Introduction to Web Security and Evaluation Methods of Web Application Vulnerabilities

Petra Holbíková, Roman Jasek
Department of Artificial Intelligence and Informatics
Faculty of Applied Informatics, Tomas Bata University in Zlin
Zlin, Czech Republic
e-mail: holbikova@fai.utb.cz, jasek@fai.utb.cz

***Abstract*—In this paper, we focus on basic security rules of Web applications or Websites, as well as recommendations for developers in terms of what should be avoided while creating Web applications. The paper is divided into two parts. In the first part we describe the basic security rules and common Web risks. In the second part, a system for risk assessment of vulnerability - Common Vulnerabilities Score System is introduced and described.**

Keywords - Web Security; Secure Socket Layer; Threat Risk Modelling; Common Vulnerabilities Score System; Basic Web Security; Security Risks.

I. INTRODUCTION

There are basic principles to secure Web sites and Web applications that are recommended to always be used. The effort to minimize the vulnerability of Web applications is already evident in the initial design. Integrating several programming languages into one project potentially increases the risk of a security error. Some of the possible impacts include theft and misuse of users personal data, financial repercussions on the company operating the Web applications, reputation harm or technical consequences - destruction of the entire application, or its theft [11]. A list of common potential hazards and errors is generally considered as the minimum of what developers have to test while developing their applications [7].

The essential set of recommendations will be presented in the following sections. In Section 2 we describe the basic security rules, like types and usage of Secure Socket Layer certificate [7], why it is advisable to regularly update the system, common mistakes at form validation and passwords creation. The last part of Section 2 is dedicated to top 10 risks in Web applications. In Section 3 we describe Common Vulnerability Scoring System and its Base metrics scoring system [10].

II. BASIC SECURITY RULES

Secure Socket Layer (SSL) certificate is one of the most common and basic ways to secure Websites. It is used for secure communication and data transfer between Web browsers and Web servers. SSL is mostly used in situations where, for example, sensitive data is entered by the user in a login form and includes username and password. Other examples of SSL use include cases of stored sensitive information in user accounts such as credit card numbers, emails, address and passwords, or communications and trusted information exchanges. To provide secure connection, an SSL certificate is installed on the Web server. The SSL certificate has two functions. The first function is to authenticate the identity of the website and the second is to encrypt transmitted data.

A. SSL Certificates

Choosing a trusted certification authority is an important part of an SSL certificate selection. The most used certification authority is probably Symantec, due to its cooperation with Verisign, GeoTrust, Thawte and low-cost Rapid SSL authorities. SSL certificates can be divided into two groups. The certificates in the first group are based on the number of owned domains or subdomains [9].

- Single Certificate protects communications between the server and a Web browser only for one domain or subdomain
- Wildcard certificate can be used for one domain and unlimited number of its subdomains
- Multi-Domain certificate is used for secure multiple domain names

The second type of certificates is based on the level of validation.

- Domain Validation certificate covers basic encryption and verification of the ownership of the domain name registration.
- Organization Validation certificate also authenticates additional details of the owner.
- Extended Validation (EV) certificate provides the highest degree of security. The user is informed within the Web browser by the icon of padlock on the left side of uniform resource locator (URL) and a green address bar. The most frequent usage of this type of certification is in the financial and banking sector.

The last type of SSL certificate, which is also often used, is a Code Signing Certificate. This certificate allows developers and software companies to sign their applications that are intended for distribution over the Internet [8].

Copyright (c) IARIA, 2016. ISBN: 978-1-61208-493-0

B. Regular system updates

Although some safety information is visible to the user, other safety information is better to be hidden from the user or potential attackers, especially if well-known and widely available open source Content Management System (CMS) or system is used.

It is a good idea to hide information, such as what kind of system type, what version, or information on bug reports is used. If a potential attacker finds out what system is used and its version, it is much easier to lead a targeted attack on a website. Knowledge of the system version can be used to detect information about weaknesses in various databases or existing exploit program. A common defense method against the exploit attacks is to perform regular updates.

Detailed error information should not be provided to the users themselves. A typical situation is when users log into their account. It is not advisable to show the user what part (username or password) was wrongly filled out. If the potential attacker sees what information in a login form is correctly filled, the attacker can then focus on the part that is filled wrong.

C. Form validation, file upload and passwords

Other weak points of Web applications are any input fields for users to type in information, weak user passwords or file uploads on the frontend.

Form validation is recommended on both sides of the Web application. The Web browser can validate filled-out data by JavaScript, as well as on the server side filled-out data are validated in the programing language in which the application is programmed. The Web browser can seize small errors and limit user in using some types of characters. These initial validations, however, can be overcome. After that, the server captures and filters these surpassed validations. On the server side, the inserted code is also removed. Otherwise, the inserted code can damage databases.

File Upload Form is a simple way for an attacker to upload harmful files on the server. The basic rules are limit file size and multipurpose internet mail extensions (MIME) types. Thereafter, it is appropriate to store files outside the document root and rename files during-their uploading to a Web server. At the same time, users who are not logged in should not be allowed to upload files.

