation.
3. Going back to the Databases section, we must click on the Check Privileges link associated to moodledb, which will allow us to create a new administrator user. The example data that we have used in this step consists on the name usermoodle and the password 123456, inserting in the Host text box the value local host.
4. Before finishing the user creation process, it is necessary to click on the Check All link under the Global Privileges title, which will grant all the managing rights to our newly created user.

The last phase consists in the installation and configuration of Moodle. In order to accomplish this, we must enter the address http://localhost/moodle at the web browser and follow the instructions given during the process. The data that we have used in our testing environment is shown in Figures 4 and 5. Using this testing environment, we created a course named C101, enrolled two fake students in it and generated a backup file. After retrieving the file moodle.xml from the zipped backup, we were able to confirm that the MD5 hash of the password of both students and the administrator were included in the backup. The text below is a moodle.xml excerpt which contains one of the hashed passwords.



Please confirm the locations of this Moodle installation

Web Address: Specify the full web address where Moodle will be accessed. If your web site is accessible via multiple URLs then choose the most natural one that your students would use. Do not include a trailing slash.

Moodle Directory: Specify the full directory path to this installation Make sure the upper/lower case is correct.

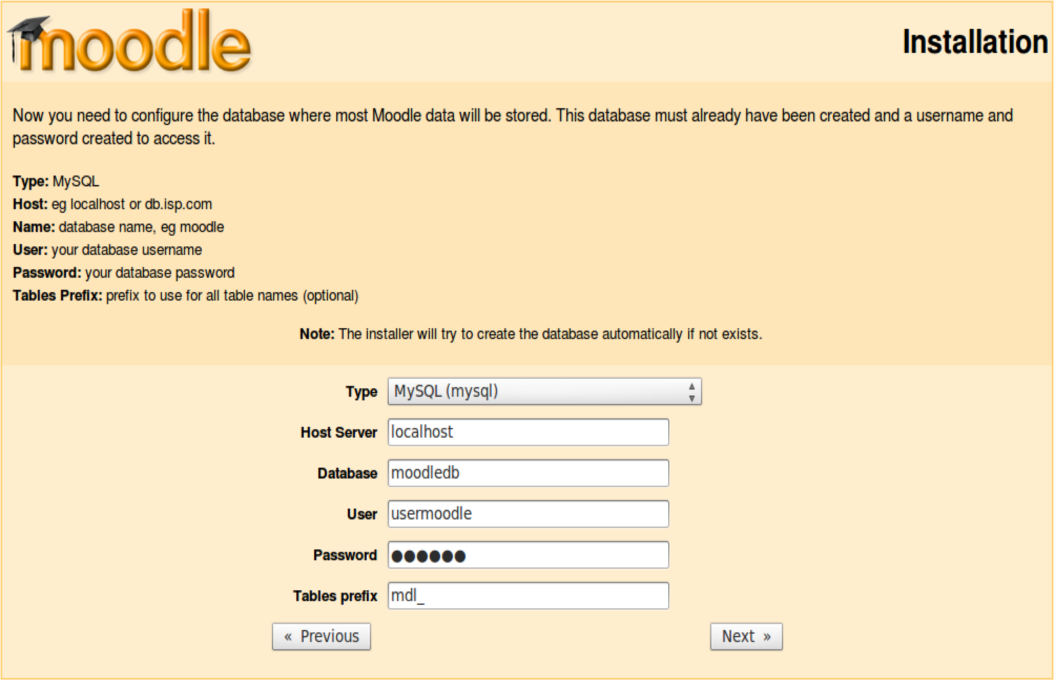
Data Directory: You need a place where Moodle can save uploaded files. This directory must be readable AND WRITEABLE by the web server user (usually 'nobody' or 'apache'), but it must not be accessible directly via the web.

Web address http://localhost/moodle

Moodle Directory /opt/lampp/htdocs/moodle

Data Directory /opt/lampp/moodledata

Figure 4. Locations of the Moodle installation



Now you need to configure the database where most Moodle data will be stored. This database must already have been created and a username and password created to access it.

Type: MySQL
Host: eg localhost or db.isp.com
Name: database name, eg moodle
User: your database username
Password: your database password
Tables Prefix: prefix to use for all table names (optional)

Note: The installer will try to create the database automatically if not exists.

