Securing E-learning Platforms

Mohammad Derawi
Smart Wireless Systems
Gjøvik University College, Norway
Email: mohammad.derawi@hig.no

***Abstract*—E-learning is becoming an increasingly popular form of education but what about security? That is an important issue in the actual educational context where e-learning increases in popularity and more and more people are taking online courses. The approach to teach many students at the same time, and the students possibility to learn material at their own pace have been very successful. There are many important elements that must be taken into consideration: access control, authentication, data integrity, content protection, etc. Information security can be obtained using functions such as cryptography and network protocols. In this paper, we will highpoint some important security issues that must be taken into consideration in developing and using an e-learning platform. We will also inspect some security aspects of one of the most popular open-source e-learning systems.**

***Index Terms*—Information Security, e-learning, education, attacks, Moodle.**

I. Introduction

E-learning eases and improves the learning process through the usage of devices based on computer and communications technology. E-learning covers a extensive category of applications and processes, such as education via the Internet / Intranet (web based learning), education provided via computer (computer based learning), virtual classrooms and digital collaboration. The material is offered electronically over the Internet, Intranet, audio or video tapes, satellite or DVD/BluRay. Usually the e-learning term is understood as online education (web based learning) and online courses. Seeing this aspect, the computer based learning process can be understood as an elearning component, which does not require a nonstop interaction with an coach and other students as illustrated in Figure 2 . E-learning offers substantial advantages to companies and it is perfect adapted to specific and exact training in business. [10]

E-learning is a way of distance learning since the volunteers and the tutor can exist in in dissimilar locations, and the communication is mostly asynchronous.

Underneath is a list of characteristics of the e-learning systems presented :

- the learning process takes place in a virtual classroom;
- the educational material is accessible on the Internet and contains text, images, links to additional online resources, images, audio and video presentations;
- the virtual classroom is supervised by an tutor who plans the activity of the students, debates aspects of the course applying a discussion forum or chat, delivers auxiliary resources, etc;

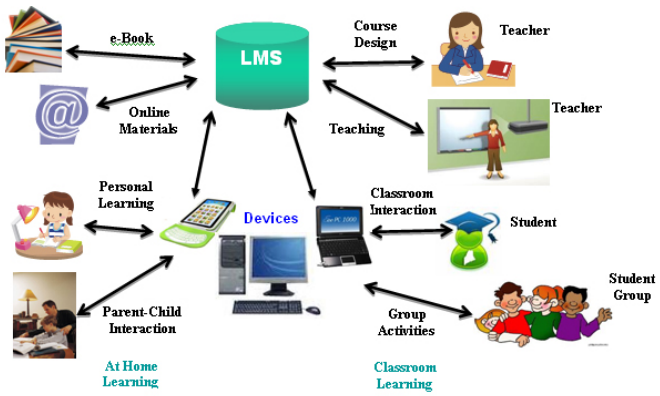

Fig. 1. E-learning Management System (e-LMS) [1]

- the learning is converted into a social process; a learning community is shaped via the interaction and collaboration between the tutor and students;
- almost all e-learning systems permit the activity monitoring of the participants, and some of them also simulations, the work on subgroups, audio and video interaction, etc [2].

Due to the novel tendencies in development of educational systems and the need of developing applications that can be retrieved remotely, the security management of e-learning systems and the access control have involved progressively more the attention of researchers and web applications developers.

Meeting the security requirements in an e-learning system is an tremendously multifaceted problem because it is required to shield the content, services and the personal data not only for the external users, but also for the internal consumers, including system administrators.

II. Security Requirements

The following elementary security aspects should be obeyed for any kind of e-learning platforms: confidentiality, integrity, authenticity (CIA), access control, availability, non-repudiation.

A protected authentication is obligatory to recognize the user who will apply the web application and to control his access privileges. This method avoids the attackers to entree another user's account, to view sensitive information or to perform unauthorized processes. In addition, once authenticated, the user should have the option to change his password.

