Security report

Security report

11-01-2023

Student: Saeed Ba Wazir

Class: S3-CB05

Version: 1.2

Table of contents

○

1. OWASP top 10

	Likelihood	Impact	Risk	Actions Possible	Planned
A01: Broken Access Control	Low	Severe	High	Fixed.	Yes
A02: Cryptographic Failures	Low	Severe	Low	1.Not saving the user information in the local storage and save short live token.	Yes
A03: Injection	Minimum	Severe	Low	Fixed.	Yes
A04: Insecure Design	High	Severe	High	1. Adding unit tests. 2. Adding integrational tests. 3. Design anti-bot.	Yes, but not the third action
A05: Security Misconfiguration	Low	Moderate	Low	1. Using docker when deploying .	Yes
A06: Vulnerable and Outdated Components	Low	Moderate	Low	Fixed.	Yes
A07: Identification and Authentication Failures	Medium	Moderate	Moderate	1. Notifying the admin incase of potential attacks. 2. Adding Application session timeouts.	No, risk accepted
A08: Software and Data Integrity Failures	Low	Moderate	Moderate	1. Ensure that the data is sent	No, risk accepted

				to secure clients.	
A09: Security Logging and Monitoring Failures	Maximum	High	High	1. Making monitoring system 2. Adding logging to the system	No, risk accepted
A10: Server Side Request Forgery	Low	Severe	Moderate	Fixed	Yes

2. Explanation and motivation

A01: Broken Access Control

Broken access control is a serious security vulnerability that can have significant consequences for an organization. Access controls are put in place to ensure that only authorized users have access to certain resources or sensitive information.

There are several reasons why broken access control vulnerabilities can have a high likelihood of occurrence. One reason is human error, such as when a user accidentally grants access to an unauthorized user or fails to properly secure their login credentials. Another reason is inadequate security measures, such as weak passwords or a lack of multi-factor authentication, which can make it easier for unauthorized users to gain access to a system.

The impact of broken access control can be severe, as it can result in unauthorized access to sensitive data, modification of system resources, and even complete system compromise. This can have significant consequences for an organization, including financial losses, damage to reputation, and legal liabilities. It is therefore important for organizations to prioritize addressing broken access control vulnerabilities and implementing strong access controls to prevent unauthorized access and protect against potential security breaches.

A02: Cryptographic Failures

Cryptographic failures are security vulnerabilities that occur when a system's cryptographic protections are compromised or fail to properly protect against attacks. These types of vulnerabilities can have a low likelihood of occurrence, as they often require a high level of technical expertise and specialized knowledge to exploit. However, the impact of cryptographic failures can be severe, as they can allow attackers to gain access to sensitive information or resources that were previously protected by cryptography.

One common cause of cryptographic failures is the use of weak or outdated cryptographic algorithms. These algorithms may be easier for attackers to crack, potentially allowing them to access protected information. Another cause of cryptographic failures is the improper implementation of cryptography, such as using the same cryptographic keys for multiple purposes or failing to securely store keys.

It is important for organizations to prioritize addressing cryptographic failures and to regularly review and update their cryptographic protections to ensure that they are effective at protecting against attacks. This includes using strong cryptographic algorithms and implementing proper key management practices to prevent unauthorized access to sensitive information. Ignoring cryptographic failures can have severe consequences, including financial losses, damage to reputation, and legal liabilities.

For possible solutions, instead of saving the user information in text plain, it is possible to save a token only and send a request to get the user information with every request. With this solution it will increase the security of the website.

A03: Injection

Injection is a type of security vulnerability that occurs when an attacker is able to execute arbitrary code or commands by injecting them into a system. This type of vulnerability can have a low likelihood of occurrence, as it often requires a high level of technical expertise and specific knowledge of a system's vulnerabilities to exploit. However, the impact of injection vulnerabilities can be severe, as they can allow attackers to gain access to sensitive data or resources, modify system configurations, or even take complete control of a system.

