Security & GDPR

Cristin Rusnac

Fontys University of Applied Science

# OWASP TOP 10

## A01 Broken Access Control

Any access is granted only through a valid session, only an authenticated and authorized user can access sensitive information and only in the boundaries of own permission. An invalid session will result in having no permission to use certain features of the system.

## A02 Cryptographic Failures

Data exposure is handled through encryption, in the perspective of the existing system the only sensitive data would be the authentication credentials of the user, which are encrypted and stored in an encrypted form inside of the database. The credentials are not shared among other services of the network.

## A03 Injection

Injection is impossible, the databases used within the system are of a NoSQL type and there are data models implemented for data validation. As well as having versioning for events emitted within the system to ensure the data flow is appropriate.

## A04 Insecure Design

The system was developed in a secure oriented way, there is no client sensitive data exposure to the system, so the client data is only handled by the auth. service. All data is validated using data models. There is data versioning for the data flows within the system, so if a version of data is handled with a delay, all new updates of a record will be set as pending until it is updated according to the versioning. Regarding the JWT and Stripe tokens those are stored as secrets within the k8s cluster and cant be accessed. As well cases like purchase of the same product by two different users simultaneously is handled through reservation, so every time a user intends to make a purchase the product is reserved for a period of time (that is displayed to the user) so the user has to finalize the purchase within that timeframe, otherwise the reservation is cancelled and the product is made available again for purchase.

## A05 Security Misconfiguration

The configuration of the system is implemented so no sensitive data is exposed to any type of user. K8s secrets are used to store tokens related to purchase service and JWT. The error messages are also modeled so it sends an error message and a status code without exposing any information from within the system.

## A06 Vulnerable and Outdated Components

The versioning of all dependencies used by the services were checked if are up to date and not deprecated, there are no conflicts or errors while being compiled. Google Cloud has also a built in service that updates the versions of k8s and docker to the latest stable. There are no unused dependencies or shortcuts left in the system, those are either deleted or commented out and left just for developers as a reference.

## A07 Identification and Authentication Failures

The authorization is done through a session that is granted at signup or signin. If the session token is modified then it wont be validated by the auth. service hence the authorization is terminated. At any request made by the user the session token is validated and so there is no possibility of unauthorized requests. The token is also stored in an http only cookie that cant be accessed from within the code.

## A08 Software and Data Integrity Failures

Data integrity is validated at any user request using predefined data models, the data is also distributed within the system so that any service has access only to data required by its functionality. The libraries used by the system are installed each time a new deployment is pushed, so if there are any dependency conflicts those will be spotted at redeployment. The user sensitive data is also checked at authentication, so for example the provided password is first validated to the record stored in the database then the authentication is granted. Moreover the client application is built with Next JS and is server side rendered, this means the webapp is static and doesn’t grant direct connection to the system, so no access to sensitive data is granted.

## A09 Security Logging and Monitoring Failures

Google Cloud logs all the events that occur within the system backend wise and also frontend. Also the purchase process is logged by Stripe so its independent and does not interfere with the functioning of the system.

## A10 Server Side Request Forgery (SSRF)

The webapp provided to end user is static due to using server side rendering, all client interaction are validated through session validation and data format validation. All urls do not contain any sensitive data, the maximum would be the id of the product that is viewed or purchased, but until the purchase process is not over the status of the product cant be modified.

# GDPR

## Lawful, fair and transparent processing

The data processed by the system do not contain any personal data of the user, the only processed data is the email of the user and password in an encrypted form and is only handled by the authentication service without being available to other services. Despite having a payment function the system itself does not handle any Credit Card information, instead the payment process is handled by a 3rd party service.

## Limitation of purpose, data, and storage

The purpose of user data stored within the system is only authentication and authorization, it is only handled by the auth. service. Any other data like footwear information and order information is not of sensitive type and is also only handled within the system.

At this stage no personal data is required for the functionality of the system, at a later stage when more features and functions are implemented some more data may be required from the user like the delivery address of the product, which will be handled just by the system, and can be removed by the user at any time.

## Consent

No sensitive data is handled by the application so the only consent of data provided are the authentication credentials which are provided by the user at the signup.

## Privacy by design

Privacy and security is ensured through encryption, the proper distribution of data, so no personal data is shared with other services within the system as well as a session based authorization.

## Data protection and impact assessment

Not Applicable

## Data transfers

There is no user data transfer within the system, except the ‘userId’. In the perspective of payment process the system uses Stripe, the only information provided to it is the userId, and amount, Stripe key is stored as a secret inside the cluster.

## Data protection officer

Not Applicable

## Awareness and training

I have studied the GDPR regulations and also applied them previously, during the internship since I developed a work logging plugin and had to agree on which data the employer is allowed to gather regarding the employee.

## Data subject rights

Each user has access to own data and orders history as well as the products available for purchase.