Oct 28th, 4:00 PM - 5:30 PM

Security Assessment of Web Applications

Renelada Kushe
Polytechnic University of Tirana, rkushe@fti.edu.al

Security Assessment of Web Applications

Renelada Kushe

Faculty of Information Technology
Politechnic University of Tirana, Albania
rkushe@fti.edu.al

Abstract.A web application is an application that is accessed by users over a network such as the internet or intranet. The term also refers an application that is coded in a browser-supported programming language and reliant on a common web browser to render the application executable. Web applications are vulnerable to varies exploits from those which manipulate the application via its graphical web interface (HTTP exploits), to tampering the Uniform Resource Identifier (URI) or tampering HTTPS elements not contained in the URI. Getting started from the accessibility and the variety of exploits, the security assessment is a necessity for providing protected data and secure navigation through the web application. In this paper we will present a case study for security assessment of a web application and also will insert our scrip to a web application as an example of a cross site scripting exploitation.

Keywords: Security, Web application, Exploit, Cross site

Introduction

Hacking a web application refers to carrying out unauthorized access of a website or website details. A web application is an application that is accessed by users over a network such as the internet or intranet. The term may also mean a computer software application that is coded in a browser-supported programming language and reliant on a common web browser to render the application executable. Web hacking refers to exploitation of application via HTTP which can be done by manipulating the application via its graphical web interface, tampering the Uniform Resource Identifier (URI) or tampering HTTPS elements not contained in the URI.
In this paper we will present a case study for security assessment of a web application and also will insert our scrip to a web application as an example of a cross site scripting exploitation.
In section two, are described vulnerabilities of a web application such as Remote code execution, SQL injection for the web application database, Format String vulnerabilities, Cross Site Scripting (XSS) and Username enumeration. In section three are presented experimental setup and results. In this section, are described two real life experiments; first we have presented the security assessment for the Faculty of Information Technology web page (fti.edu.al domain), by utilizing Acunetix tool and secondly we have implemented a Cross Site Scripting (XSS) exploit over the bWAPP framework. In the last section are given some conclusions and future work.

Vulnerabilities of a web application

A web application is always vulnerable against hackers attack and every day there are different new way how to hack a web application and it is harder to find the right defense.

The 5 most vulnerabilities of a web application are [1]:

1. Remote code execution
2. SQL injection
3. Format string vulnerabilities
4. Cross Site Scripting (XSS)
5. Username enumeration

These vulnerabilities will be described briefly in the following paragraphs.

Remote code execution

As the name suggests, this vulnerability allows an attacker to run arbitrary, system level code on the vulnerable server and retrieve any desired information contained therein. Improper coding errors lead to this vulnerability.

At times, it is difficult to discover this vulnerability during penetration testing assignments but such problems are often revealed while doing a source code review. However, when testing Web applications is important to remember that exploitation of this vulnerability can lead to total system compromise with the same rights as the Web server itself.

SQL injection

SQL injection is a very old approach but it's still popular among attackers. This technique allows an attacker to retrieve crucial information from a Web server's database. Depending on the application's security measures, the impact of this attack can vary from basic information disclosure to remote code execution and total system compromise.

Format string vulnerabilities

This vulnerability results from the use of unfiltered user input as the format string parameter in certain Perl or C functions that perform formatting, such as C's printf().
A malicious user may use the %s and %x format tokens, among others, to print data from the stack or possibly other locations in memory. One may also write arbitrary data to arbitrary locations using the %n format token, which commands printf() and similar functions to write back the number of bytes formatted. This is assuming that the corresponding argument exists and is of type int* .
Format string vulnerability attacks fall into three general categories: denial of service, reading and writing.

Cross Site Scripting (XSS)

The success of this attack requires the victim to execute a malicious URL which may be crafted in such a manner to appear to be legitimate at first look. When visiting such a crafted URL, an

attacker can effectively execute something malicious in the victim's browser. Some malicious Javascript, for example, will be run in the context of the web site which possesses the XSS bug.

Username enumeration

Username enumeration is a type of attack where the backend validation script tells the attacker if the supplied username is correct or not. Exploiting this vulnerability helps the attacker to experiment with different usernames and determine valid ones with the help of these different error messages.

Experiment Setup and Results

There are performed two real life experiments; In the first paragraph we will present the security assessment for the Faculty of Information Technology web page (fti.edu.al domain), by utilizing Acunetix tool [2]; and in the second paragraph we will present the implementation of a Cross Site Scripting (XSS) exploit by inserting our script to a web application over the bWAPP framework [3].

Security assessment

In this paper, we have performed and generated a security assessment report for the Faculty of Information Technology web page. This process generally is realized by utilizing tools such as web vulnerabilities scanners. In this experiment we have configured the Acunetix tool. Acunetix automatically tests websites and web applications for SQL Injection, XSS, XXE, SSRF, Host Header Attacks & over 3000 other web application vulnerabilities. In addition, Acunetix provides powerful Vulnerability Management tools for ensuring vulnerabilities are not only discovered, but remediated in context of business-criticality; as well as providing management with the tools and reports required to make strategic decisions.

