Security research Blackboard Academic Suite

September 22, 2010

Authors:

Michiel Prins

Jobert Abma

Online 24
Zilverlaan 2
9743 RK Groningen
The Netherlands

Tel.: +31 50 711 9220
www.online24.nl

Version
1.0
September 22, 2010

Contents

1 Introduction

Blackboard is the *de facto* standard e-learning web application. Officially called Blackboard Academic Suite and then renamed to Blackboard Learn since version 9.0. It's world wide used by universities to provide course information, content management, student communication, collaboration, assessment management etc.

Early 2010, Dutch security company Online 24 conducted a security research on Blackboard. This document consists of parts of the original security research, which was published to Blackboard and a small group of IT security managers at universities in The Netherlands. The technical details of the vulnerabilities that were found during the research aren't disclosed yet. A full disclosure date will be anounced soon. The problems discussed in this article affect Blackboard 8 Service Pack 6 (8.0.475.0), still a widely used version. Though many of these problems still exist in more recent versions of Blackboard, like 9.0 and 9.1.

Chapter two will cover the different types of vulnerabilities that have been discovered during the security research. The third chapter contains statistical information about the vulnerabilities that have been found in Blackboard. These statistics are represented visually with a collection of charts.

Chapter four analyzes the potential risks of the found vulnerabilities. The risks are classified to the confidentiality, integrity and availability (CIA) triad model.

The final chapter features a conclusion.

2 Vulnerabilities

During the research 84 different vulnerabilities were found. These vulnerabilities can be categorized in eight vulnerability types. Each type of vulnerability is explained and examples of exploitation of these vulnerabilities are given.

The categories are categorized to the WSAC Threat Classification[1] description. This description describes web application attacks and weaknesses.

2.1 Cross-site scripting

This type of attack allows an attacker to inject malicious code into a user's web browser. For example, it can be used to hijack a user's session or infect a visitor with a trojan horse using vulnerabilities in the user's web browser.

Users of Blackboard are put at a serious risk. During the research 63 different cross-site scripting (XSS) vulnerabilities were found. All of these vulnerabilities could be exploited to hijack a user's session or even steal his/her login credentials.

During the research two different types of cross-site scripting vulnerabilities were found: persistent XSS and non-persistent XSS. Persistent XSS means that the XSS vulnerabilities will persist after the request is submitted (e.g. it's permanently stored inside Blackboard). A non-persistent XSS vulnerability always needs special interaction between the user and Blackboard for successful exploitation and will not be stored anywhere. These non-persistent XSS vulnerabilities are not specifically less dangerous.

Imagine this situation:
"The attacker exploits a (persistent) XSS vulnerability to redirect the victim to a genuine looking Blackboard login screen. Once the user opens the infected page, he/she is being confronted with the fake login screen. The user logs in and the attacker intercepts the username and password and displays a message telling the user the login attempt failed. The user will think he or she made a typo and takes a second attempt, which will succeed."

2.2 Insufficient authorization

Insufficient authorization (IA) is a type of vulnerability, which occurs when a web application does not perform adequate checks if the user

[1] `http://projects.webappsec.org/Threat-Classification`

is performing an action or accessing data the user is actually allowed to do, or to access.

Insufficient authorization is the second most common type of vulnerability discovered during the security research on Blackboard, after cross-site scripting. In Blackboard 11 vulnerabilities of this type have been found.

During the research vulnerabilities were found which could enable attackers to read, modify or delete every Blackboard user's personal data (i.e. calendar items, preferences and address book items).

2.3 Information leakage

Leakage of information (IL) occurs when a web application makes sensitive data visible to potential attackers. For example, it's called an information leak when a web application leaks technical details of the web application itself or the environment it's running on. In Blackboard a total of four information leakage vulnerabilities have been found.

For example, if an error occurs, Blackboard hides a full Java stacktrace in a HTML comment in the page. This stacktrace contains useful information about the framework that is used and how user input is handled.