Below are listed decent practices concerning authentication:

978-1-4799-5739-2/14/$31.00 ©2014 IEEE

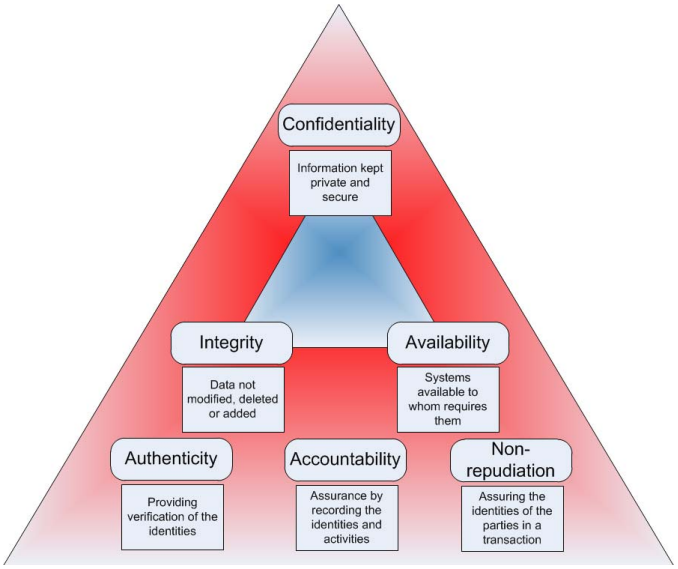

Fig. 2. Basic Requirements for Security

- Demanding re-authentication at specified time intervals;
- Put in force the users to use strong passwords (including some capital letters, at least one number, some special characters, etc);
- Implementing the access control based on dissimilar roles.

The *access control* is available throughout the authentication stage when the user is approved with all the essential rights. In this manner, the user will execute in the system only his permissible operations.

The *role-based authorization* model is an tactic of limiting the system access of unapproved users; it permits groups definition, users inclusion in groups or even groups in another groups. This model lets a flexible and rough control of the rights of each user. The permissions of executing certain operations are allocated to some specific roles (administrator, editor, instructor, student, registered user, unregistered user, etc). Staff members (or other system users) are assigned with specific roles, and thus they obtain the permissions to perform particular system functions. Since the users doesn't have the permissions directly allocated to them, but they only acquire them through their role or roles), management of individual user rights is converted into a matter of simply assigning appropriate roles to the user.

The *confidentiality* of an e-learning system must be guaranteed via the access control to resources and by securing the transmission and the storage of the data.

Integrity of the data and programs is an extremely important subject even supposing it is frequently abandoned in everyday life. Integrity means that only authorized subjects (i.e. users or computer programs) are allowed to adjust data (or executable programs). Secrecy of data is thoroughly connected to the integrity of programs and operating systems. If the integrity of the operating system is despoiled, then the reference monitor might not work correctly any longer. The reference monitor is a mechanism that protects that only authorized subjects are able to entree data and perform actions. It is clear that secrecy of information cannot be definite if the mechanism that checks and limits access to data is not working. Therefore, it is significant to shelter the integrity of operating systems in order to protect the secrecy of data itself [4]. The data replication through dissimilar sites represents a decent security practice that assistances to uphold the integrity. The replication process rises the security of the system and also increases the speed of data operations [5].

Non-repudiation means that users are not able to reasonably deny to have carried out operations. Assuming that a tutor erases his/her student's exam results. In this case, it should be likely to trace back who removed them by looking into some log files. Furthermore, these log files must be dependable and tamper-proof. Auditing is the method used to achieve these prerequisites. Another countermeasure for non-repudiation is digital signature.

