Methodology of Security testing of IKID website and Security Vulnerabilities

Mustofa Kamil^{1*}

¹Master of Information and Technology, Swiss German University, Tangerang 15143, Indonesia

*Corresponding author: mustofa1981@gmail.com

Abstract. Due to the large amount of data stored in web applications and the increasing number of transactions on the web, the right Web Application Security Testing is very important day by day and web application is an important in business life. By increasing complexity of web systems, Security testing has become a very necessary and important activity of the life cycle of developing web applications, web security testing consists of searching for information about the network, application and looking for holes and weakness.

Keyword: Web Security, Parameters of web security, and Security attack, security vulnerabilities

1. Introduction

Due to customer data going online by adapting to online, and GDPR must be compliant for online business, security web testing must be performed before going live. A web security test to search security databases for known vulnerabilities against the system component as summaries below

- Perform network scans to identify any unexpected and/or unauthorized services and ports are in use.
- Perform targeted scan on in scope Network devices and system to identify vulnerabilities
- Conduct a manual investigation to validate the vulnerability
- Vulnerability ranking is based on the level of threat, potential loss and possible exploitation and the availability of exploitation code.

2. Materials and Methods

There are several testing methods available:

- a. Black box Testing – The test is carried out without prior knowledge about the internal structure or design or implementation of the object being tested.
- b. Grey Box testing – Testing must be done with partial knowledge of the internal structure or design or implementation of the object being tested.
- c. White Box Testing - Testing must be done with knowledge of the internal structure or design or implementation of the object being tested.

Testing activities are divided into three categories: Non-invasive, less invasive, and more invasive. These categories are organized into four stages each of which is based on NIST SP800-115 defined Penetration Testing Methodology.

Figure 1 Four-Stage Penetration Testing Methodology

As can be seen in Figure 1, there are several stages in penetration testing methodology.

a. Stage 1 Planning

This stage establishes the basis for successful penetration tests, such as scheduling and resources, where actual testing takes place

b. Stage 2.1 Discovery (Identify Targets and Network mapping)

This penetration test phase involves counting systems in scope and mapping networks and systems from an external perspective to find systems in the target environment, which basically maps the surface of the attack for the remaining phases. Even in a Grey box testing (in which target system personnel provide the testers with a diagram and inventory of target machines), still applies this phase to verify the information provided at the outset of the test. To discover potential target machines. It applies numerous methods and usually do not direct the target environment to abnormal traffic.

- Identify potential Vulnerabilities via Web Research and/or Network Discovery tools.
Information that may be collected from the internet could allow testers to build a profile of how the target environment shows to attackers.
- Network Mapping: This step allows identifying what services are being listened to at the target IP address, previous networks and even hosts behind a firewall. A network diagram shall be built during this step and the asset register. All services identified must be targeted at the next steps.

c. Stage 2.2 Discovery (Vulnerability scanning)

This stage can be divided into two parts, Application level and Network level scanning, and web application level scanning. Application level and Network level scans could be performed to recognize network services that might be vulnerable to attack. All vulnerabilities identified in this stage must be validated by thorough research. Vulnerability scanning suffers from false positives and several vulnerabilities may not apply to the target system. Vulnerabilities must include but are not limited to buffer overflows, network services that are not configured correctly, trust relationships that are not configured correctly, authentication mechanisms that are not safe and outdated network services that are aware of vulnerabilities & etc. This activity does not usually lead to a regularly scanned environment. There are no scanning activities specifically designed to cause disruption to the environment, such as scanning for denial of service vulnerabilities.

d. Stage 3 Attack (Exploitation and password guessing)

This stage forms an interesting part of the penetration test. This is where the testes stand the chance of gaining a footprint on the environment and beating the protection mechanisms. Exploitation can mostly be carried out with certain exploit tools. A few websites can be used to find exploit code.

Manually testing will be conducted via searching vulnerable databases. Password can be conducted with a list of common passwords and potential usernames. These activities while not likely to cause disruption to environment are a more invasive than previous stages.

e. Stage 4 Reporting.