2.4 Mail command injection

Exploitation of this vulnerability is possible when a user is able to influence the communication between mail servers and a web application. Mostly this vulnerability is in forms that are used to send e-mails. An attacker can modify such a form in a way that it's possible to inject headers into the e-mail that's being sent. During the research, two mail command injection (MCI) vulnerabilities have been discovered in Blackboard.

For example, it allows an attacker to send spam to external domains, using the mail server assigned for use with Blackboard. What's even worse is the fact that an attacker could send mail to the internal domains which can be very harmful to teachers or co-students. When a student uses this technique to fake the sender ID, the student can operate on behalf of a instuctor.

2.5 Abuse of functionality

This type of vulnerability lets a user abuse an application's own functionality to attack others or itself. It allows attackers to use the application's

functionality for other purposes than it was originally intended. This can lead to an attack on other systems, which owner's will think the abused web application performed the attack. The actual attacker remains invisible. In Blackboard two abuse of functionality (AoF) vulnerabilities were found.

For example, during the research a vulnerability was found, which gives all users of Blackboard the ability to bypass the JavaScript filters in text fields. This is due to the fact that JavaScript filtering is done on the client side with the WYSIWYG editor. This editor can be turned off at the touch of a button, bypassing the filters. Bypassing these filters attackers are allowed to perform XSS attacks.

2.6 Local file inclusion

A local file inclusion (LFI) is an exploit technique that allows an attacker access files by modifying user input. Using this technique the attacker is able to obtain sensitive information about Blackboard and even about the underlying operating system. During the research one LFI vulnerability was found.

Imagine this situation:
"An attacker is able to gain access to files outside Blackboard's document directory. The attacker includes Blackboard's access logs. These logs can be influenced by the attacker, so exploitation would lead to a new highly dangerous vulnerability that allows the attacker to execute custom commands on the Blackboard server."

2.7 Improper filesystem permissions

Vulnerabilities of this kind are mostly extremely dangerous, because it's a threat against confidentially, integrity and availability of Blackboard. The problem occurs when the executing daemon (i.e. Apache) has got permissions to files, folders and symlinks, which it should not have. Improper filesystem permission (IFP) vulnerabilities allow attackers to delete, overwrite or read files on the filesystem.

In Blackboard one vulnerability of this kind was found. This vulnerability enables the attacker to delete every single file on the disk. This means that the attacker could delete critical configuration files, which could result in a total crash. When other servers, such as mail servers, file servers or domain controllers, depend on Blackboard, it is even possible to crash multiple systems.

2.8 Cross-site request forgery

A cross-site request forgery (CSRF) is an attack that involves in sending an HTTP request to a target destination without the knowledge of the victim. This technique is mostly exploited in a way that lets an attacker submit a web form using the authenticated session of a victim. In Blackboard every form is vulnerable to this type of attack.

Imagine this situation:
"A course instructor is being triggered to click a link provided by one of his/her students. The instructor follows the link. Behind the scenes, invisible to the victim, the victim submitted a form to change the student's course role to "instructor". Voila, the student is now a course instructor, with serious consequences."

3 Statistics

This chapter summarizes the findings of the security research in a few graphical representations. All the single vulnerabilities, which aren't disclosed in this document, have a classification assigned. This classification depends on the seriousness of the vulnerability.

3.1 Vulnerabilities

The chart below shows the various types of vulnerabilities. The most common type, as seen in this chart, is the cross-site scripting vulnerability, followed by the insufficient authorization vulnerability.

The other vulnerabilities found in Blackboard, are not as common as the two mentioned above. However, this does not mean that the vulnerabilities are less important. The classification of the found vulnerabilities is taken out of the question.

The cross-site request forgery vulnerability currently has one occurrence in this chart. However, the vulnerability itself occurs multiple times within Blackboard. It is listed as one occurrence, because the solution applies to all occurrences.

