Advanced Engineering Technology and Application
An International Journal

http://dx.doi.org/10.18576/aeta/050301

Architecture of Web-Based Intellectual Vulnerability Scanners for OWASP Web Application Auditing Process

Nurmyshev Serik *, *Kanat Kozhakhmet and Lyazzat Atymtayeva*

Kazakh - British Technical University, Almaty, Kazakhstan

Received: 15 Feb. 2016, Revised: 2 Mar. 2016, Accepted: 30 Apr. 2016
Published online: 1 Sep. 2016

Abstract: Nowadays, many business applications such as online banking, e-insurance, e-commerce, mail, etc., are often made as web applications. The rapid popularization and the usage of web technology everywhere, transition of traditional business into the sphere of web applications has made them more attractive to attack for hackers, with the aim to steal confidential user information and use this information for personal gain. Vulnerability scanners are often used in web application security assessments, but there are few properly developed web-based vulnerability scanners, that used intellectual expert based tools. Development of intellectual web-based security scanners for OWASP security standard has high demand in security auditing area. Expert systems in vulnerability scanners can increase effectiveness and decrease a cost of OWASP auditing process.

Keywords: intellectual vulnerability scanner, expert system, OWASP auditing process

1 Introduction

Vulnerability scanners are monitoring and management tools with which it is possible to check computer networks, separate computers, applications and web applications for an existence of problems in security. Nowadays, expert systems were going to be widely used in information security auditing process [1]. They help to decrease the cost of information security auditing process. In addition, using of intellectual approach in OWASP security process becomes obliged, because of high complexity features in the auditing process. In recent years, increased the number of publications applied to this new trend in the field of information security, as an adaptive network security [2]. This line consists of two major technologies - security analysis (safety assessment), and detection of attacks (intrusion detection). In this paper, we consider security analysis. Briefly illuminating traditional active auditing process, we can highlight following issues; the network consists of communication channels, routers, switches, hubs, servers, etc. All of these network elements must be assessed for their effectiveness to prevent from attacks. Vulnerability scanning tools allow explore the network looking for ”weak” place, and by analysing found issues and based on the results of scanning process different kinds of reports are generated. Example of the problems that can be revealed during scanning process [3]:

–Backdoor in code from third-party libraries;
–Use of default or weak passwords;
–Misconfiguration of the firewall, web-servers and other server infrastructure;
–Etc.

Very often people write about unique opportunities of security analysis systems (scanners), bringing readers to the belief that these systems are panacea from all troubles and that they allow finding all vulnerabilities. However, when users come up against a situation which can be described the question asked me recently: ”I have read in Bug track about the new vulnerability in my operating system yesterday. Why the network security scanner doesn’t find it?”. The answer to the asked question is very simple. That vulnerability is not present the database of the vulnerability scanner and it is one of the aspects which inherent in all security analysis systems. They are intended for detection only of the known vulnerabilities, whose description they have in the database. Therefore, they are similar to anti-virus systems, which need to update the database of signatures constantly for effective work. One of the possible solutions is to use multiple expertise of auditors, which can increase the effectiveness of OWASP [4] auditing process.

* Corresponding author e-mail: s.nurmyshev@gmail.com

2 The architecture of vulnerability scanner

Most of a vulnerability scanner consist of four main parts, namely, a Scan Module, a Database Module, a Report Engine and a UI [5].

1. The Scan Module performs system checks for vulnerabilities in accordance with the specified settings, all vulnerability scan logic incorporated in this module. Depending on the implementation of the module can scan multiple parallel resources.
2. The Database Module is a specialized database that contains information about vulnerabilities and their methods of use (for the attacks). These data are supplemented with a recommendation on measures to address vulnerabilities, performing recommendations results in reducing the risk to the security of the system. In general, the database is used for the analysis of security and intrusion detection.
3. Report Engine based on the collected information generates reports describing the discovered vulnerabilities. An important point is that report contains recommendations to address the problems detected. Detailed reports help to remove quickly the detected defects without losing time searching descriptions of vulnerabilities detected. The report can be obtained in a convenient form for the end user.
4. The UI allows operating the vulnerability scanner. Mostly scanner will have Graphical User interface with the option to run scanner just in a command line interface without GUI.

Most vulnerability scanners have a modular architecture; it is convenient because the desired functionality can be turned off. Start the process of scanning based on selected parameters.