Type MySQL (mysql)

Host Server localhost

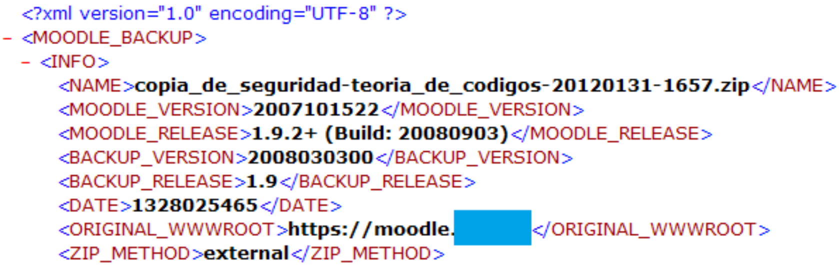
Database moodledb

User usermoodle

Password ●●●●●●

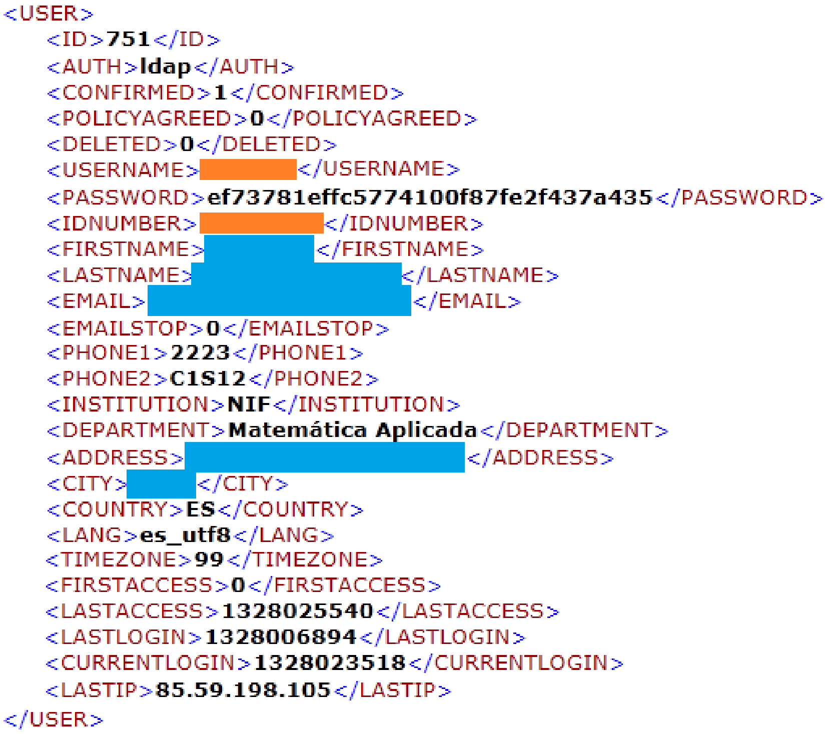
Tables prefix mdl_

Figure 5. Moodle database information



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <MOODLE_BACKUP>
  - <INFO>
      <NAME>copia_de_seguridad-teoria_de_codigos-20120131-1657.zip</NAME>
      <MOODLE_VERSION>2007101522</MOODLE_VERSION>
      <MOODLE_RELEASE>1.9.2+ (Build: 20080903)</MOODLE_RELEASE>
      <BACKUP_VERSION>2008030300</BACKUP_VERSION>
      <BACKUP_RELEASE>1.9</BACKUP_RELEASE>
      <DATE>1328025465</DATE>
      <ORIGINAL_WWWROOT>https://moodle.[redacted]</ORIGINAL_WWWROOT>
      <ZIP_METHOD>external</ZIP_METHOD>
```

Figure 6. Backup information contained in the moodle_backup.xml file