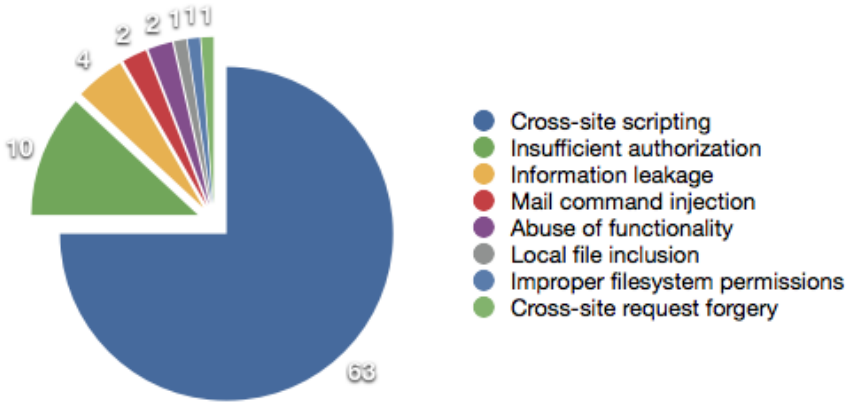

Figure 1: Type of vulnerabilities

3.2 Classification

The chart below displays the classifications of the various security issues found in Blackboard. The biggest part of the discovered vulnerabilities, has received the classification high.

A close second, is the low classification. The third most common classification is medium, which takes up to 23 percent of the vulnerabilities. The fourth and final classification is very low, which only claims two percent of all vulnerabilities.

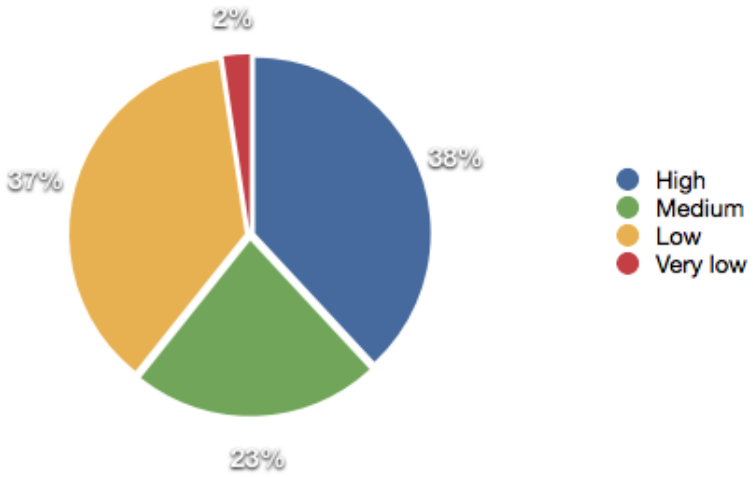

Figure 2: Classifications

3.3 Classification by type

The chart at the end of this paragraph displays the different types of vulnerabilities and the classifications given to them.

This chart shows that most of the vulnerabilities classified as low and very low are cross-site scripting vulnerabilities.

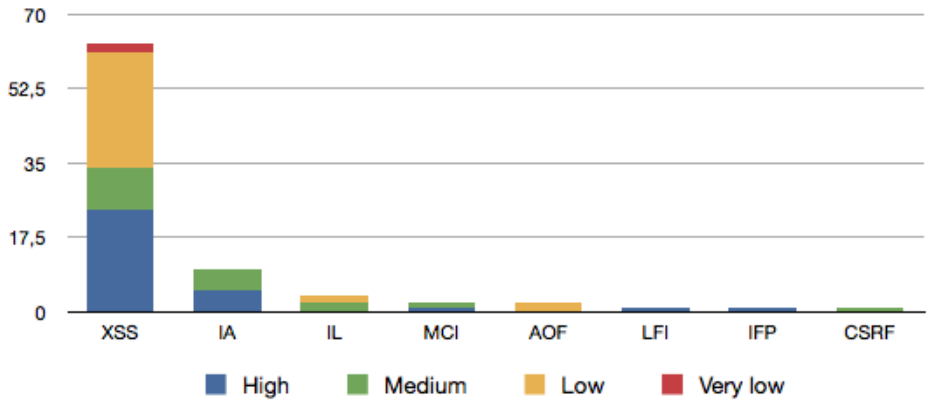

Figure 3: Classification per type

4 Risk analysis

This chapter discusses the potential risks of using Blackboard.