III. E-LEARNING PLATFORM SECURITY

In this section, we will attempt to highpoint some explicit security issues of e-learning platforms. When it is implemented, an e-learning platform should be verified for external intrusion issues using functions like:

- XSS (or Cross Side Scripting);
- Sql injection in the site address (URL SQL injection);
- Remote injection applying a virus/trojan/malware file;
- Direct SQL code injection in the netpage;
- Perform dissimilar searches applying search engines to recover personalized information about site: password, username;
- Password cracking applying decryption systems;
- Guessing the website session id (session prediction);
- The web indexing of the site should not reveal security functions like scripts, database address connection, etc;

Cross Site Scripting (or XSS) is one of the greatest mutual application-layer web attacks. XSS commonly aims at scripts embedded in a page which are executed on the client-side (in the user's web browser) rather than on the server-side. XSS as whole is a threat that is brought about by the internet security weaknesses of client-side scripting languages, with HTML and JavaScript as the prime culprits for this exploit. The idea of XSS is to change client-side scripts of a web application to execute in the manner desired by the malicious user. Such an operation can embed a script in a page that then can be completed every time the page is loaded, or whenever an associated event is executed.

An XSS attack can be applied to realize the following malicious results:

- Identity theft;
- Denial of service attacks;
- Web application defacement;
- Accessing sensitive information;
- Altering browser functionality;

To avoid such attacks, an e-learning platform designer can accomplish the following executions:

- Guarantee that the pages in the Web site return user inputs only after authorizing them for any malicious code;
- Do not entirely trust web sites that use HTTPS (Secure Sockets Layer) when it comes to XSS; HTTPS ensures secure connections, but processing of the data entered by the user is internal to the application. If the application has XSS vulnerabilities, the attacker may send a malicious script that can still be performed by the application and lead to XSS intrusions;
- Convert all non-alphanumeric characters to HTML character entities before displaying the user input in search engines and forums;
- Apply testing tools lengthily throughout the design phase to remove such XSS holes in the e-learning application before it goes into function.

SQL injection is a rather humble type of attack, and can be circumvented with strict adherence to some basic coding practices. Applying this approach, a hacker can pass string input to an application with the confidence of gaining unapproved access to a database. Hackers enter SQL queries or characters into the web application to execute an unforeseen action that can then act in a malicious way. Such queries can consequence in entree to unauthorized data, bypassing of authentication, or at the closure of a database even if it exist in on the same web server or on a separate server.

The most common approaches to avoid this type of vulnerability are:

- Inspection of the user's input for dangerous characters like single-quotes;
- Using organized statements, which tell the database precisely what to expect before any user-provided data is passed to it;
- Encryption of sensitive data;
- Ensuring that the return error messages give nothing away about the internal architecture of the application or the database.

The SQL injection can be also used for URLs, which can be altered by an attacker in order to entree important information. To avoid this it is advisable to prevent sending significant parameters in the URL. Sessions are the way of saving the state and user specific variables across subsequent page requests. This is realized by passing a unique and difficult-to-guess identity value (*session id*) to the browser (either in a cookie or the URL) which the browser submits with every new request [9]. The session is active as long as the browser keeps sending the id with every new request. Session Prediction is the same as guessing a valid session id applying several tools and approaches (like brute force technique). The attack is possible when session id is faintly encrypted, Too short or assigned sequentially. Sessions that do not expire on the HTTP server can permit an attacker unlimited time to guess or brute-force a valid authenticated session id and sooner or later permit access to that user's web counts. Additionally, a session id can be