```
<USER>
    <ID>751</ID>
    <AUTH>ldap</AUTH>
    <CONFIRMED>1</CONFIRMED>
    <POLICYAGREED>0</POLICYAGREED>
    <DELETED>0</DELETED>
    <USERNAME>[redacted]</USERNAME>
    <PASSWORD>ef73781effc5774100f87fe2f437a435</PASSWORD>
    <IDNUMBER>[redacted]</IDNUMBER>
    <FIRSTNAME>[redacted]</FIRSTNAME>
    <LASTNAME>[redacted]</LASTNAME>
    <EMAIL>[redacted]</EMAIL>
    <EMAILSTOP>0</EMAILSTOP>
    <PHONE1>2223</PHONE1>
    <PHONE2>C1S12</PHONE2>
    <INSTITUTION>NIF</INSTITUTION>
    <DEPARTMENT>Matemática Aplicada</DEPARTMENT>
    <ADDRESS>[redacted]</ADDRESS>
    <CITY>[redacted]</CITY>
    <COUNTRY>ES</COUNTRY>
    <LANG>es_utf8</LANG>
    <TIMEZONE>99</TIMEZONE>
    <FIRSTACCESS>0</FIRSTACCESS>
    <LASTACCESS>1328025540</LASTACCESS>
    <LASTLOGIN>1328006894</LASTLOGIN>
    <CURRENTLOGIN>1328023518</CURRENTLOGIN>
    <LASTIP>85.59.198.105</LASTIP>
</USER>
```

Figure 7. User information in the moodle_backup.xml file used in the attack