4.1 Risks

The types of vulnerabilities are rated according to the confidentiality, integrity and availability triad. These aspects are rated on a scale of one to three. Higher scores indicate higher risks.

4.1.1 Cross-site scripting

Due to the vast amount of cross-site scripting vulnerabilities found in Blackboard, the confidentiality of the data is endangered. This type of vulnerability can provide users with the ability to, for example, steal user authentication data such as usernames and passwords. The vulnerability can also be used in combination with cross-site request forgery to obtain advanced permissions or roles. Therefore, Blackboard is a vulnerable platform to use, when it comes to critical information.

Confidentiality	This vulnerability gives users the ability to steal login information and therefore a lot of data, related to the specific user (3).
Integrity	Combining this vulnerability with cross-site request forgery can lead to malformation of data (2).
Availability	Cross-site scripting is mostly harmless to the availability of the application (1).

4.1.2 Insufficient authorization

The insufficient authorization vulnerabilities found in Blackboard, are mostly related to personal data. This vulnerability is a threat to the confidentiality of the data stored in Blackboard. It allows users to access material they originally were not allowed to access, such as personal address books and calendars. While these vulnerabilities are dangerous when viewing is allowed, they are even more dangerous when users are able to modify the data.

Confidentiality	Personal information from all other users, such as tasks and address book items, can be read (3).
Integrity	In addition to reading, the information mentioned above can be modified or removed (3).
Availability	Insufficient authorization vulnerabilities generally don't have an effect on the availability of the application (1).

4.1.3 Information leakage

Information leakage can be considered a risk of the use of Blackboard as it displays critical information that comes in handy when performing a different type of attack. The confidentiality of the data is compromised by this type of vulnerabilities, as it displays critical information about components related to Blackboard. The vulnerabilities found, display information about the database, the network infrastructure and stack traces of program code.

Confidentiality	This type of vulnerability displays critical information about the network infrastructure and the database (2).
Integrity	The found vulnerabilities cannot modify data (1).
Availability	Information leakage itself has no effect on the availability (1).
Notes	The information retrieved from these types of vulnerabilities can be used to perform other attacks, which could inflict damage to the integrity and availability of the application.

4.1.4 Mail command injection

The mail command injection vulnerabilities allow users to send e-mails to multiple receivers, using Blackboard's mail server. While this in itself

may not be enough to consider it a vulnerability worthy of the high classification, it's effectiveness should not be underestimated. When a user performs a mail command injection, the sending end, in this case Blackboard's mail server, can be held responsible for sending spam. The trusted domain is damaged by these actions and may appear on a blacklist. This vulnerability can become highly dangerous, when infected attachments are sent using the mail command injection.

Confidentiality	It isn't possible to access confidential information with this vulnerability (1).
Integrity	This vulnerability allows users to send e-mails with a different sender label (2). The integrity of the sender label in e-mails is therefore unreliable.
Availability	Mail command injection has no effect on the availability (1).

4.1.5 Abuse of functionality

With abuse of functionality, features of Blackboard are used for a different purpose than originally intended. While they are technically speaking not considered leaks, they are effective for hijacking, hacking or abusing Blackboard.

One of the found vulnerabilities gives all users of Blackboard the ability to bypass the JavaScript filters in text fields. This is due to the fact that JavaScript filtering is done on the client side with the WYSIWYG editor. This editor can be turned off at the touch of a button, bypassing the filters.

Confidentiality	This vulnerability doesn't affect the confidentiality (1).
Integrity	Abuse of functionality vulnerabilities don't impose a threat for integrity (1).
Availability	The availability of the application stays intact with or without this vulnerability (1).
Notes	Abuse of functionality can be used to perform other types of attacks.

4.1.6 Local file inclusion

The local file inclusion vulnerability found in Blackboard allows an instructor to read files stored on the hard disk, on which Blackboard is running. The local filesystem is therefore exposed through the use of Blackboard, making it accessible via the web. The confidentiality of the local

files has been damaged by this vulnerability.