In the password case, the user should be informed how a strong password looks like. The password length is generally recommended to be at least 8 characters. Another rule for a password creation is to combine characters. It is not recommended to use any common words. These passwords can be broken by brute force attack where attackers use dictionaries - Dictionary attack.

The longer and more complicated the password, the safer it is.

It is appropriate to choose a combination of uppercase and lowercase letters, numbers and characters. It is advised to use those characters that can be found any language keyboard. All these rules are possible to be set up and filtered in the Web browser and then passed to the user.

Another recommendation, which cannot be part of the user limitation, is not to use any personal information in a password. Also, it is not recommended to use your name or the names of your loved ones, nicknames or date of birth, ID card numbers or any words that can be easily connected with a user. The user should use a different password for each service, or at least modify it. Passwords should be changed regularly [7].

D. OWASP Top 10

OWASP is an official name for Open Web Application Security [6]. Project OWASP Top 10 analyses Web risks at three-year intervals and monitors the changing trends in Web applications vulnerability. These risks are frequently only a fraction of developer testing [6]. The summary of common security risks is shown in Table 1 and Table 2. Table 1 shows the most common security risks of years 2004 and 2007.

TABLE I. OWASP TOP 10 2004, 2007 COMPARISON [6]

	2004	2007
1	Unvalidated Input	Cross Site Scripting
2	Broken Access Control	Injection Flaws
3	Broken Authentication and Session Management	Malicious File Execution
4	Cross Site Scripting	Insecure Direct Object References
5	Buffer Overflow	Cross Site Request Forgery
6	Security Misconfiguration	Information Leakage and Improper Error Handling
7	Improper Error Handling	Broken Authentication and Session Management
8	Insecure Storage	Insecure Cryptographic Storage
9	Application Denial of Service	Insecure Communications
10	Insecure Configuration Management	Failure to Restrict URL Access

Table 2 shows the continuation of OWASP Top 10 research of security risks for years 2010 and 2013.

TABLE II. OWASP TOP 10 2010, 2013 COMPARISON [6]

	2010	2013
1	Injection	Injection
2	Cross Site Scripting	Broken Authentication and Session Management
3	Broken Authentication and Session Management	Cross-Site Scripting
4	Insecure Direct Object References	Insecure Direct Object References
5	Cross Site Request Forgery	Security Misconfiguration
6	Injection Flaws	Sensitive Data Exposure
7	Insecure Cryptographic Storage	Missing Function Level Access Control

Copyright (c) IARIA, 2016. ISBN: 978-1-61208-493-0

8	Failure to Restrict URL Access	Cross-Site Request Forgery
9	Insufficient Transport Layer Protection	Using Components with Known Vulnerabilities
10	Unvalid Redirects and Forwards	Unvalidated Redirects and Forwards

In general terms, it can be said that the security risks trends have not particularly changed.

III. THREAT RISK MODELING

There are several methods for vulnerability and security assessment of the site. The best known methods are STRIDE and DREAD used by Microsoft [11]. Their names are derived from the initial letters of the evaluated categories.

Another extended evaluation methodology is AS / NZS 4360: 2004 Risk Management that became the first formal standard for documenting and managing risks [5].

The US Department of Homeland Security (DHS) has introduced a group of National Infrastructure Advisory Council Vulnerability Disclosure Working Group which works with the outputs from Cisco Systems, Symantec, ISS, Qualys, Microsoft, CERT / CC, and eBay. One of the outcomes of this group is the system used for assessing the vulnerability of Web applications, Common Vulnerability Scoring System (CVSS) [2][3].

The first version of this system was established in February 2005 with the aim of creating an open and standardized evaluation of the degree of severity of software vulnerabilities. Subsequent development standards continued until the current version (April 10, 2016) CVSSv3.0 introduced in 2015.

The basic score is computed using six metrics that can be divided into two subgroups. The first subset is Exploitability.

- **Attack vector (AV)** is a metric system that asks from which source might be the attack led. Four options are evaluated - network, adjacent network, local or physical.
- **Access Complexity (AC)**, metric system is figuring out how easy or difficult it is to use the detected error. The options are high or low.
- **Privileges Required (PR),** metric system describes the level of privileges that an attacker must have to be able to successfully exploit errors. The values are none, low or high. The highest rating has the value none.
- **User Interaction (UI)** metric system determines whether the vulnerability can be exploited only by an attacker or if a user different from the attacker, is involved. The best score are obtained by Web sites where there is no user interaction.

The second subgroup, called Impact metrics, includes an assessment of Confidentiality (C), Integrity (I) and Availability (A) impacts. We ask whether there is a data destruction, irreparably damaged data or unavailability of data or service. For the evaluation, three evaluation levels of severity are used.

The last component of the basic evaluation is a metric Scope. It indicates if the error affects only the funds managed by the same authority or not [1].