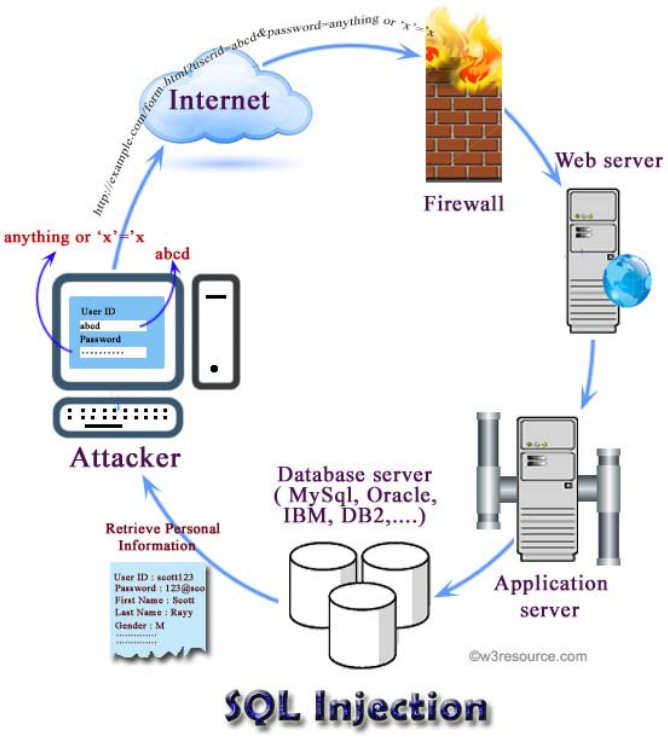

Fig. 3. Demonstration Page for Moodle

possibly logged and cached in proxy servers. When transmitted via an URL parameter, GET requests can potentially be kept in browser history, cache and bookmarks. It can be also simply viewable then.

To avoid issues regarding the session security, the following best practices should be fulfilled:

- Adequately long and unpredictable Session id;
- Validate session id;
- See if the session id has been generated by the application (was not manually introduced by the user);
- Regenerate session id after a period of time or when the user privilege level has changed;
- Apply only cookies to propagate session id;
- Prevent "remember me" option (persistent logins);
- Expire session on security error;
- Expire session after a period of inactivity;
- Erase session cookie when a session is destroyed.

IV. E-LEARNING SECURITY CASE STUDY – MOODLE

There are many open source e-learning systems that can be installed simply and have a wide community of users and developers. One of them is Moodle (current version is 2.7.2 (September,2014), see Figure 4) which is also the mostly common used as well with its competitor Coursera (see Figure 5, both well known platforms.

Below, we will study some security holes of this platform. Moodle (Modular Object-Oriented Dynamic Learning Environment) is a software package which gives the opportunity of establishing Internet-based courses and a good support for se-

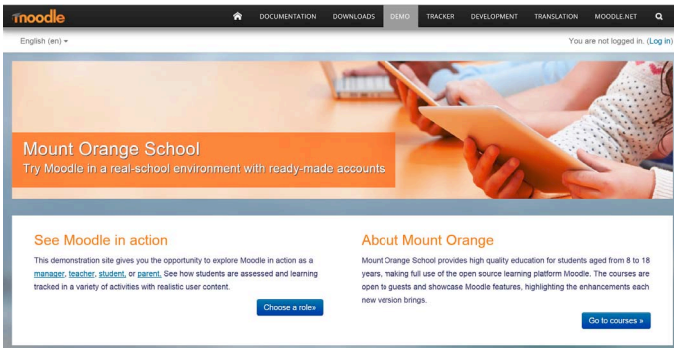

Fig. 4. Demonstration Page for Moodle

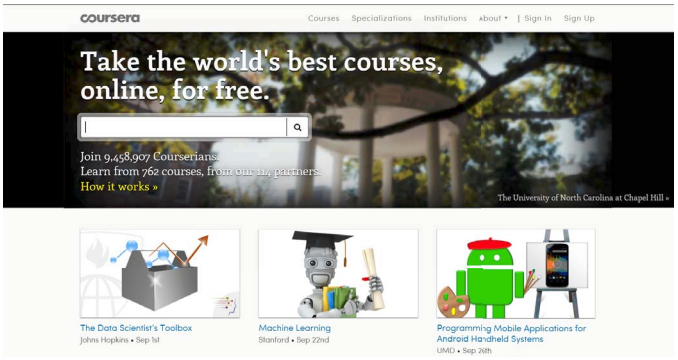

Fig. 5. StartPage for Coursera

curity and administration [3]. The source code is programmed in the PHP language and the their corresponding databases are in MySQL and PostgreeSQL. It is a free web application where instructors can apply this to generate well-organized online learning sites.