3 The limitations of vulnerability scanners

The restrictions of vulnerability scanners are the following [6]:

1. Single scan: Vulnerability scanners perform a safety assessment of system or network only in a certain period. That is why, security scanning of the system should be carried out as often as possible because new vulnerabilities may be resulted due to changes in system configurations or new security holes may appear in the security of the system due to software updates, which are used.
2. Human judgment is needed: Most of the vulnerability scanners can only detect vulnerabilities that are already described in their logic and exist in their knowledge base. After scanning is finished, a human must review the finite report, to make the final decision.
3. Others: They cannot identify other security threats, such as those associated with logical, procedural errors.

Furthermore, many vulnerability scanners use plugins to identify potential vulnerabilities. Plugins are part of the knowledge base such as logic, instructions and other data which allow the scanner to detect vulnerabilities. The scanner can identify only those vulnerabilities that exist in the set of plugins. Despite the fact that scanning to identify vulnerabilities - a powerful tool for the analysis of systems of security, themselves vulnerability scanners cannot fix the situation with the information security in the company. Scan results should be interpreted correctly and, based on these results, adequate measures to protect information assets need to be taken. Also, drawbacks of all scanners should be noted: there is no possibility to add your own review.

4 Using expert systems in combination with vulnerability scanners

Expert System - a software tool that uses the knowledge of experts, to give high-performance solutions to problems. An expert system is called a system, not just a program, since it contains the knowledge base, problem solver and interface. The last one helps the user to interact with the main program. Expert - a person who is able to articulate his thoughts and enjoys the reputation of a specialist who knows how to find the right solutions to problems in a particular subject area. The expert uses his expirience to search more efficient solutions, and expert system models all of his strategies. Expert systems tries to make decisions, which can be done only by experts. So expert systems emulate experts choice making decisions, it is closely related to Artificial Intelligence area. There are different methodologies and approaches that are used [7];

– Rule-based systems.
– Semantic or associative nets.
– Fuzzy systems.
– Neural nets.

You may ask How expert systems can be applied to vulnerability scanning process?. Experts can analyse the vulnerabilities, which are found by the scanner during the process of scanning, and then make a final decision about the general risk level of vulnerabilities and give some recommendations how to fix that. Some vulnerabilities may be used in combination with other ones and by applying social engineering can be resulted in critical risk level. When multiple experts provide their evaluations, after that by combining or choosing the best opinion, by using this knowledge base [8], the system will report about many potential attacks, which cannot be detected by traditional vulnerability scanners. Here is an overview of an authors expert system which uses fuzzy sets and logic to analyze experts evaluation for found vulnerabilities and make a final decision about general risk level and the recommendation for the scanned system

based on experts opinions. Scan process is launched by user and then user waits until report with recommendations is ready. After scanning process is finished experts see all alerts found by scanner. For each alert an expert fills an evaluation and recommendation form with fields:

–Risk level of the vulnerability.
–Confidence of an expert.
–Urgency of fixing vulnerability.
–Use of vulnerability in combination with other ones.
–Experts level in this area
–Solution and recommendation

The fields of this form are used as crisp input variables for fuzzy system. Fuzzy systems also require defining rules, which are the series of if then statements. These statements usually made by experts to get an optimum result, example:

1.If (risk is high) and (confidence is low) and (urgency is ignore) and (is-comb-avail is impossible) and (expert-level is low) then (general-risk is low)
2.If (risk is high) and (confidence is high) and (urgency is later) (is-comb-avail is possible) and (expert-level is med) then (general-risk is med)
3.If (risk is high) and (confidence is high) and (urgency is immediate) and (is-comb-avail is for-sure) and (expert-level is med) then (general-risk is high)
4.If (risk is medium) and (confidence is high) and (urgency is immediate) and (is-comb-avail is possible) and (expert-level is high) then (general-risk is high)
5.Etc

By applying all steps defined in fuzzy systems[9], namely

–Fuzzification of the input variables.
–Rule evaluation
–Aggregation of the rule outputs
–Finally defuzzification

We get final result about general risk level based on experts evaluation for a particular alert. Next step is to make combined recommendation from all experts by applying same process to make a final report for a user.

5 Using vulnerability scanners in OWASP auditing process

The number of threats is growing in proportion to the growth of the business, however, as demonstrated by long-term practice, 99Classification of attack vectors and vulnerabilities is made by people involved in community OWASP (Open Web Application Security Project). It is an international non-profit organization focused on analyzing and improving software security. OWASP has created a list of 10 most dangerous attack vectors to Web-based applications, this list is called OWASP the TOP-10, and it focuses on the most dangerous vulnerabilities, which can cost a lot of money, from undermining the goodwill, up to loss of business. We will go over the list of OWASP TOP-10 and get a closer look at each of these attack vectors[10].

