Security Report

Name: Mitran Matei-Cristian

Student Number: 4207734

A1:2017-Injection

This consists of NoSQL injection (because I use MongoDB). In my application, I use backend input validation and a safe API using parameterized interfaces to prevent this.

A2:2017-Broken Authentication

This consists of using “brute force” (using millions of different usernames and passwords) to gain access to an administrator account and compromise the website. In my application, I implemented weak passwords check and no default credentials for administrator accounts to prevent this.

A3:2017-Sensitive Data Exposure

This consists of encrypting sensitive data poorly, thus exposing sensitive information to hackers. In my application, I use the predefined BCryptPasswordEncoder to prevent this.

A4:2017-XXE

This consists of a user uploading malicious XML files to compromise dependencies, integrations, or weak code. In my application, I use less complex data formats such as JSON, and avoiding serialization of sensitive data.

A5:2017-Broken Access Control

This consists of hackers using tools to exploit unfunctional access control configurations. In my application, I make sure the users cannot modify the access control or metadata.

A6:2017-Security Misconfiguration

This consists of hackers trying to access unused pages or unprotected files or directories. In my application, I do not have any unnecessary features/components/frameworks and the security configuration is automatized to minimize the effort in securing the environment.

A7:2017-XSS

This consists in hackers using automated tools to exploit all three forms of XSS (PHP, J2EE/ JSP, ASP.NET). In my application, I use ReactJS as my frontend framework, which is one of the frameworks that escapes XSS by design.

A8:2017-Insecure Deserialization

This consists in hackers hiding malicious data in a series of objects. In my application, I use a logger to monitor every request and users are not permitted to spam deserialize.

A9:2017-Using Components with Known Vulnerabilities

This consists in using unnecessary vulnerable

dependencies/files/components/etc. that have public exploits available. In my application, there are no unnecessary components and all the components used are from official sources and signed packages.

A10:2017-Insufficient Logging/Monitoring

This consists in poorly logging actions happening therefor not monitoring and not knowing what is going on. In my application, every controller has a logger as well as every service.