Internal
Doc. Highly
Classified

March 24, 2022

Information Security Audit Report

ISM Digital Assignment-II

Winter Semester 2021-22
Based on Use Case 14

Presented to

Prof. Lavanya K

Presented by

Tushar Verma, 19BCE0662
Sharadindu Adhikari, 19BCE2105
Soumyadip Mondal, 19BCE2107

INDEX

1. PRELUDE & USE CASE DESCRIPTION

1.1. Client Background

An alliance of independent banks and one of the largest financial players in Scandinavia. Having more than 300 branches all over the region, the bank offers finance, savings, insurance, and payment products to its private and corporate customers.

1.2. Business Challenge

- To improve banking service offerings and drive customer engagement- banks move away from legacy systems, introduce modern customer-facing applications, back-office solutions based on the latest technologies.

- To optimize its credit activities, we're to implement a new core-banking solution: to automate and ensure an efficient process of determining a borrower's credit worthiness and granting of credits.

1.3. Implementation (Solution)

Advanced web-based solution to support credit-analysis and decision-making process:

- To focus on optimizing and increasing the efficiency of process flow: automated data capture, objective decision-making criteria, support for automated pricing, and credit approval.
- To be integrated with the bank's internal and third-party systems.
- To design modern, vendor-agnostic architecture with flexible configuration.

1.4. Auditing Framework

In the given use case, (USE CASE-14), the solution is implemented using a web application to support the decision-making process. The OWASP auditing framework is a commonly used framework used to assist web developers and security practitioners to better secure web applications. The proliferation of poorly-written web applications may lead to easily exploitable vulnerabilities, putting the client at risk of malware, identity theft, and other attacks. OWASP has become a standard auditing framework for web application testing and has helped increase the awareness of security issues in web applications through testing and better coding practices.

Given use case incorporates automated data capture, objective decision-making, automated pricing, and credit approval. It's integrated with third parties as well inculcating a vendor-agnostic flexible configuration. Thus, OWASP is a feasible choice for an auditing framework as it includes Injection, Broken authentication, Sensitive Data Exposure, XML External Entities, Broken Access Control, Security Misconfiguration, XSS, Insecure deserialization, Use of components with known vulnerabilities, and Insufficient Logging & Monitoring.

Auditing can be done by focusing on specific features of the given use-case with the help of the modules supported by OWASP- Information gathering, Configuration management, Authentication testing, Session management, Authorization Testing, Business logic testing, Data validation testing, Denial of service testing, Web services testing and AJAX testing.

OWASP offers a good balance between effort invested and results as it allows to provide a correct platform analysis, ensuring that all attack vectors have been analyzed and all security

issues have been detected. This process helps improve the security and protection of the client's IT systems. Using this framework, the web application is analyzed for usual weaknesses associated with a greater impact on the system security.

2. INTRODUCTION

2.1. Background

Jeevan Bank retained the services of SoftSeq LLC to perform an application security audit of Jeevan Bank Secure Web solution according to the requirements of OWASP Application Security Verification Standard 3.0.1.

This document describes the timing, scope, and methodologies taken during the security assessment and audit. The assessment relies on information gathered from Q&A, additional meetings with technical staff, project documentation, as well as on the results of manual and automatic testing of specific threat scenarios. It included theoretical and practical assessment methodologies, best practices used to mitigate potential threats, and techniques of attacks performed by a malicious entity.

OWASP ASVS 3.0.1 Level 2 had been chosen as an audit baseline, and is asuperset of:

- OWASP Top 10 issue classes
- CWE-25
- PCI DSS technical requirements to the security of developed applications

2.2. Purpose

To conduct a comprehensive Information Systems and Security Audit of Credit Analysis and Approval for a Large Bank, including IT Governance. Bank seeks to have an external examination of the IT security.

- To ward off risks in the IT Domain and to appraise the findings thereof to the Management.
- To determine the effectiveness of planning and oversight of IT Activities.
- Evaluating adequacy of operating processes and internal controls.
- Determine adequacy of enterprise-wide compliance efforts relating to Policies and Internal Control Procedures.
- Identifying areas with deficient Internal Controls recommend corrective action to address deficiencies.