V1 - Injection Most known vulnerability in the top 10 list is SQL Injection. This vulnerability is well-known and can be detected by most of the vulnerability scanners.

V2 - Broken Authentication and Session Management Unlike V1, V2 does not belongs to a specific category of exploits, all type of vulnerabilities that lead to Authentication and Session Management vulnerabilities belong to V2. This category of vulnerabilities cannot be automatically identified by most of the vulnerability scanners. For example, users password is stored in plain text in the database, good practice to use the hash instead of that. An automated web vulnerability scanner can never know how user credentials are stored in the backend of a target system. An expert only can check it. But some of the security issues related to V2 can be detected by automated scanners. For example, session IDs posted in URL or in the cookie or the sending of user credentials through an unencrypted connection.

V3 - Cross-Site Scripting (XSS) There are several types of XSS, persistent and DOM XSS. As technical vulnerabilities, they all of them can be revealed by a security scanner. Before choosing web security scanner make sure that chosen scanner can detect XSS, especially it must support detecting DOM XSS.

V4 - Insecure Direct Object references This category of vulnerabilities that contain logical security problems in target systems. Problems with logic as already told before are security issues, which cannot be automatically detected by vulnerability scanners. V4 refers security issues where some resource, which must have limited access, is not secured properly and can be accessed by anyone. For example, a user of a target system has access to some sensitive information, which he must not see. To avoid that system must check the role and privilege of the user before giving him access. Scanner cannot identify if a current user should have access role to a some URL or not. Only a human who is familiar with a business process of a target system can determine the correct role and privileges for every users.

V5 - Security Misconfiguration This category of vulnerabilities is resulted in misconfiguration in the server during the initial setup of server, framework etc. Unnecessary network services: Turn of unnecessary services such as FTP, DNS and SMTP. The scanner can identify the whether service is launched or not, but human must determine is it needed or not and setup service correctly or shut it down. Out of Date Software: Fore example if the system is built using old versions of some framework, which contains well-known security holes, the scanner will alert about that. The scanner also can identify the language of you framework: PHP, .NET etc., version of the framework and name of the framework:

WordPress, Drupal, etc. Security Settings of Development framework: System can be launched in production with developer options. For example debugging may be enabled, and some functionality may be disabled to speed up the development process. Default Accounts and Passwords: Weak passwords may be detected by brute force, which uses special dictionaries, or default password that comes from the vendor is not changed to new one.

V6 - Sensitive Data Exposure Most of the web applications do not protect sensitive data such as credit cards and credentials for authentication. Attackers may steal or modify such weakly protected data to be used for their own purposes. The simplest example - the transfer of data over HTTP. The fact that data transmitted over HTTP protocol being not encrypted, and the passage of data from the user's computer to the Web-server data will be transferred from a router or a home office router, ISP router, the router on the channel, hosting providers data center router and so on. At each of these nodes hidden malware can exist, for example, sniffer program that reads all the traffic and sends to the attacker, who can view the personal data and credit card data. Such data shall be transmitted only over HTTPS, which is to be read as the corresponding inscription in the address bar of your browser.

V7 - Missing Function Level Access Control The essence of vulnerability, as the name implies, is the lack of availability of proper access to the requested object. Most web application checks the access rights before displaying the data in the user interface. However, applications must perform the same access control checks on the server when requesting any function. After all, there is still a lot of support service requests, which are often sent in the background asynchronously using AJAX technology. If the query parameters are not sufficiently carefully checked, the attackers will be able to forge a request to access the data without proper authorization.

V8 - Cross-site Request Forgery The CSRF attack vector, also known as XSRF, allows an attacker to perform actions on behalf of the victim on the server, where it is not implemented additional testing. For example, in a payment system to transfer funds to another account, there is a page of the form: bank.kz/transfer_operation.asp?operation_amount=4400 &operation_account=55824185

where operation_amount - the amount of money to translate and operation_account - account number where money must be sent.

If the victim visits a site created by the attacker, an attacker sends a request to the page mentioned above of the payment system. As a result - the money goes to the account of the attacker, then, are likely to be quickly converted to Bitcoin, or translated into another irrevocable payment system where money cannot be returned. It is assumed that the victim should have been pre-authenticate to the payment system and must be opened active session (for example, payment system page is open in another browser tab).