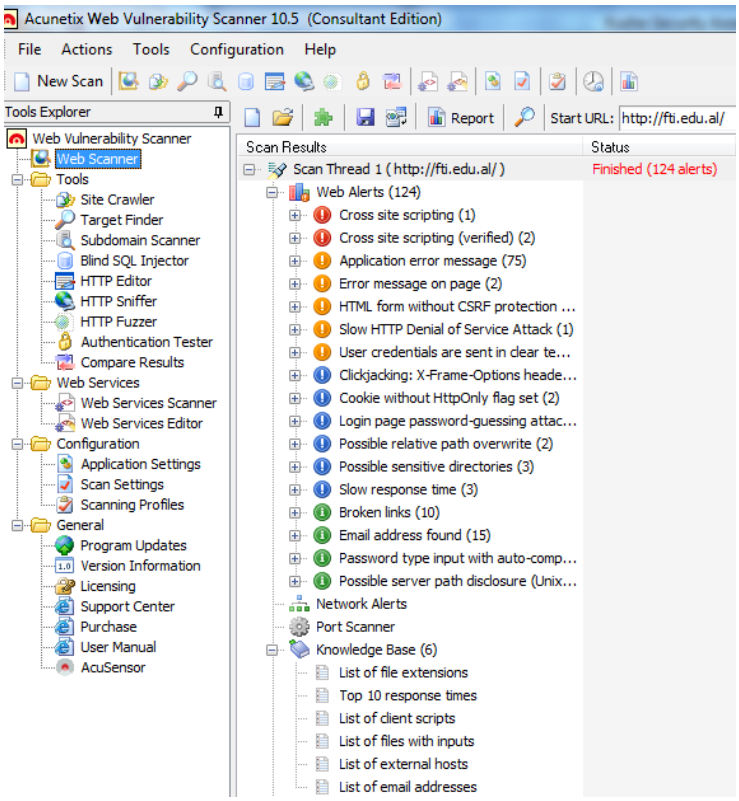

Fig. 1. The main menu of Acunetix and scan results

In Figure 1, is shown the main menu of Acunetix vulnerability scanning tool, for configuring the features for the vulnerability scanning process; also a detailed list of scanning results for fti.edu.al. A summary of alerts and also the ranking of threat level are given in a report form for generating statistics and making analyses. As shown on Figure 2, the threat level for our target is ranked Level 3: High. This ranking is due to three potential XSS vulnerabilities.

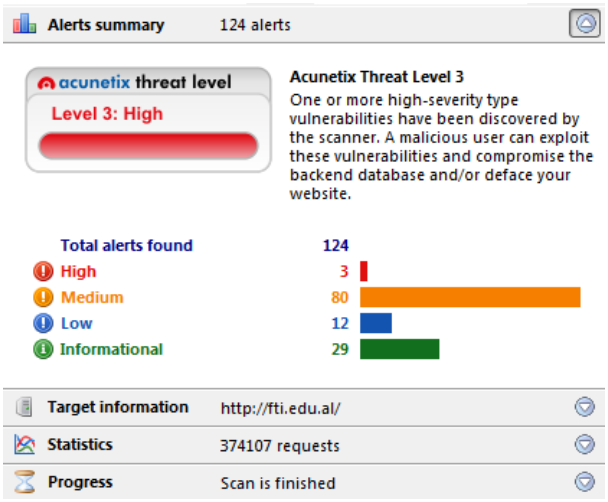

Fig. 2. Alert summary and threat level

Cross Site Scripting (XSS) exploitation

For implementing Cross Site Scripting exploit, we have utilized the bWAPP framework. The bWAPP is a free and open source deliberately insecure web application in PHP that uses a MySQL database. It can be hosted on Linux/Windows with Apache/IIS and MySQL. It is supported on WAMP or XAMPP. In this experiment, we have configured the bee-box, a custom VM customized with bWAPP. The bWAPP framework is created for security testing and educational purposes only.
First we have to create an account in bWAPP, and then may perform exploits. The portal of bWAPP, offers a wide range of expoits to implement, but in our experiment we are focused on XSS. In Figure 3, is described the script inserted to the XSS exploit. `<script>alert("JENI SULMUAR")</script>`. After the execution, a pop up window will be generated, as shown in Figure 4.

Conclusions

Scanning the web for vulnerabilities is a necessity. A web application should always be secure to protect the client's data and offer security among navigation.
Acunetixis a powerful vulnerability scanning tool which automatically tests websites and web applications for SQL Injection, XSS, XXE, SSRF, Host Header Attacks & over 3000 other web application vulnerabilities. The scanning target in this paper fti.edu.al has a threat level 3, due to three potential XSS vulnerabilities.
The XSS exploitation is easy to implement and that is the reason why web applications suffer mostly from this type of exploit.

Fig. 3. The script inserted to XSS exploit

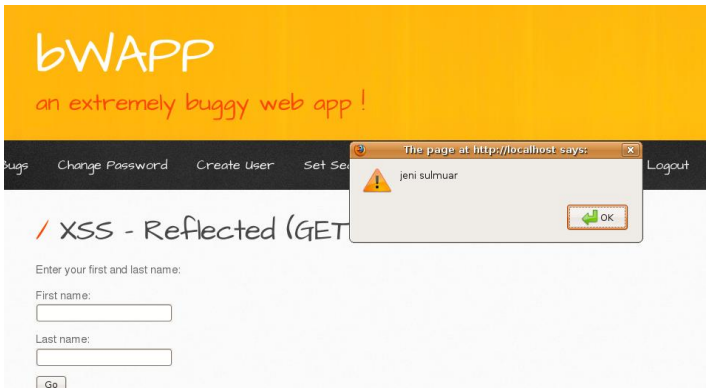

Fig. 4. The result of the exploit

References

1. Siddharth, S., Doshi P.: Five common Web application vulnerabilities,Available: https://www.symantec.com/connect/articles/five-common-web-application-vulnerabilities
2. Acunetix. “Website Security with Acunetix Web Vulnerability Scanner,” Available: http:// www.acunetix.com/ , January 2014.
3. bWAPP documentation, Available: http://www.itsecgames.com/