2.2. Objectives

The main objective of the security assessment was to perform an in-depth security review in order to identify security vulnerabilities at the application levelthat may jeopardize customers' systems and customers' information.

To this end, the requirements of OWASP Application Security VerificationStandard 3.0.1 Level 2 were strictly adhered to.

Other security assessment objectives include:

- Verifying that authentication & authorization controls are implementedproperly in the application.
- Inspecting business logic at the design and implementation levels.
- Detecting security vulnerabilities at the application level, which could potentially jeopardize **web developers and security practitioners'** systemsand data that is processed and/or stored in **Jeevan Bank Secure Web**.
- Reviewing the security practices used in the configuration of the databases, application servers, and other application-supporting components, modules, or integrated third-party components.
- Providing mitigating controls for secured design, implementation, andconfiguration of the product.

2.3. Testing Methodology

1. Information Gathering
    - Looking for information on publicly available resources
    - Inserting technical information provided by the organization
    - Non-intrusive scan to determine systems, servers and services
2. Planning and Analysis
    - Analysing the possible risks and vulnerabilities
    - Designing the overall testing approach
3. Vulnerability Detection and Identification
    - Searching for vulnerabilities on the resources
    - Enumerating known flaws, loopholes and mis-configurations
    - Manually probing the target, looking for vulnerabilities
4. Pre-emptive Exploitation
    - Customizing and using readymade exploits for a few known vulnerabilities
    - Building exploits for uncommon specific security loophole
    - Testing the exploits against vulnerabilities
    - Escalating the privileges to exploit higher roles, systems and services
5. Reporting
    - Executive Report for Top Management

2.4. Definition of Terms

- BANKRUPTCY: A state where a person or firm is unable to meet their financial obligations.
- MANAGEMENT: management is the study of decision-makers from the supervisor and line managers at lower levels to the Board of Directors.
- LOANS AND ADVANCES: These are credit facilities granted by banks to their customers. They could be short, medium or long-term depending on the length of period of repayment.
- OVERDRAFT: A credit facility (usually short term) granted by banks to current account holders and it carries interest charges on daily basis.
- BANK: Section 61 of BOFIA 1991 Act defines a banking business as business of receiving deposits on current account or other similar account paying or collecting cheques drawn by or paid in by customers.

- CUSTOMER: A person is a customer if he or she has account with the bank.
- FINANCIAL RATIO: These are ratios usually expressed in mathematical terms to test the financial obligations.
- FINANACIAL STATEMENT: They are firm balance sheets, profit and loss account and classified statement which show the financial state of affairs of the firm.
- GUARANTOR: A person or group of persons who stand for bank customers for credit facilities.
- COLLATERAL/ SECURITIES: is an asset presented by a customer to his bank to secure a credit facility granted to him by the bank.

2.5. Report and Compliance

The testing report includes the following sections:

- Overall High-Level Summary and Recommendations (non-technical).
- Methodology walkthrough and detailed outline of steps taken.
- Each finding and any additional items that were not included.

This report can be used to support the regulatory and compliance requirements of:

- CERT-IN
- ISO 27001 ISMS
- PCI-DSS
- HIPAA
- GLBA

3. EXECUTIVE SUMMARY & AUDITING TEMPLATE

A comprehensive security assessment has been conducted in order to determine existing vulnerabilities and establish the current level of security risk associated with the environment and the technologies in use. This assessment harnessed testing and social engineering techniques to provide Bank's IT Management with an understanding of the risks and security posture of their corporate environment. The purpose was to point out security loopholes, business logic errors, and missing best security practices. The tests were carried out assuming the identity of an attacker or a malicious user but no harm was made to the functionality or working of the application/network.