```
<USERNAME>user1</USERNAME>
<PASSWORD>25d55ad283aa400af464c76d713c07
ad
</PASSWORD>
<IDNUMBER></IDNUMBER>
<FIRSTNAME>User</FIRSTNAME>
<LASTNAME>One</LASTNAME>
<EMAIL>userone@test.com</EMAIL>
```

In order to check that the vulnerability had been solved in Moodle 1.9.7, we repeated the installation process with this version. After the installation, we created again the C101 course, enrolled the same two ficticious users, and generated another backup file. Within the backup we located the moodle.xml file and compared its content with the one generated with the previous installation. As a result, we confirmed that effectively the hashed passwords included in the <PASSWORD> items had been removed from the backup file.

5. Real environment

After confirming that the attack could be effectively performed, we decided to try it in a real working scenario using the backup of one of our courses. Figure 6 shows part of the INFO structure of the file moodle_backup.xml managed by the Moodle 1.9.2 installation that we have used as the target of the attack. Besides, Figure 7 presents the content related to one of the USER elements included in the same file.

Tables 1 and 2 show the summary of the information about two different users whose details are stored in the moodle_backup.xml target file.

Table 1. Data from user 1 retrieved from the moodle_backup.xml target file

Backup data	User 1 information
USERNAME	X12345 (fictitious)
PASSWORD	ef73781effc57741 00f87fe2f437a435
FIRSTNAME	Alice (fictitious)
LASTNAME	Smith (fictitious)
EMAIL	alice@edu.com (fictitious)

Table 2. Data from user 2 retrieved from the moodle_backup.xml target file

Backup data	User 2 information
USERNAME	X54321 (fictitious)
PASSWORD	8fe87226333c05d4 996c46a0d4165cb7
FIRSTNAME	Bob (fictitious)
LASTNAME	Smith (fictitious)
EMAIL	bob@edu.com (fictitious)

If we take the MD5 passwords and feed one of the MD5 cracking tools with that information, we will obtain almost immediately the actual passwords. Table 3 present the real passwords recovered using this method.

Table 3. Actual passwords retrieved with a MD5 cracking tool

PASSWORD	DECRYPTED
ef73781effc57741 00f87fe2f437a435	1234abcd
8fe87226333c05d4 996c46a0d4165cb7	maria08

As a result of this research, we have proved that the possibility of accessing to the backup files of some versions of Moodle permits an attacker to obtain the MD5 hashed passwords, which is a very important weakness. In fact, such information allows the attacker to disclose the password of any user in a very efficient way. It must be taken into account that, in general, each teacher in a Moodle course is able to create a backup file that will include important information from other teachers involved in the course and from the students attending the course, such as their complete names, emails or hashed passwords. This risk is even greater in education institutions where users have the same password not only to login into Moodle, but also to authenticate to the e-mail service, to get the payroll data, or to grade the students' work, to name just a few examples.

6. Additional security recommendations

Apart from updating Moodle to a version which does not include the hashed passwords in the backup, there are several configuration options that should be taken into account when deploying Moodle in a production platform. The following list includes the most important options that must be set by the site administrator in order to create a secure and privacy-aware version of Moodle.

1. Include Moodle user data: Do not include user data in the automated backups (in case they are performed) in order to prevent undesired dissemination of names, emails, etc.
2. Protect usernames: By selecting this option, the file forget_password.php does not display any hint that would allow guessing of usernames or email addresses.
3. Self registration: It is recommended not to allow self registration, so potential users cannot register themselves and create accounts.
4. Password policy: Force Moodle to check user passwords against a valid password policy. It is advisable that passwords consist of upper-case and lower-case letters, digits, and non-alphanumeric characters.
5. Force users to login: When activating this option, users must login before accessing the site front page.
6. Force user to login for profiles: This setting forces individuals to login as a real (non-guest) account before viewing any user's profile.
7. Open to Google: Do not allow Google to enter the site as guest.
8. Required Flash player version: This option allows defining the minimum supported Flash version. It is important to update regularly this setting, as the Adobe Flash plug-in is known to be vulnerable to attacks and Adobe releases new versions every few weeks.
9. Use HTTPS for logins: Force Moodle to use a secure HTTPS connection for the login page, reverting afterwards to HTTP for performance reasons.
10. Use Clam AV on uploaded files: In the server workload allows it, use the Clam antivirus to scan all uploaded files.

7. Conclusions

One of the most widely used e-learning platforms, particularly in North American, Spanish, Brazilian, and British educational centers, is the Virtual Learning Environment Moodle. This virtual campus allows online interactions between teachers and students, either as a complement of traditional education or as the teaching tool for distance learning. This type of services accessed through the Internet must be completely secure, since the information exchanged between users is usually confidential, and the data stored at the platform can be used to validate the students' grades.

In this paper we have analysed the security of Moodle regarding a specific type of data storage attack which can lead to valid authentications from illegitimate users. The example that illustrates this attack shows that it is very easy for a teacher to create a backup file from a course and obtain the username and password of other users.

As a result of that research, we have proved that a significant proportion of Moodle installations are vulnerable to this attack, so it is paramount that Moodle administrators update the version of their Moodle installations at least to version 1.9.7, or that they configure the backup file in order to avoid sensitive information be included in it. In addition to that, administrators are recommended to configure several Moodle settings in order to avoid the leakage of sensitive information belonging to the users.