V9 - Using Components with Known Vulnerabilities Often, web-applications are written using special libraries and frameworks, which are supplied by third parties. In most cases, these components are open source, which means that not only you use the code, but also millions of people around the world who study the source code for vulnerabilities and it should be noted; often find errors in the code. In addition, often vulnerabilities are found in low-level system components, such as database server, web-server, and finally in the operating system components up to its core. It is important to use the latest versions of the components and monitor for known vulnerabilities appearing on sites like securityfocus.com.

V10 - Unvalidated Redirect and Forwards Web-based applications frequently redirect the user from one page to another. In these process not properly verifiable parameters, indicating the final destination of the redirect page can be used. Without proper checks, an attacker can use these pages to redirect the victim to a fake website that, may have very similar or indistinguishable interface, but will steal your credit card information or other sensitive private data. This type of vulnerability, as well as many others listed above, is a type of incoming data validation errors (input validation).

6 Conclusion

Using intellectual scanners and development of knowledge base system can improve affectivity of information security auditing processing OWASP. In Addition, creating the correct knowledge base of vulnerability sets in expert system of intellectual vulnerability scanners can reduce the cost of the auditing process.

References

[1] Atymtayeva L., Kozhakhmet K., Bortsova G. Building a Knowledge Base for Expert System in Information Security. // Proceedings of the 14th International Symposium on Advanced Intelligent Systems (ISIS2013), 13-16 November 2013, Daejeon, Korea, // Springer Journal Advances in Intelligent Systems and Computing, Volume 270 "Soft Computing in Artificial Intelligence ", pp. 57-77

[2] Atymtayeva L., Kozhakhmet K., Bortsova G., Inoue A. Methodology and Ontology of Expert System for Information Security Audit // Proceedings of the 6th International Conference on Soft Computing and Intelligent Systems and the 13th International Symposium on Advanced Intelligent Systems, 20-24 November 2012, Kobe, Japan, pp. 238-243

[3] Richard R. Linde, Operating system penetration. Santa Monica, California pp. 361-365

[4] Paul E. Black, M. Kass, E. Fong, Proceedings of Workshop on Software Security Assurance Tools, Techniques, and Metrics, Information Technology Laboratory National

Institute of Standards & Technology Gaithersburg, February 2006, pp. 41-42.
[5] S. Kals, E. Kirda, C. Kruegel, and Ne. Jovanovic, SecuBat: A Web Vulnerability Scanner, Proceedings of the 15th international conference on World Wide Web, Pages 247-256
[6] The Government of the Hong Kong Special Administrative Region, An overview of vulnerability scanners, February 2008
[7] N. K. Kasabov, B. Gabrys, K. Leiviska, J. Strackeljan, Evolving Connectionist Systems with Evolutionary Best Practice for Selection and Combination of Intelligent Methods, Springer Berlin Heidelberg 2005, pp 181-202
[8] Sheriyev M., Atymtayeva L. Automation of HCI Engineering processes: System Architecture and Knowledge Representation // Int.Journal "Advanced Engineering Technology and Application (AETA)", Natural Science Publishing, Vol.4, N2 (May 2015), ISSN 2090-9535, pp. 41-46.
[9] Zadeh, L. Fuzzy sets as a basis for a theory of possibility, Fuzzy Sets Syst., 1978, pp. 350
[10] D. Wichers, Owasp top-10 2013, OWASP Foundation, February, 2013.

Nurmyshev Serik received the master's degree in Information Systems at Kazakh British Technical University in Almaty, Kazakhstan. His research interest includes Robotics, Distributed Systems, Expert systems and Vulnerability Scanners.

Kanat Kozhakhmet received the PhD in Computer science, computer technology and management at Kazakh British Technical University in Almaty, Kazakhstan. His research interest includes expert systems in information security.

Lyazzat Atymtayeva received the PhD and Doctor of Science degree in Mechanics, Mathematics and Computer Science at al-Farabi Kazakh National University, Kazakhstan. Her research interests are in the areas of mechanics, applied mathematics and computer science including the numerical and rigorous mathematical methods and models for mechanical engineering and computer science, intelligent and expert systems in Information Security, Project Management and HCI. She has published research papers in reputed international journals of mathematical and computer sciences. She is reviewer and editor of international journals in mathematics and information sciences.