3.1. Scope of Testing

We have raised five priority 1 recommendations, fifteen priority 2 recommendations and four priority 3 recommendations, where I believe there is scope for improvement within the control environment. These are summarised below:

A. IT Security

Security of the Bank's network as well as its data and information is dependent on appropriate IT security controls being defined and policies being implemented. We looked at various areas of IT security as set out in the scope of the audit and found that, although there are controls and policies in place, these should be enhanced to improve security and provide a greater level of protection. For example, we shall recommend that the network password controls be

improved, greater audit logging and monitoring is performed, additional authentication is introduced for remote access, the requirement to perform a laptop asset audit to help ensure that the latest security controls have been employed on all portable PCs, similarly all mobile phones that have an email and data capability should have additional protection such as passwords and encryption enforced when issued as standard. In addition, we have also recommended that only approved Council issued encrypted USB devices be permitted.

B. Information Security and Governance

We've performed audit testing on the Information Security Policies, roles and responsibilities over Information, Data Classification and Asset Ownership. As a result of the audit, we've identified that currently Information Asset Owners have not been identified for Council information, that there is no process in place to identify and classify data according to its sensitivity. We've also identified that there is no formal records management process in place. The audit identified that although a data sharing protocol is in place, this was created in 2001 and has not been reviewed for some time.

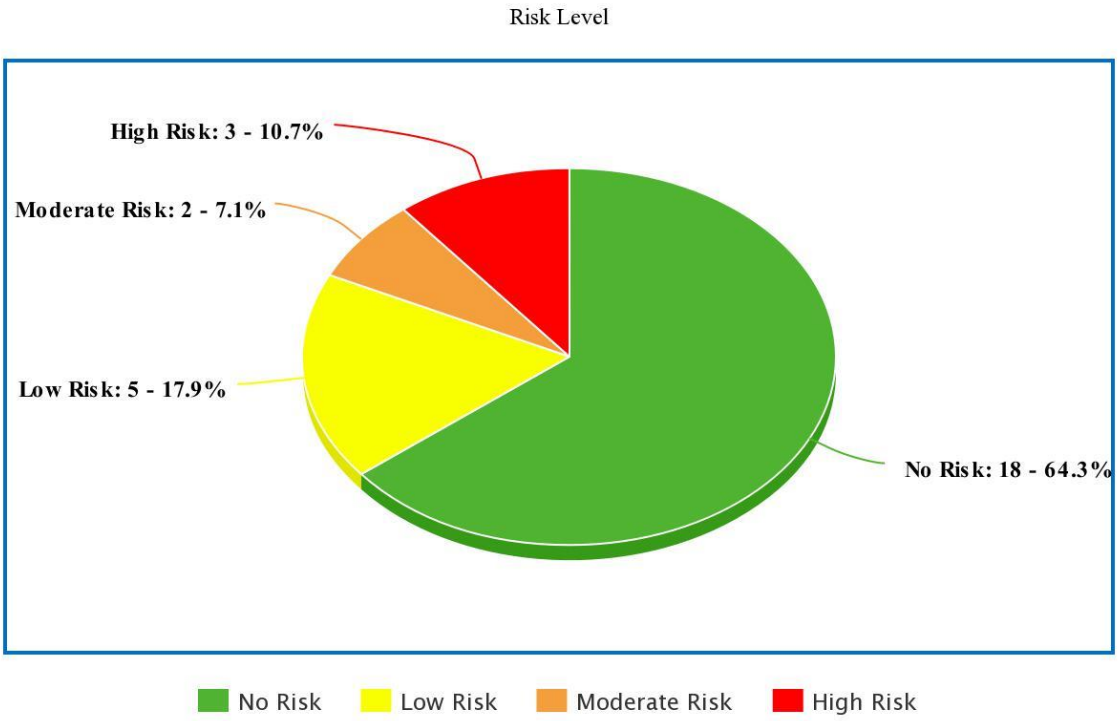
In addition, we could not identify who had signed up to the protocol. Although the Council has a number of policies and procedures established for Information Security, these could benefit from consolidation into a single document covering all major areas of IT related security. We shall also recommend that the ownership of this document be formally assigned as the Information and data sharing protocols and agreements in place have not been reviewed for a long time to confirm validity and adequacy and since the departure of the Information Security Officer, there is currently no specific designated Security Officer in post. This role is currently performed by an IT officer who apart from this role and their other IT roles is also designated as the Records and Data Management officer.