Every item identified, as a problem that may occur, must be followed by exploits that can be proven or not. Problems recognized in one stage must be registered at the end of the stage and carried out for further investigation at the next stage.

Assessment Tools

Following tools will be using for the web security assessment whereas applicable, and they are namely

- Nessus Security Scanner
- Nmap
- Kali Linux
- Burp Suite
- Acunetix Web Vulnerability scanner.

Testing Scenario

This research will use Web Penetration test scenario. In this scenario, pen tester-initiated attacks are done from the external network of the being tested in-scoped system components

Constraints and Assumptions

Penetration testing activities will consist of target identification, network mapping, vulnerability scanning, exploitation of vulnerabilities and password guessing. While most of these activities do not usually have an impact on such environmental operations, stages that involve exploitation of a vulnerability can cause system disruption. There are no tests specifically designed to interfere with computer system operations to be performed. All activities will be carried out in a maintenance window agreed with the operating staff and supporters responsible for the environment as listed above in the Schedule section. All relevant and appropriate penetration testing activities will be informed as they are undertaken and when they have finished. This penetration test is not intended to steal sensitive information and data from the environment. Where access to sensitive information is believed to have been obtained, testing will be terminated, and the vulnerability confirmed with the system owner. Any sensitive information that is accidentally taken by a penetration tester will be safely removed or disguised for reporting purposes.

3. Results and Discussion

Detailed results of open network ports / services identified for the target scope and origin perspective will be listed in Table 1 below.

Vulnerability of cross-site request (CSRF) is a vulnerability might arise when an application relies only on attaching link email or text to identify the user who has issued a specific request. Because the browser automatically adds a cookie to the request regardless of origin, attackers can create malicious websites that fake cross domain requests to vulnerable applications. In order to be vulnerable from CSRF, the next conditions must apply.

Requests can be issued across domains, such as using HTML forms. If the demand contains a header or nonstandard contents, then the demand can only be deleted from pages that are from the identical domain.

This application only relies on http cookies or Basic Authentication to recognize the user who issued the request. If the application displays a section related token using MFA (multifactor authentication) elsewhere in the request, then it might not be sensitive.

- The demand performs several specific actions in the application, which change the application status based on the user's identity issued.
- An attacker can specify all parameters needed to build requests that perform actions. If the demand includes a value that can't be determined or predicted by the attacker, then it is not vulnerable.

Table 1 Vulnerabilities scan result

Findings	Vulnerability name	Risk Level	CVSS Score	Vulnerability Description/Risk
TLSv1 service port is enabled	TLSv1.0 Protocol Detection	High	7	TLSv1 is not longer used anymore, affected software and version of remote code execution.
TRACE /Nessus1046189052.html HTTP/1.1 Connection: Keep-Alive Pragma: no-cache User factor: Mozilla version 4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0) Receive: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, image/png, */* Receive-Language: En Receive-Charset:iso-8859-1,*,utf-8	HTTP TRACK Methods is active or allowed	Medium	5	Remote web server supports TRACK methods. TRACE and TRACK are HTTP methods used to debug web server connections.
Client-to-server Cipher Block Chaining algorithm. They Are supported: 3des-cbc aes128-cbc aes192-cbc aes256-cbc	SSH Server CBC Mode Ciphers Enabled	low	2	The SSH server is configured to support Cipher Block Chaining (CBC) encryption. This allows the attacker to recover the plaintext message from the ciphertext.

Findings	Vulnerability name	Risk Level	CVSS Score	Vulnerability Description/Risk
blowfish-cbc cast128-cbc Server-to-client Chaining Block Chaining Algorithm They are supported: 3des-cbc aes128-cbc aes192-cbc aes256-cbc blowfish-cbc cast128-cbc on port 22				

CSRF is a one form of penetration techniques in website security gaps. and This technique uses a method for entering data user of a site. with this technique is certainly very detrimental website application users because they can steal personal data, sensitive data exist in an application system data accessed by an attacker. To secure from against Cross-Site Request Forgery vulnerabilities is to include in the request relevant additional tokens that are not sent in the cookie: such as parameters in hidden form fields. where tokens are used to generate random cryptographic keys for sending data securely, so it's not possible for an attacker to assign or set any token value issued to other users. The MFA token shall be opened with a user session, and the application must ratify that the valid token must be received before taking any action that results from the request. Use random validation tokens because Different token values are required for each data delivery form. An optional suggestion, it will be easier to implement, to validate that the Host and referrer Header in the relevant request is present and uses similar host/domain name. Nevertheless, using this is somewhat less powerful: historically, the habits in browsers and add-ons that often require attackers to request cross domain counterfeiting that make use of this header to add such defences. And for vulnerabilities Disabling the use of the TLSv1.0 protocol for cryptographically powerful protocols such as TLSv1.2.