Confidentiality	This vulnerability allows instructors to read files from the local hard disk (2).
Integrity	This local file inclusion doesn't have the ability to modify files. The integrity of the files remains intact (1).
Availability	Reading the local files through a local file inclusion has no effect on availability (1).
Notes	The information found with local file inclusion, can be used to perform a different type of attack.

4.1.7 Improper filesystem permissions

Improper filesystem permissions allow access to files which were not intended to be accessed/modified. One improper filesystem permission vulnerability was found in Blackboard. It allows an instructor to delete a file from the local file system. With precision, Blackboard can be taken offline when removing critical files from the filesystem. It therefore compromises the availability of the application.

Confidentiality	Files that were invisible for the instructors, became visible with this vulnerability (2).
Integrity	With this vulnerability, files can be removed (2).
Availability	An instructor can inflict damage to the system by removing important files from the filesystem, endangering the availability (3).

4.1.8 Cross-site request forgery

One of the bigger vulnerabilities of Blackboard, is the cross-site request forgery vulnerability. It allows users to perform actions by simply accessing a URL. When a person is tricked into clicking a malicious link, for example, it can be used to gain instructor rights or advanced permissions.

Confidentiality	Cross-site request forgery by itself is not able to reveal confidential data (1).
Integrity	The vulnerability allows a user to trick an instructor in giving his or her permissions, resulting in malformed data (3).
Availability	Cross-site request forgery generally has no effect on availability (1).
Notes	This vulnerability can be used to attain a different user role or advanced permissions, changing all of the three aspects above to high (3).

4.2 Analysis statistics

The risks of the usage of Blackboard are displayed in the chart at the bottom of this paragraph. The chart is based on the CIA triad ratings of each type of vulnerability. The two largest vulnerabilities in Blackboard are insufficient authorization and improper filesystem permissions. The insufficient authorization vulnerability, allows all users to access and edit personal modules, such as address books and calendars.

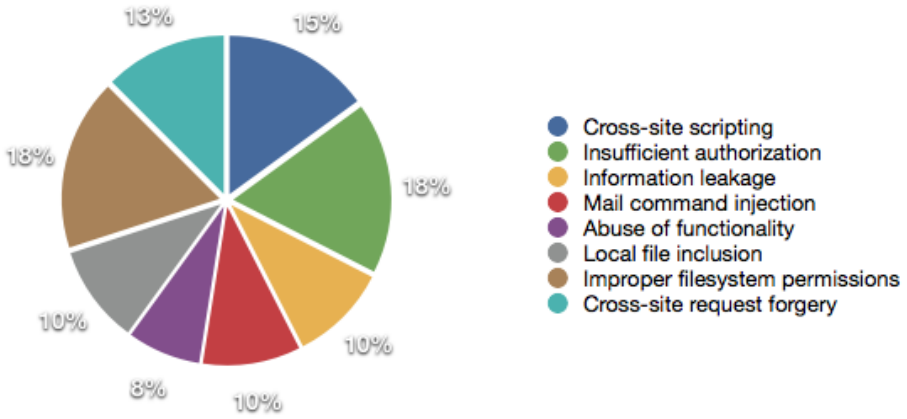

Figure 4: The risk of the usage of Blackboard

5 Conclusion

The most common vulnerability in Blackboard is cross-site scripting. These XSS vulnerabilities are not all as dangerous as the other. Most of the cross-site scripting issues are classified as low. But even most of the vulnerabilities classified as low could be exploited with CSRF techniques in a way that can be very dangerous for all users on Blackboard.

Though the other discovered security issues which are not as common as the cross-site scripting vulnerability, should not be underestimated. For example, insufficient authorization could damage the integrity and reliability of Blackboard. Improper filesystem permissions also affect the integrity and reliability but could damage the availability as well.

Starting with the cross-site scripting vulnerabilities, it is very helpful to keep in mind that every user input is potentially dangerous. During the research on Blackboard the user input filtering attracted attention. Most of the user input is filtered, but can be bypassed in a few simple steps. Because of the seriousness of the vulnerabilities which are found during the research, it is not recommendable to use Blackboard as the main e-learning application. When using Blackboard, the user's privacy can't be guaranteed and the confidentiality and integrity are put at risk.