8. Acknowledgment

This work has been partially supported by Ministerio de Ciencia e Innovación (Spain) under the grant TIN2011-22668.

References

[1] V. Gayoso Martínez, L. Hernández Encinas, A. Hernández Encinas, and A. Queiruga Dios, “Disclosure of sensitive information in the Virtual Learning Environment Moodle,” in 6th International Conference on Computational Intelligence in Security for Information Systems, 2013, pp. 517–526.

[2] European Higher Education Area. (2010) European higher education area website 2010-2020. [Online]. Available: http://www.ehea.info

[3] J. A. González, L. Jover, E. Cobo, and P. Muño, “A web-based learning tool improves student performance in statistics: A randomized masked trial,” Computers & Education, vol. 55, no. 2, pp. 704–713, 2010.

[4] L. Ma, D. Vogel, and C. Wagner, “Will virtual education initiatives succeed?” Information Technology and Management, vol. 1, no. 4, pp. 209–227, 2000.

[5] G. E. McCray, “The hybrid course: Merging online instruction and the traditional classroom,” Information Technology and Management, vol. 1, no. 4, pp. 307–327, 2000.

[6] Moodle. (2012) Moodle.org: Open-source community-based tools for learning. [Online]. Available: http://moodle.org

[7] Blackboard. (2013) Blackboard. [Online]. Available: http://uki.blackboard.com/sites/international/globalmaster/

[8] Coursera. (2013) Coursera. [Online]. Available: https://www.coursera.org/

[9] edX. (2013) edx. [Online]. Available: https://www.edX.org/

[10] Udacity. (2013) Advance your education with free college courses online - Udacity. [Online]. Available: https://www.udacity.com/

[11] Moodle. (2012) Moodle.org: About. [Online]. Available: http://moodle.org/about/

[12] Moodle. (2012) Moodle.org: Moodle statistics. [Online]. Available: http://moodle.org/stats/

[13] E. Gutiérrez, M. A. Trenas, J. Ramos, F. Corbera, and S. Romero, “A new Moodle module supporting automatic verification of VHDL-based assignments,” Computers&Education, vol. 54, no. 2, pp. 562–577, 2010.

[14] T. Martín-Blas and A. Serrano-Fernández, “The role of new technologies in the learning process: Moodle as a teaching tool in Physics,” Computers & Education, vol. 52, no. 1, pp. 35–44, 2009.

[15] D. C. Luminita, “Information security in elearning platforms,” Procedia-Social and Behavioral Sciences, vol. 15, pp. 2689–2693, 2011.

[16] Z. F. Zamzuri, M. Manaf, A. Ahmad, and Y. Yunus, “Computer security threats towards the e-learning system assets,” in Software Engineering and Computer Systems, ser. Communications in Computer and Information Science, J. M. Zain, W. M. b. Wan Mohd, and E. El-Qawasmeh, Eds. Springer Berlin/Heidelberg, 2011, vol. 180, pp. 335–345.

[17] D. Bradbury, “The dangers of badly formed websites,” Computer Fraud & Security, vol. January, pp. 12–14, 2012.

[18] F. Graf, “Providing security for eLearning,” Computers & Graphics, vol. 26, no. 2, pp. 355–365, 2002.

[19] M. Nickolova and E. Nickolov, “Threat model for user security in e-learning systems,” International Journal of Information Technologies and Knowledge, vol. 1, pp. 341–347, 2007.

[20] T. Scholte, D. Balzarotti, and E. Kirda, “Have things changed now? An empirical study on input validation vulnerabilities in web applications,” Computers & Security, vol. 31, no. 3, pp. 344–356, 2012.

[21] J. Diaz, D. Arroyo, and F. Rodriguez, “An approach for adapting Moodle into a secure infrastructure,” in Computational Intelligence in Security for Information Systems, ser. Lecture Notes in Computer Science, A. Herrero and E. Corchado, Eds. Springer Berlin/Heidelberg, 2011, vol. 6694, pp. 214–221.

[22] S. Kumar and K. Dutta, “Investigation on security in LMS Moodle,” International Journal of Information Technology and Knowledge Management, vol. 4, no. 1, pp. 233–238, 2011.

[23] Z. Stapic, T. Orehovacki, and M. Danic, “Determination of optimal security settings for LMS Moodle,” in Proceedings of 31st MIPRO International Convention on Information Systems Security, 2008, pp. 84–89.

[24] D. Mileti´c, Moodle Security. Birmingham, UK: Packt Publishing, 2011.
[25] NIST, Guide to General Server Security, National Institute of Standards and Technology SP 800-123, 2008.

[26] A. Melnikov. (2012) Free password generator. [Online]. Available: http://www. freepasswordgenerator.com

[27] N. Provos and Mazières, “A future-adaptable password scheme,” in USENIX Annual Technical Conference, FREENIX Track, 1999, pp. 81–91.

[28] Moodle. (2008) Moodle 1.5.2 release notes - MoodleDocs. [Online]. Available: http://docs.moodle.org/dev/Moodle_1.5.2_release_notes

[29] Moodle. (2012) Moodle 1.9.1 release notes - MoodleDocs. [Online]. Available: http://docs. moodle.org/dev/Moodle_1.9.1_release_notes

[30] D. Dagon, W. Lee, and R. Lipton, “Protecting secret data from insider attacks,” in Financial Cryptography and Data Security, ser. Lecture Notes in Computer Science, A. Patrick and M. Yung, Eds. Springer Berlin/Heidelberg, 2005, vol. 3570, pp.16–30.

[31] R. Rivest, RFC 1321–The MD5 Message-Digest Algorithm, Internet Engineering Task Force, 1992.

[32] A. J. Menezes, S. A. Vanstone, and P. C. V. Oorschot, Handbook of Applied Cryptography. Boca Raton, FL, USA: CRC Press, Inc., 1996.

[33] X. Wang and H. Yu, “How to break MD5 and other hash functions,” in Advances in Cryptology, EUROCRYPT 2005, ser. Lecture Notes in Computer Science, R. Cramer, Ed. Springer Berlin/Heidelberg, 2005, vol. 3494, pp. 561–561.

[34] A. Sotirov, M. Stevens, J. Appelbaum, A. Lenstra, D. Molnar, D. A. Osvik, and B. de Weger, “MD5 considered harmful today. Announced at the 25th chaos communication congress,” 2008.

[35] Y. Sasaki and K. Aoki, “Finding preimages in full MD5 faster than exhaustive search,” in Advances in Cryptology, EUROCRYPT 2009, ser. Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 2009, vol. 5479, pp. 134–152.

[36] L. Forchino. (2012) MD5 decrypt online. [Online]. Available: http://www.md5decrypt.org

[37] hashcat. (2012) Hashcat–advanced password recovery. [Online]. Available: http://hashcat.net

[38] Apache Friends. (2013) Apache friends - XAMPP. [Online]. Available: http://www.apachefriends. org/en/xampp.html