Vulnerability Management and Penetration Testing

Web Application Security Assessment Report

January 17, 2022

Document Version

Title	Version	Author	Reviewer	Date
Web Application Security Assessment Report	1.0			January 17, 2022

Contents

1. Introduction

This Web Application Security Assessment report documents the findings of sample company whistleblowing application in production environment. The purpose of the engagement is to utilize exploitation techniques based on the OWASP10 framework to identify and validate any potential vulnerabilities.

2. Scope

The Web Application Security Assessment is conducted on Whistleblowing application which residing in production environment. The goal is to assess the security posture of internal web application located by URLs below and hosted on the web server (xxx.xxx.xxx.xxx)

- https://apps.sample.com.sg/whistleblowing/

Note:

- The Pentest does not include any consideration of Network security control in place such as IPS and Firewall
- Any URLs, database, and application link to Whistleblowing are not tested
- Testing is perform from the same segment of the target

3. Web Application Security Assessment Activities

The assessment provides a point-in-time security analysis and consequential recommendations for improving the security of the application. Security Assessment includes the following activities:

- **Information Gathering** techniques were used through meeting with IT Project Manager and Application Architect Team to review and understand how the application works, what its purpose is and how it has been implemented.
- **Reconnaissance** involved performing active assessment techniques in order to fingerprint the technologies and versions of software in use as well as mapping the available functionality of the application.
- **Authentication Mechanisms** were examined to determine the effectiveness and resilience to subversion techniques.
- **Session Management** implementations were assessed and attempts were made to violate session state to become another valid user or to escalate privileges.
- **Authorization Access Controls** that enforce authorization levels for the application were analyzed in detail to assess the user segregation methods employed and to validate their effectiveness.
- **Data Validation** routines were subjected to tests that consist of supplying unexpected data of various types and lengths, in order to ascertain the potential for exploitation of several classes of vulnerabilities

4. Key Findings and Recommendations

This section describes the key findings identified during the security assessment and recommendations to remediate the findings. The findings are outlined in order of descending risk rating value. There are four criticality levels based on the calculation by business impact as well as likelihood of occurrence. Please refer to Appendix A for details on the risk assessment methodology used in this report.

Findings	Description
H1	Unrestricted file upload
M1	Weak Ciphers Enabled

4.1 Whistleblowing Application H1: Unrestricted file upload

Technical risk: High	Privileges required: None

Description

Many application's business processes allow for the upload and manipulation of data that is submitted via files. However it is necessary to only allow certain file types / extension to be uploaded. The risk in that by allowing users to upload files, attackers may submit an unexpected file type that could be executed and adversely impact the application or system through attacks that may deface the web site, perform remote commands, browse the system files, browse the local resources, attack other servers, or exploit the local vulnerabilities.

Vulnerabilities related to the upload of unexpected file types is unique in that the upload should quickly reject a file if it does not have a specific extension. Additionally, this is different from uploading malicious files in that in most cases an incorrect file format may not by itself be inherently malicious but may be detrimental to the saved data. For example, if an application accepts Windows Excel files, if a similar database file is uploaded it may be read but data extracted may be moved to incorrect locations.

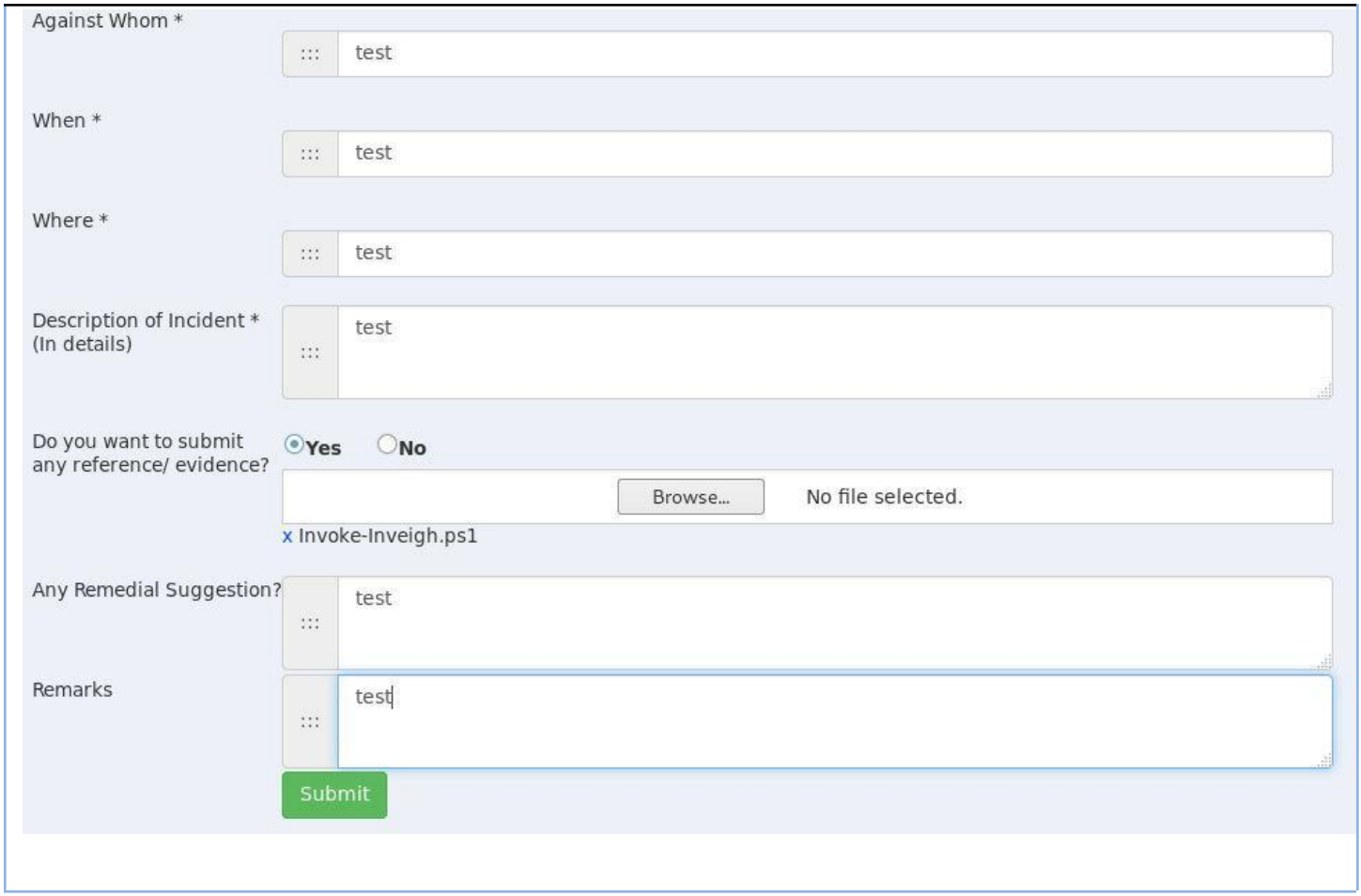
4.1.1: Proof Of Concept

Validation

By uploading unsupported file types

1. Review the website (https://apps.sample.com.sg/whistleblowing/) and perform some exploratory testing looking for the file types that should be unsupported by the application/system.

2. Try to upload a text file with Powershell script extension and verify if the file is properly rejected.



4.1.2: Impact

Description

1. Payload Delivery

The payload may compromised the web server by uploading and executing a web-shell which can run commands, browse system files, browse local resources, attack other servers, and exploit the local vulnerabilities, and so forth.

2. Phishing

Phishing page may potentially uploaded into the website.

4.1.3: Recommendation

Description

1. Restrict File Extension

Restrict the application to accept a filename with specific allowed extensions without having a white-list filter. Ideally this function should be implemented using code that actively checks and allows only valid file types based on a complete file inspection rather than a check of the file extension name.

2. Accept Only Alpha-Numeric

If there is no need to have Unicode characters, it is highly recommended to only accept alpha-numeric characters and only one dot as an input for the file name and the extension.

3. Limit The File Size

Limit the file size to a maximum value in order to prevent denial of service attacks.

4. Limit Upload Directory Permission

Uploaded directory should not have any "execute" permission.

5. Implement Server Side Validation

Don't rely on client-side validation only

4.1.4: Conclusion

Description

1. Fixed on file upload function

The upload function has been fixed by blacklisting all other extension except JPG or PDF.

4.2 Whistleblowing M1: Weak Ciphers Enabled

Technical risk: Medium	Privileges required: None

Description

Sensitive data must be protected when it is transmitted through the network. As a rule of thumb, if data must be protected when it is stored, it must be protected also during transmission.

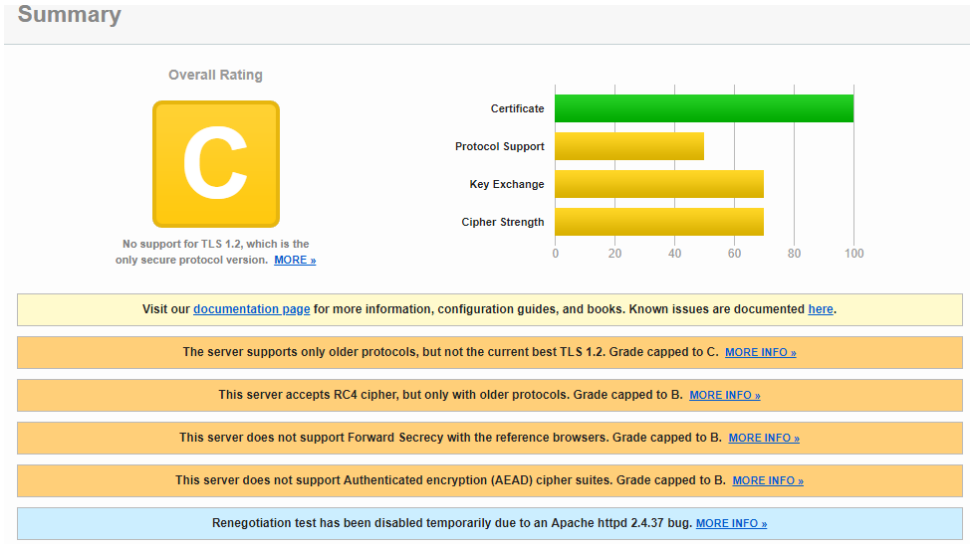
HTTP is a clear-text protocol and it is normally secured via an SSL/TLS tunnel, resulting in HTTPS traffic. The use of this protocol ensures not only confidentiality, but also authentication. Servers are authenticated using digital certificates and it is also possible to use client certificate for mutual authentication.

Even if high grade ciphers are today supported and normally used, some misconfiguration in the server can be used to force the use of a weak cipher - or at worst no encryption - permitting to an attacker to gain access to the supposed secure communication channel. Other misconfiguration can be used for a Denial of Service attack.

4.2.1: Proof Of Concept

Validation

1. Using SSLscan and web application scanning application, we are able to determine the cipher strength and the weakness of encryption.
2. Using Netsparker



3. Using SSLscan

4.2.2: Impact

Description

Attackers might decrypt SSL traffic between your server and your visitors.

4.2.3: Recommendation

Description

1. For Microsoft IIS, you should make some changes to the system registry.

Click Start, click Run, type regedt32 or type regedit, and then click OK.

In Registry Editor, locate the following registry key:
HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders

Set "Enabled" DWORD to "0x0" for the following registry keys:

SCHANNEL\Ciphers\DES 56/56
SCHANNEL\Ciphers\RC4 64/128
SCHANNEL\Ciphers\RC4 40/128
SCHANNEL\Ciphers\RC2 56/128
SCHANNEL\Ciphers\RC2 40/128
SCHANNEL\Ciphers\NULL
SCHANNEL\Hashes\MD

4.2.4: Conclusion

Description

1. Reduced risk for the weak cipher

The risk impact for the weak cipher has been reduced as there is no credential being transmitted over the client and server side.

5. Appendix A – Risk rating and definitions

The overall risk rating is defined as a product of the Likelihood and Impact:

- Likelihood indicates the finding's potential for exploitation. It takes into account aspects such as skill level required of an attacker and relative ease of exploitation.

- Impact (Technical Factor) indicates the finding's effect on technical and business operations. It covers aspects such as the confidentiality, integrity, and availability of data or systems; and financial or reputational loss.

The table below provides a risk rating matrix to classify each identified threats into one overall risk level.

Overall Risk Rating = Likelihood x Impact				
Impact (Technical Factor)	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Low	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood (Threat Agent and Vulnerability)			

Rating	Definition of technical risk rating
High	Finding reveals a serious vulnerability that could result in a loss of system control, access, application control and/or exposure of customer data via the compromise of administrative accounts and/or other system functions. It could also create an exposure of confidentiality or integrity, resulting in many user accounts being compromised or restricted system functions being accessed.
Medium	This vulnerability does not directly lead to a compromised administrative or user account, but could be used in conjunction with other techniques to compromise accounts or perform unauthorized activity on the site or application.
Low	This vulnerability creates limited exposure for compromise of user accounts or unauthorized access to data due to configuration issues, outdated patches and/or policy.