One common type of injection vulnerability is SQL injection, which occurs when an attacker is able to execute malicious SQL commands by injecting them into a database. Another type of injection vulnerability is cross-site scripting (XSS), which occurs when an attacker is able to inject malicious code into a website, allowing them to steal user data or manipulate website content.

It is important for organizations to prioritize addressing injection vulnerabilities and to implement secure coding practices to prevent attackers from injecting malicious code into a system. This includes input validation, sanitization, and proper handling of user-supplied data to prevent attackers from injecting malicious code. Ignoring injection vulnerabilities can have severe consequences, including financial losses, damage to reputation, and legal liabilities.

A04: Insecure Design

Insecure design is a security vulnerability that occurs when a system is designed in such a way that it is inherently vulnerable to attacks. This type of vulnerability can have a high likelihood of occurrence, as it often results from a lack of security considerations during the design process. The impact of insecure design can be severe, as it can allow attackers to easily exploit vulnerabilities and gain access to sensitive data or resources, potentially leading to a security breach.

There are several reasons why insecure design can be a high-risk vulnerability. One reason is that it can be difficult to detect and fix, as it is often built into the very foundation of a system. Another reason is that it can have far-reaching consequences, as it can affect multiple users or systems that rely on the vulnerable design.

It is important for organizations to prioritize addressing insecure design vulnerabilities and to incorporate security considerations into the design process to prevent vulnerabilities from being built into a system. This includes conducting security assessments and reviews, implementing secure coding practices, and regularly testing and updating systems to ensure that they are secure. Ignoring insecure design vulnerabilities can have severe consequences, including financial losses, damage to reputation, and legal liabilities.

For possible solutions, adding unit tests to ensure the security of each component. Adding integrational tests to ensure the security of more than one component. Design anti-bot isn’t possible at this stage due to time and it is the end of the project time.

A05: Security Misconfiguration

Security misconfiguration is a security vulnerability that occurs when a system is not properly configured to protect against attacks. This type of vulnerability can have a low likelihood of occurrence, as it often requires a high level of technical expertise and specific knowledge of a system's configuration to exploit. However, the impact of security misconfiguration can be moderate, as it can allow attackers to gain access to sensitive data or resources, or to modify system configurations.

There are several common causes of security misconfiguration, including using default configurations, failing to secure configuration files, and not properly configuring access controls. It is important for organizations to prioritize addressing security misconfiguration vulnerabilities and to properly configure their systems to prevent attackers from exploiting them. This includes setting strong passwords, securing configuration files, and implementing proper access controls to prevent unauthorized access.

Ignoring security misconfiguration vulnerabilities can have consequences, including financial losses and damage to reputation. However, the risk of these consequences may be low if the likelihood of the vulnerability being exploited is also low. It is important for organizations to carefully assess the likelihood and impact of security vulnerabilities and to prioritize addressing those that pose the greatest risk.

For possible solutions, using docker to auto deploy the application will help with making sure that the configuration is set in the right method.

A06: Vulnerable and Outdated Components

Vulnerable and outdated components are security vulnerabilities that occur when a system uses components that have known vulnerabilities or that are no longer supported by their developers. These types of vulnerabilities can have a low likelihood of occurrence, as they often require a high level of technical expertise and specific knowledge of a system's components to exploit. However, the impact of vulnerable and outdated components can be moderate, as they can allow attackers to gain access to sensitive data or resources, or to modify system configurations.

One common cause of vulnerable and outdated components is the use of software or libraries that are no longer supported or maintained by their developers. These components may have known vulnerabilities that have not been fixed, making them easier for attackers to exploit. Another cause is the failure to regularly update components, which can leave a system vulnerable to newly discovered vulnerabilities.

It is important for organizations to prioritize addressing vulnerable and outdated components and to regularly update and patch their systems to prevent attackers from exploiting them. This includes staying up-to-date with the latest security patches and updates, and regularly reviewing and replacing components that are no longer supported or secure. Ignoring vulnerable and outdated components can have consequences, including financial losses and damage to reputation. However, the risk of these consequences may be low if the likelihood of the vulnerability being exploited is also low. It is important for organizations to carefully assess the likelihood and impact of security vulnerabilities and to prioritize addressing those that pose the greatest risk.

A07: Identification and Authentication Failures

Identification and authentication failures refer to instances where a person or system is unable to properly identify or authenticate a user or device. These types of failures can have a moderate impact and a moderate likelihood, depending on the specific context and circumstances.

In terms of motivation, it is important to address identification and authentication failures because they can compromise the security and integrity of a system. If a person or device is able to gain unauthorized access to a system, they may be able to access sensitive information or disrupt the normal functioning of the system. This can lead to financial losses, reputational damage, or other negative consequences for the organization or individuals involved.

There are various ways to mitigate the risk of identification and authentication failures. Some common approaches include implementing strong password policies, using two-factor authentication, and regularly updating and patching system vulnerabilities. It is also important to regularly train and educate users on the importance of proper identification and authentication practices. By taking proactive measures to address these issues, organizations can better protect themselves and their users from the potential impacts of identification and authentication failures.

For possible solutions, notifying the admin incase of attacks, and adding application session timeout are good solutions to ensure Identification and Authentication Failures, however these are accepted risks in this project due to the scope of the project.

A08: Software and Data Integrity Failures

Identification and authentication failures refer to instances where a person or system is unable to properly identify or authenticate a user or device. In cases where the likelihood of these failures is low, the impact and risk may still be moderate due to the potential consequences of a successful identification or authentication failure.

One example of a low likelihood, moderate impact, moderate risk identification and authentication failure might be a scenario where an organization has implemented strong security measures to prevent unauthorized access to its systems, but a user's credentials are accidentally compromised through a phishing attack. In this case, the likelihood of the failure is low due to the organization's strong security measures, but the impact and risk could be moderate if the attacker is able to gain access to sensitive information or disrupt the normal functioning of the system.

To mitigate the risk of identification and authentication failures, it is important for organizations to implement robust security measures and regularly train and educate users on how to identify and prevent potential attacks. By taking a proactive approach to security, organizations can reduce the likelihood of identification and authentication failures and better protect themselves and their users from the potential impacts of these failures.

For possible solutions, ensuring that the data is sent to secure clients is one of the solutions, however this solution isn’t possible at this stage of the project, therefore this solution won’t be implemented.

A09: Security Logging and Monitoring Failures

Security logging and monitoring failures refer to instances where an organization's security logging and monitoring systems are unable to properly function or detect potential security threats. These types of failures can have a high impact and a maximum likelihood, depending on the specific context and circumstances.

In terms of motivation, it is important to address security logging and monitoring failures because they can compromise the security and integrity of a system. If an organization's security logging and monitoring systems are not functioning properly, it may be unable to detect and respond to security threats in a timely manner. This can lead to significant financial losses, reputational damage, or other negative consequences for the organization.

There are various ways to mitigate the risk of security logging and monitoring failures. Some common approaches include regularly testing and maintaining security logging and monitoring systems, implementing robust security policies and procedures, and regularly training and educating users on the importance of proper security practices. By taking proactive measures to address these issues, organizations can better protect themselves and their users from the potential impacts of security logging and monitoring failures.

A10: Server Side Request Forgery

Server-side request forgery (SSRF) is a type of cyber attack in which an attacker manipulates a server to send unauthorized requests to other servers or systems on behalf of the server. In cases where the likelihood of an SSRF attack is low, the impact of the attack may still be severe due to the potential consequences of a successful attack.

One example of a low likelihood, severe impact, moderate risk SSRF attack might be a scenario where an organization has implemented strong security measures to prevent unauthorized access to its systems, but a server is compromised through a vulnerability that was not previously known. In this case, the likelihood of the attack is low due to the organization's strong security measures, but the impact could be severe if the attacker is able to gain access to sensitive information or disrupt the normal functioning of the system.

3. Conclusion

The security aspect is one of the most important for applications, therefore for the OWASP top 10, it will be advisable to fix the highest risk and finish what is already planned. However the application won’t be used for public use, therefore the risks are acceptable.