Methods of testing computer systems for various kinds of penetration

A I Alchinov¹, S I Polovneva² and G A Ivashchenko³

¹ Institute of Control Sciences of the Russian Academy of Sciences
² Irkutsk National Research Technical University, 83, Lermontov str., Irkutsk, 664074, Russia
³ Bratsk State University, Makarenko Str., 40, Bratsk, 665709, Russia

E-mail: alchinov46@mail.ru

Abstract. Today, information is the most valuable resource. Its possession by the third parties, as well as destruction, may lead to irreversible losses both for any business and for individuals. Accordingly, it is important to protect information systems at all stages of their use. The content of this paper will be useful both to specialists in the field of information security and to ordinary users. The paper focuses on all the main existing ways to penetrate information systems. The paper will allow creating a personal security check sheet.

1. Introduction

Currently, there are various standards of information security formed both by state bodies and expert communities:

- PCI Penetration testing guidance 1.0 (general)
- OWASP Testing guide 4.0
- Bank of Russia ИББС-2.6-2014 (general)

Objectives of the security analysis:

- to conduct express assessment of the current level of the application security;
- to identify drawbacks in existing protection mechanisms.

This process may be carried out, for example, using the BSIMM (Building Security In Maturity Model) [1, 2].

2. Results and Discussion

The Open Web Application Security Project (OWASP) foundation classifies attack vectors and weaknesses of web applications. This is an international non-profit organization focused on analyzing and improving the software security [3-5].

In 2013 OWASP created a list of the 10 most critical security risks to web applications – OWASP TOP-10. It focuses on the most dangerous weaknesses that may cost some people a lot of money and undermine business reputation, or even lead to the loss of business.
Injections
As a rule, all data is stored in special databases, access to which is built in the form of queries. Applications use these queries to receive, add, modify or delete data, for example, when a user edits personal data or completes a questionnaire on the site. If the user does not verify the data well, the attacker may enter a special code in the web interface form of the application containing a piece of “extra” query.

Broken Authentication and Session Management
In order to distinguish one user from another, the web application uses the so-called session cookies. After the user has entered a login and a password, and the application authorization, the browser storage retains a special ID, which the browser later presents to the server each time you request the page of your web application. If the identity is stolen by an attacker, and the corresponding checks were not implemented in the system, the attacker may gain access to the system with the rights of the user account [6-8].

Cross-Site Scripting XSS
This is another user data validation error that allows sending a JavaScript code for execution to the user browser. Attacks of this kind are also often called HTML injections, because their implementation mechanism is very similar to SQL injections, but unlike the latter, the embedded code is executed in the user browser.

Insecure Direct Object Reference
This type of weakness is also a consequence of insufficient validation of user data. When displaying any sensitive data, such as personal messages, an identifier is used to access the object, which is transmitted in open form in the address bar of the browser, and verification of access rights to
the objects is not implemented. By sorting the number after “id=” it will be possible to read other people’s personal messages.

**Security Misconfiguration**

The security of a web application requires a secure configuration of all infrastructure components: application components (frameworks), web server, database server, and the platform itself. Default server component settings are often unsafe and open up opportunities for attacks.

**Sensitive Data Exposure**

Many web applications do not protect sensitive data, such as credit cards and authentication data. Attackers can steal or modify such poorly protected data for their own self-interest. Data is transmitted via HTTP, which in turn is not encrypted in any way, and when data passes from the user’s computer to the web server, the data will pass many different network nodes on which they may be intercepted.

**Missing Function Level Access Control**

Most web applications check access rights before displaying data in the user interface. However, applications must perform the same access control checks on the server when requesting any function. After all, there are still many auxiliary service queries, which are often sent in the background asynchronously. If the query settings are not rigorously checked, attackers will be able to spoof the query to access the data without proper permission.

**Cross-Site Request Forgery**

If the victim enters the site created by the attacker, and for example, a request to the above (false) page of the payment system is secretly sent on its behalf, then it is possible to intercept the payment details.

**Using Known Vulnerable Components**

Often, web applications are written using special libraries that are supplied by third-party companies. In most cases, these components have open source, which means that not only you have them, but also millions of people around the world who work on their source code, including for vulnerabilities.

**Unvalidated Redirects and Forwards**

Web applications often forward a user from one page to another. The process may use improperly validated parameters with the destination page of the forwarding. Without appropriate checks, the attacker can use such pages to forward the victim to a fake site, which, for example, may have a very similar or indistinguishable interface.

### 3. Conclusion

Currently, the focus should be on the injection type of weaknesses (in the OWASP classification it is A1 and A3). Attacks of this type involve mixing code and data in the absence of validation of values controlled by the attacker.

Data transmitted from the user can be processed as XML, which is represented as values or tags. This can be dangerous, because the means of parsing any poorly formed document are used “as is”, but no one is involved in their configuration. As a result, anything can happen from unauthorized actions to executing code on an arbitrary system inside the network (however, in most cases, reading files).

### References

[1] Karlina A I, Kondrat'ev V V, Kolosov A D, Balanovskiy A E and Ivanov N A 2019 Production of new nanostructures for modification of steels and cast irons *IOP Conference Series: Materials Science and Engineering* **560**(1) 012183

[2] Martyushev N V, Egorov Yu P and Utiev M 2002 Computer analysis of the material structure *Proceedings of the 8th International Scientific and Practical Conference of Students, Post-Graduates and Young Scientists: Modern Technique and Technologies* pp 159–161
[3] Kondratiev V V, Karlina A I, Guseva E A, Konstantinova M V and Kleshnin A A 2018 Processing and Application of Ultra disperse Wastes of Silicon Production in Construction *IOP Conference Series: Materials Science and Engineering* **463**(3) 032068

[4] Gura D A, Gura A Y, Volkova T A and Lipilin D A 2020 Recreational space as a factor of sustainable development of coastal geosystems of the Krasnodar region *IOP Conference Series: Earth and Environmental Science* **579**(1) 012007 doi:10.1088/1755-1315/579/1/012007

[5] Voropai N, Ukolova E, Gerasimov D, Suslov K, Lombardi P and Komarnicki P. 2019 A Study on Cost-Effectiveness of Energy Supply Based on the Energy Hub Concept *IEEE PES Innovative Smart Grid Technologies Europe* 8905736

[6] Buryanina N, Korolyuk Yu, Koryakina M, Lesnykh E and Suslov K 2019 Algorithm of Current Protection Based on Three Instantaneous-Value Samples, *IEEE PES Innovative Smart Grid Technologies Europe* 8905542

[7] Balanovskiy A E, Karlina A I, Kolosov A D, Karlina Y I 2021 Study of the effect of nanomodifiers from silicon production wastes on morphological form of gray cast iron graphites. *CIS Iron and Steel Review* **21** 64–69

[8] Sivtsov A V, Elkin K S, Pan’kov V A, Karlina A I 2021 Specific Features of the Electric Mode of the Technological Process of Smelting of Commercial Silicon. *Metallurgist* **64**(9-10) 923–930