The Base Score is a function of the Impact and Exploitability sub score equations, where the Base Score is defined as,

If (Impact sub score =< 0) 0 else,
Scope Unchanged Round up (Minimum [(Impact + Exploitability),10])
Scope Changed Round up (Minimum [1.08 × (Impact + Exploitability),10])

and the Impact sub score (ISC) is defined as,

Scope Unchanged $6.42 \times ISC_{Base}$
Scope Changed $7.52 \times [ISC_{Base} - 0.029] - 3.25 \times [ISC_{Base} - 0.02]^{15}$

where,

$$ISC_{Base} = 1 - [(1-C) \times (1-I) \times (1-A)] \quad (1)$$

And the Exploitability sub score is,

$$Exploitability = 8.22 \times AV \times AC \times PR \times UI \quad (2)$$

AV ..... Attack Vector
AV..... Attack Complexity
PR ..... Privileges Required
UI ..... User Interaction
C..... Confidentiality Impact
I..... Integrity Impact
A ..... Availability Impact

The calculated final Base Score is evaluated form 0 to 10. The lower the score, the smaller the vulnerability rate. Scoring can be converted into a verbal evaluation, as shown in Table 3.

TABLE III. QUALITATIVE SEVERITY RATING SCALE [5]

Rating	CVSS Score
None	0.0
Low	0.1 – 3.9
Medium	4.0 – 6.9
High	7.0 – 8.9
Critical	9.0-10.0

Other metrics can be used for more comprehensive evaluation of the vulnerability of Web applications. They are classified into two groups. Temporal Metrics measure the current state of the use of techniques, availability of the

code, the existence of patches or workaround that may vary over time. This metrics system has three parts – Exploit Code Maturity (E), Remediation Level (RL) and Report Confidence (RC). Environmental Metrics group describes the impact of vulnerability. This metrics system customizes the CVSS score depending on the importance of the affected information technology asset to organization, measured in terms of Confidentiality (CR), Integrity (IR) and Availability (AR).These two systems are shown in Fig. 1.

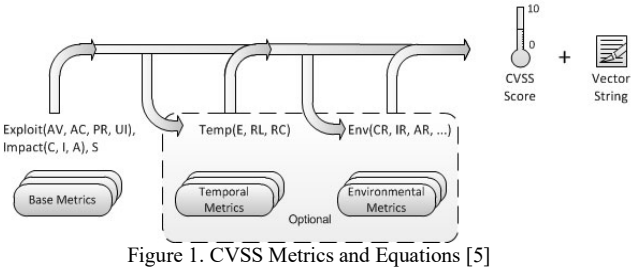

Figure 1. CVSS Metrics and Equations [5]

These two additional metrics are used for clarification of the score system and are optional. The basic metrics is sufficient for security risks assessment of most Web sites [4].

IV. CONCLUSION

Web security is one of the basic issues while creating Web applications. This article outlines the basic rules of Web application security and constitutes one of the methods of assessing Web applications security. The first section summarizes the applicable rules for Web applications creation, like SSL certificates, creating passwords and input forms. The second part describes the method Common Vulnerability Scoring System, Base Metric Group, due to which we can obtain basic CVSS score of Web application. This work is the basis for further development of this method and its application on fuzzy logic.

REFERENCES

[1] H. Li, R. Xi and L. Zhao "Study on the distribution of CVSS environmental score" ICEIEC 2015 - Proceedings of 2015 IEEE 5th International Conference on Electronics Information and Emergency Communication, art. no. 7284502, 2015, pp. 122-125.

[2] H. Holm and K. K. Afridi "An expert-based investigation of the Common Vulnerability Scoring Systém", Computers and Security, 53, 2015, pp. 18-30.

[3] G. Spanos and L. Angelis "Impact Metrics of Security Vulnerabilities: Analysis and Weighing", Information Security Journal, 24 (1-3), 2015, pp. 57-71.

[4] I. V. Anikin "Information security risk assessment and management method in computer networks", 2015 International Siberian Conference on Control and Communications, SIBCON 2015 - Proceedings, art. no. 7146975, 2015.

[5] Common Vulnerability Scoring System v3.0: Specification Document. FIRST [Online]. Available from: https://www.first.org/cvss/specification-document [retrieved June 2016]

[6] OWASP [online]. Avaiable from: https://www.owasp.org [retrieved June 2016]

[7] Why File Upload Forms are a Major Security Threat. Acunetix [Online].Available from: http://www.acunetix.com/websitesecurity/upload-forms-threat [retrieved June 2016]

[8] What is an SSL certificate? Verisign [Online]. Available from: https://www.verisign.com/en_US/domain-names/web-presence/website-optimization/ssl-certificates/index.xhtml

[9] Symantec [online]. Available from: https://www.symantec.com/ [retrieved June 2016]

[10] Scafrone, K., Mell, P. "An analysis od CVSS version 2 vulnerability scoring", Proceeding of the 3rd International Symposium on Empirical Software Engineering and Measurement, IEEE, 2009, pp.516-525

[11] Microsoft [online]. 2016, Available form: https://msdn.microsoft.com [retrieved June 2016]

Copyright (c) IARIA, 2016. ISBN: 978-1-61208-493-0