It consist of a powerful course management function that includes the creation of lessons, assignments, quizzes, documents and extra. It comes with several modules that help students and instructions to communicate with each other like chat, forum, survey or workshop.

The first design defect of Moodle platform is related to the brute force attack. A brute force attack means to apply every possible code, combination, or password until the right one is observed. This kind of attack may be executed to guess the password or user name. To guess the password, the user sends numerous requests to the web server with the blank cookie field so that the login error count is reset to zero [8]. To guess the user, a number of usernames are sent with an arbitrary password. Typically, if the reply from the server is lengthier, the chances to guess the user are difficult. To avoid this, Moodle added a password policy system (starting from version 1.9) which may be set up from: *Administration - Security - Site policies*. This issue may be resolved also using a captcha system in the login page.

Another security issue may occur when a session hijacking attack is used. Session hijacking is the act of taking control of a user session after successfully gaining or generating an authentication session ID.

Session hijacking involves an attacker applying captured, brute forced or reverse-engineered session IDs to seize control of a legitimate user's Web application session while that session is still in progress. The session is handled in Moodle by using two cookies: MoodleSession and MoodleSessionTest that can be captured because Moodle uses only SSL tunnels on the login service and a few administration services. In that way, the HTTP requests are done in plaintext which may be captured and decoded. Once the cookie is received, an attacker can apply this data in its own HTTP request to take control of the target user session. The only way to avoid this is to use SSL protocol (Secure Socket Layer). The entire site should create SSL connections with its clients (for this the web server needs an SSL certificate).

V. Conclusion

This paper shows several security aspects of e-learning platforms in general and mainly, we examined the most significant concerns of the famous open source learning system, Moodle. The development of the e-learning systems should be created by applying secure functions and internationally recognized standards. The development requires that security services (e.g. authentication), encryption, access control, managing users and their permissions to be implemented. The data transfer between the system and administrators or content operators should be implemented on encrypted SSL channels via the web administration interface. A secure learning platform must integrate all aspects of security and secure mechanism without influencing the system performance a lot.

References

[1] *E-learning Systems*, http://www.bapsis.com/elearningsystems.htm, [Online; accessed 19-September-2014].

[2] Craciunas, S. & Elsek, I, (2009) *The standard model of an e-learning platform*, Bucharest, Romania, (Chapter 2).

[3] Dobre, I., (2010) *Critical Study of the present e-learning systems*, Academia Romana, Romania, (Chapter 2).

[4] Edgar, R. W., (2005) *Security in e-learning*, Springer. Vienna University of Technology, Austria, (Chapter 1).

[5] Iacob, N., (2010). *Data replication in distributed environments*, Proceedings of International Scientific Conference ECO-TREND: Brancusi University Targu Jiu, 629-634.

[6] Jalal, A. & Ahmad, M., (2008). *Security Enhancement for E-Learning Portal*, Proceedinggs of International Journal of Computer Science and Network Security, Department of Computer Science City University, Peshawar, Pakistan, 41-45.

[7] Kritzinger, E. & Solms S., (2006). *E-learning: Incorporating Information Security Governance*, Proceeding of Informing Science and IT Education Conference, Salford (Greater Manchester), England, 319-325.

[8] Kumar, S. & Kamlesh, D., (2011). *Investigation on Security in LMS Moodle*, Proceedinggs of International Journal of Information Technology and Knowledge Management, Kurukshetra University, Kurukshetra, India, 233-238.

[9] Przemek, S. (2007), *PHP Session Security*, Poland, (Chapter 1).

[10] Smeureanu, I. & Isaila, N, *The Knowledge Transfer Through E-Learning in Business Environment*, Economy Informatics, 97-98.