C. Site Visits

As a result of the original information security incident, we've also visited three randomly selected sites within the bank. Some of the issues identified include records created on spreadsheets or databases that are not protected, records are retained longer than necessary, and access to files and folders is not restricted.

3.2. Key Findings & Vulnerabilities' Summary

Total Tests	**28**
Vulnerabilities found	10
No risk level	18
Low risk level	5
Moderate risk level	2
High risk level	3



4. TESTING

4.1. Assessment Phases

The security assessment was performed by a team comprised of applicationsecurity experts, and included the following activities:

1. Analysis of the product structure, interfaces, data flow, sensitive modules, infrastructure and architectural aspects, reliance on third-party products orinterfaces, and identifying classes of vulnerabilities.
2. Information gathering from various sources - human and technological.This included communicating with both technical people and management.
3. Hands-on testing of the product in various scenarios, with respect to previously obtained knowledge of the product and its data flow scenarios.
4. Analysis of gathered data and results from the previous security assessments. The analysis includes categorizing the detected vulnerabilities and prioritizing them according to the business andtechnical context of the application.
5. A final and comprehensive report of the security review activity, summarizing the entire review process, the methodology, and the detailedfindings.

4.2. Assessment Report

Sl. No.	Check/ Audit Requirements	Audit Findings	Recommendations/ Corrective Actions	Risk Level
1.	Verify that all application components are	Deprecated functional and unused codebases were found during the audit.	Refactor or remove deprecated/legacy functionality, delete	

	identified and are knownto be needed.		unused codebase. The legacy codebase can lead to unpatched security issues within it (which won't be fixed since features are unused) and potential reuse of vulnerable code in the future.	
2.	Verify that all components, such as libraries, modules, and external systems, that are not part of the application but that theapplication relies on tooperate are identified.	All external components that the application relies on have been identified by providing an architecture scheme and a complete list of application libraries.	None required	
3.	Verify all security controls (including libraries that call externalsecurity services) have a centralized implementation.	Some security controls do not have a centralized implementation (such as access controls and HTML sanitization). Input filtering and authentication are centralized.	Develop centralized security controls (authorization, sanitization) for the applicationor, at least for the most criticalpart of the application such as, the admin panel.	
4.	Verify that components are segregated from each other via a definedsecurity control, such as network segmentation, firewall rules, or cloud-based securitygroups.	Admin, API, and the client application are deployed asseparate applications. Azure DB is used as anapplication database. Application servers do nothave any limitations on accessing external resources.	Processes within the applications server should have restricted access to external resources (except application, crawler, and system updates)	
5.	Verify that there is no sensitive business logic, secret keys, or other proprietary information inclient-side code.	Sensitive business logic, secret keys, or other proprietary information hasn't been found in the client-side code during the audit.	None required	
6.	Verify that all application components, libraries, modules, frameworks, platforms, and operatingsystems are free from known vulnerabilities.	A list of components has been provided by developers and checked against a known vulnerability database. No issues have been identified.	None required	

7.	Verify that a path can bebuilt from a trusted CA toeach Transport Layer Security server certificate and that each server certificate is valid.	A server is using acertificate for *.**JeevanBankSecureWeb.com** with SHA-256, RSA-2048/4096 algorithms.CA for Rapid SSL and a root for the certificate chainis GeoTrust certificate with SHA-1 and RSA-2048. SHA1 is known as insecure but it is acceptable for a root certificate.	None required	
8.	Verify that TLS is used for all authenticated connections/ those involving sensitive data/ functions. Verify it doesn't fall backto insecure/ unencrypted protocols and that the strongest alternative is the preferred algorithm.	TLS/HTTPS is not preferred by default for internal and external connections which exposes the application and its users to traffic spoofing and modification. For the given TLS configuration, some weak cipher suites could be used for the connection.	Enable TLS for all internal and external connections which transfer sensitive information. Disable support for weak cipher suites.	
9.	Verify that all connections to externalsystems involving sensitive information/ functions are authenticated.	External connections: - Azure SQL Database - Azure Service Bus (Emailmodule, Push module) - Media Server Not all of the connections are authenticated.	Use TLS with encryption with authentication (FIPS 140-2) for communication between application components.	
10.	Verify that TLS certificatepublic key pinning (HPKP) is implemented with production and backup public keys.	HTTP public key pinning is not implemented for the application.	Implement an HPKP header with main and backup certificate and an appropriate revocation time	
11.	Verify that HTTP Strict Transport Security headers are included on all requests and for all subdomains (e.g. Strict-Transport-Security:max-age=15724800; includeSubdomains)	HSTS header is not implemented.	Add an appropriate HSTS header for the application's responses	