4. Conclusion

There are total ZERO (1) High (H) risk, ONE (2) Medium (M) risk and ZERO (1) Low (L) risk level findings have been identified.

Table 2 Result of Web Application Penetration Test

Web Application penetration test		
Risk Level	Exploitable (suspected) (Yes/No)	No. findings of penetration Test
High	0	0
Medium	Yes	1
Low	0	0
AOI	0	0

A Design Level vulnerability or configuration flaw that makes the network or host network vulnerable to malicious attacks from internal/local or remote users. Vulnerabilities may stand in some areas of the network, such as on firewalls, file transfer protocol servers/file share server or Web servers, and operating systems, depending on the level of security risk. Successful exploitation of the vulnerability may vary from the disclosure of information about host to compromise the host.

Table 3 Severity Level

Severity	Level	Description
1	Minimal	The intruder can collect information about the host (open ports, services, and so on) and can use this information to find other vulnerabilities.
2	Medium	The intruder might be able to collect sensitive data or personal information from the host, such as the right software version. With this information, intruders can easily exploit certain software versions of known vulnerabilities.
3	Serious	Intruders may gain access to storage of information specific to the host and include security settings. Vulnerabilities at this level can include disclosure of a portion of the contents of the file, access to certain files on the host. Directory searches, disclosure of filtering rules or security mechanisms, and denial of service attacks or unauthorized use of services, such as sending emails.
4	Critical	The intruder might be able to control the host. For example, vulnerabilities at this level can include full read access to files, and potential backdoors in the system or network. And a list of all users on the system and at the host.
5	Urgent	The intruder can easily gain control of the system or host network, which can cause compromise of the entire network security. Such vulnerabilities at this level can include full access read and write access.

Table 4 Category in Severity Level

Severity	Level	Description
LOW	Low	Vulnerability with a CVSS predicate score of 0.0 TO 3.9
MED	Medium	Vulnerability with a CVSS predicate score of 4.0 TO 6.9
HIGH	High	Vulnerability with a CVSS predicate score of 7.0 TO 10.0

Table 5 Vulnerabilities website score

Vulnerabilities component	Severity Level	CVSS Score
Cross-Site Request Forgery	MED	5

Table 6 Vulnerabilities Total

Row Labels	Sum of Avg. CVSS Score	Sum of Total Finding
HIGH	7	1
LOW	2	1
MEDIUM	5	2
Grand Total	14	4

Figure 2 Percentage of CVSS Score

All loopholes are dangerous, whether it's a gap on the website that arises from coding errors. Awareness of security / website security is indeed needed by everyone. Both beginners and experts must always be aware and up to date on the latest malware.

References

- CVSS score. Available at: <https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator>
- Jaiswal, A., Raj, G., and Singh, D. (2014). Security Testing of Web Applications: Issues and Challenges. *International Journal of Computer Applications*. 88. doi: 10.5120/15334-3667.
- Pangalita, R., Noertjahyana, A., Andjarwirawan, J. Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra Available at: <https://media.neliti.com/media/publications/105546-ID-penetration-testing-server-sistem-inform.pdf>.
- Syarifudin, I. Pen testing and analisis keamanan web PAUD DIKMAS. Jurusan Teknik Informatika dan Komputer, Politeknik Negeri Jakarta, Depok, Indonesia. at: https://www.researchgate.net/publication/324182514_Jurnal_Irwan_Syarifudin_Pentesting_dan_Analisis_Keamanan_Web_Paud_Dikmas.
- NIST SP800-115 Available at: <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf>