OWASP REPORT

"Playvera"

Ivan Ovcharov
Fontys University of Applied Sciences
S3-CB-01

	Likelyhood	*Impact*	*Risk*	*Actions*	*Planned*
A01:2021-Broken Access Control	*High*	*Severe*	*High*	*Denying force browses and functionality by default*	*N/A*
A02:2021-Cryptographic Failures	*Low*	*Low*	*Low*	*Making sure that the algorithms used are up to date*	*N/A, Fixed*
A03:2021-Injection	*Not Likely*	*Moderate*	*Very Low*	*Whitelist server-side input, safe API's and using ORMS*	*N/A, Fixed*
A04:2021-Insecure Design	*Low*	*Moderate*	*Low*	*Unit/integration testing, separating layers between their importance.*	*Fixing unit testing*
A05:2021-Security Misconfiguration	*High*	*Severe*	*Low*	*Security headers, avoiding the use*	*Possibly switching to Autho*

Ivan Ovcharov
Fontys University of Applied Sciences
S3-CB-01

				of localstorage for storing JWT's	
A06:2021-Vulnerable and Outdated Components	*Low*	*Moderate*	*Low*	*Removing unused dependencies, components and files. Obtaining tools from official sources.*	*N/A*
A07:2021-Identification and Authentication Failures	*Likely*	*Moderate*	*Moderate*	*Multi-factor authentication, weak password check, session creator.*	*N/A*
A08:2021-Software and Data Integrity Failures	*Low*	*Severe*	*Moderate*	*Ensuing libraries and dependencies are from trusted repositories. Ci/cD has proper segregation.*	*N/A*
A09:2021-Security Logging and Monitoring Failures		*Severe*	*Moderate*	*Making sure that suspicious*	*N/A*

				or malicious accounts are held for enough time to be fixed. All login and access control can be logged	
A10:2021-Server-Side Request Forgery	*High*	*Severe*	*Moderate*	*Data sanitization, disabling HTTP redirections, no raw responses to clients.*	*N/A*

Ivan Ovcharov
Fontys University of Applied Sciences
S3-CB-01

Project specification

A01:2021-Broken Access Control

Broken access control can be exploited if the functionality of managing “server roles” is exploited. Furthermore, users may use external programs to breach into forbidden server paths and exploit sensitive information. Playvera shouldn’t face path exploiting, however roles may be bypassed if a software (such as Postman) is used.

A02:2021-Cryptographic Failures

Cryptographic failures are when the cause is not described, leading to data exposure and the inability to find the source of the problem. The project has exception handling and pointers to possible security breaches, however it could still be bypassed as not everything is fully described.

A03:2021-Injection

Injection is when a third party software is used to “inject” data into the application, potentially alternating sensitive functionality and breaching into forbidden data storages.

Ivan Ovcharov
Fontys University of Applied Sciences
S3-CB-01

A04:2021-Insecure Design

Insecure design describes using older implementations or libraries to create features that may be exploited, leading to potential security breakthroughs. Playvera uses the latest implementations of third-party additions and packages.

A05:2021-Security Misconfiguration

Potentially exposing user roles or allowing security components to be injected could be fatal to the application. Furthermore, test users or admin profiles are possible to be seized and used in a malicious way to alter the application’s information/configuration.

A06:2021-Vulnerable and Outdated Components

Outdated components that use older or unauthorized sources can be an information breach source. Playvera is using the latest implementations and gets everything from licensed resources.

A07:2021-Identification and Authentication Failures

Weak and easy to decrypt passwords partnered with keeping a session id not refreshed can lead to a breach into potential user information leak. The application uses an algorithm for encrypting passwords as soon as they are initialized, however it may suffer from token bypassing as it is stored locally in the browser cache.

A08:2021-Software and Data Integrity Failures

Similar to the A06:2021, not checking the integrity of the software is a huge risk that can lead to total information exposure. Using licensed software to check different libraries and additions is crucial. Playvera may not suffer from such risks, as all of the components used to build the application are from verified sources and up to date.

A09:2021-Security Logging and Monitoring Failures

Identifying malicious users and activity done fast is crucial in order to prevent a large amount of information being exposed. Playvera uses token refreshing every 30 minutes, making the user to verify their integrity by re-logging, however, this could be improved with sessions.

A10:2021-Server-Side Request Forgery

Making unauthorized http requests (potentially posting data that may break into the security architecture, etc.) is a very common attack that applications have to deal with. Playvera uses JWT's to avoid such attacks, however, due to their storage in the local cache of the user, this could be easily altered.

Ivan Ovcharov
Fontys University of Applied Sciences
S3-CB-01