12.	Ensure forward secrecyciphers are in use to mitigate passive attackers recording traffic.	Forward secrecy ciphers are in use and are preferred by the server. A list of the ciphers in preferred order is attached.	None required	
13.	Verify that proper certification revocation, such as Online Certificate Status Protocol (OSCP) Stapling, is enabled and configured.	CRL and OCSP revocations are supported.	None required	
14.	Verify that only strong algorithms, ciphers, and protocols are used, through all the certificatehierarchy, including root and intermediate certificates of your selected certifying authority.	Root CA's certificate is signed with the use of anSHA-1 algorithm which isknown as insecure, but since root CA is self-signed, usage of SHA-1 is not an issue	None required	
15.	Verify that the TLS settings are in line with current leading practice,particularly as common configurations, ciphers, and algorithms become insecure.	TLS 1.0 is known as weakand allowed for use by theserver. Ciphersuites with RC4 cipher (which is known as insecure) are allowed foruse by the server.	The server should be configured not to support TLS 1.0 protocol and RC4 cipherfor TLS connections.	
16.	Verify all pages and resources by default require authentication except those specificallyintended to be public (Principle of complete mediation).	Resources that do not require authentication areexplicitly marked with an attribute [AllowAnonymous]. Sensitive info marked with [AllowAnonymous] or out ofthe scope of authentication enforcement hasn't been identified. However, API documentation is exposed to the public but is used only by a small group of users.	Hide API documentation under authentication or custom URL	
17.	Verify all authentication controls are enforced onthe server-side.	It was found that all authentication controls areenforced on the server-side.	Not required.	

18.	Verify all authentication controls fail securely to ensure attackers cannotlog in.	Major security controls have been reviewed andfound to fail securely.	Not required.	
19.	Verify password entry fields allow, or encourage, the use of passphrases, and do notprevent password managers, long passphrases, or highly complex passwords from being entered.	The application allows using complex and long passwords (100 symbols). According to OWASP recommendations, the max password length is 160 symbols, but the current max length is fine too.	None required.	
20.	Verify that the applicationcorrectly enforces context-sensitive authorization so as to notallow unauthorized manipulation by means of parameter tampering.	No specific context-sensitive authorization is observed, see 4.1 and 4.4 for access control details.	None required	
21.	Verify the app will only process business logic flows in sequential step order, with all steps being processed in realistic human time, andnot processed out of order, skipped process steps from another user, or too quickly submitted transactions.	No multi-step business logic that can be exploited (with a known benefit for an attacker) by performing described attacks has been found.	None required	
22.	Verify the application has business limits and correctly enforces on a per-user basis, with configurable alerting and automated reactions to automated or unusual attacks.	Taking into account the importance of relevance (time-based) for sensitive marketing/concept/idea-development data, short-time access to several newest developments could represent the same risk as access to the whole history of "ideas" at the platform which makes implementation of a separate access-limit system not effective.	None required	

23.	Verify that all SQL queries, HQL, OSQL, NoSQL, and stored procedures, calling of stored procedures are protected by the use of prepared statements or query parameterization, and thus not susceptibleto SQL injection.	It was found that all database queries (in TIBox.BLL/Repositories) are implemented via EntityFramework, and stored procedures (in DBScripts/procedures) implemented with parameterized queries, which is considered secure.	None required.	
24.	Verify that structured data is strongly typed and validated against a defined schema including allowed characters, length, and pattern (e.g. credit card numbers or telephone, or validating that related fields are reasonable)	Input validation is enforced using ASP MVC models	None required.	
25.	Verify that server-side input validation failures result in request rejectionand are logged.	Malformed requests are not rejected and logged.	Provide logging functionality for input validation failure cases	
26.	Verify that input validation routines areenforced on the server-side.	Input validation is enforced on the server-side using ASP MVC models.	None required.	
27.	Verify that the runtime environment is not susceptible to buffer overflows, or that security controls preventbuffer overflows.	Buffer overflows are hardly reproducible for ASP.NET and are possible if "unsafe" keyword, Marshal and similar libraries, StructLayoutKind.Explicit is used. These and alternative signs have not been identified during the code review.	None required.	
28.	Verify that the applicationis not susceptible to LDAP Injection, or that security controls prevent LDAP Injection.	Out of scope. The application doesn't use LDAP for user access, management, or authorization. But the option is accessible for clients with on-premisesinstallation.	None required.	

4.3. Limitations

- The security engagement was based on past experiences, available information, and known threats at the time the work was conducted.
- Security issues that could negatively disrupt and impact normal system operations, including Denial of Service (DoS) or buffer overflow attempts, were not fully tested as part of this assessment.
- Technical testing activities were limited to a finite time period. Malicious users may be able to discover and attempt additional security issues over a longer period of time or through other methods such as social engineering.
- As technologies and risks change over time, the vulnerabilities associatedwith the operation of {customer} products included in the Security Review report, as well as the actions necessary to reduce the exposure to such vulnerabilities may change.
- All information systems, which are designed by, and, therefore, dependenton human beings, are always vulnerable to some degree.
- Both Social Engineering and Client-side attacks were outside the scope of this assessment.

5. RECOMMENDATIONS

5.1. General Recommendations

- Ensure that strong credentials are use everywhere in the organization.
- Establish trust boundaries.
- Implement and enforce implementation of change control across all systems.
- Implement a patch management program
- Conduct regular vulnerability assessments

5.2. Internal Audit Recommendations

All front office lending outlets should be audited regularly (at least bi-annually) as an independent check of their activities. However, more frequent inspections and audits are to be conducted if situation is demands. Particular attentions should be paid to the corporate and Authorized Dealer (AD) branches which are expected to originate and maintain the bulk of the credit and investment portfolios.

Compliance Requirements are:

- Large Bank Circulars and other regulations are maintained and updated regularly.
- Guidelines with regard to CIB reporting, provisioning and write -off of bad and doubtful debts, and suspension of interest accrual are strictly enforced. These require the approval of the. Board, as recommended by the MD-CEO.
- The performance of all external service providers (e.g. property appraisers, lawyers, insurers, CPAs, etc.) are reviewed on a periodic basis.

5.3. IT Security

- Remote Access Controls
- Review of Access to Drives, Directories and Folders

- Laptop Management
- Security of Mobile Phones
- Password Controls
- Formal user administration and leavers process
- Audit Policy Settings and Logging
- Accounts with Non Expiry Passwords
- Use of USBs (Memory Sticks)
- Hardware Disposal Procedures
- Legal Banner

5.4. Information Security and Governance

- Records and Information Management
- Data Sharing Protocols
- IT Policies
- Security Officer Responsibility
- Information Owner and Classification
- Security of Laptops
- Use of Emails – Monitoring

5.5. Observations and Recommendations from Site Visits

- File and Database Protection
- Archiving of Records
- Recycle Bin on PCs
- Data Protection Training
- Confidentiality and Data Protection Statement
- Generic use of Email Account

6. REFERENCES

[1] Web Audit - OWASP Web Security Audit | Cybersecurity (tarlogic.com)
[2] example-Compliance-Audit-Report-OWASP-ASVS-L2.pdf (softseq.com)
[3] Logging - OWASP Cheat Sheet Series
[4] Software Security Audit - SoftSeq
[5] What Is the OWASP Top 10 and How Does It Work? | Synopsys
[6] OWASP and its importance to Application Security – Conviso AppSec
[7] The benefits of OWASP | Codebots