Melany Winklaar, Jordy Samson, Jeroen Hoebert & Aaik Oosters

UCloud

Hoe veilig ben jij

Inhoudsopgave

[Inleiding 2](#_Toc532471736)

[A1:2017-Injection 2](#_Toc532471737)

[A2:2017-Broken Authentication 3](#_Toc532471738)

[A3:2017-Sensitive Data Exposure 4](#_Toc532471739)

[A4:2017-XML External Entities (XXE) 5](#_Toc532471740)

[A5:2017-Broken Access Control [Merged] 6](#_Toc532471741)

[A6:2017-Security Misconfiguration 7](#_Toc532471742)

[A7:2017-Cross-Site Scripting (XSS) 8](#_Toc532471743)

[A8:2017-Insecure Deserialization 9](#_Toc532471744)

[A9:2017-Using Components with Known Vulnerabilities 10](#_Toc532471745)

[A10:2017-Insufficient Logging & Monitoring 11](#_Toc532471746)

# Inleiding

Voor het vak security hebben wij de opdracht gekregen om een OWASP top-10 webapplicatie te bouwen. Wij hebben hier ons best voor gedaan en zijn gekomen tot vele verschillende toepassingen om dit project OWASP top-10 valide te krijgen.

Het project is terug te vinden onder de volgende link:

<https://github.com/aaikoosters/blok12-security>

Wij hebben een UCloud gemaakt. Hierin is het mogelijk dat er foto’s ge-upload worden. Deze foto’s zijn voor jou en voor jou alleen. Aan ons de kunst om dit af te handelen en op te lossen.

# A1:2017-Injection

* Gebruik SQL-commando’s om SQL-injection te voorkomen
* Valideer binnenkomend data

# A2:2017-Broken Authentication

* Geen zwakke wachtwoorden toestaan (nog implementeren in devolop)
* ~~Data encrypten voordat verstuurd wordt~~ (toegepast)
* Maximaal aantal login pogingen (nog implementeren in devolop)

# A3:2017-Sensitive Data Exposure

* ~~Geen standard encryptie toepassen. Gebruikt up to date encryptie sleutels en algoritmes~~ (toegepast)

# A4:2017-XML External Entities (XXE)

* Valideren van Xml-bestanden/ Weigeren van toestemming om Xml-bestanden te uploaden

(Wij gebruiken geen XML dus in hoeverre moet dit toegepast worden?)

# A5:2017-Broken Access Control [Merged]

* Zorg da teen specifiek iemand gegevens kunnen aanpassen aan de database Zorg dat alle toegangscontrole gemonitord en gelogd worden
* Invalideer JWT-tokens wanneer gebruikers uitloggen (toegepast?)

# A6:2017-Security Misconfiguration

* Gebruik geen HTTP
* Gebruik scanners om de zwakheden van je applicatie te vinden

# A7:2017-Cross-Site Scripting (XSS)

* Een framework die beschermt tegen XSS
* Input valideren door middel van antiforgerytokens
* Niet vertrouwde http requests niet afhandelen
* Check extensie
* Aan bestand wat voor type
* PHP-methode om formaat photo te vragen
* Als gevalideerd dat foto gemaakt is -> foto bouwen om alle code eruit te filteren.

# A8:2017-Insecure Deserialization

* Logging toevoegen zodat je kan traceren wat er is gebeurt
* Gaat jeroen doen en uitleggen

# A9:2017-Using Components with Known Vulnerabilities

* ~~Gebruik alleen officiële componenten~~ (toegepast)
* ~~Geüpdatet versie van componenten~~ (toegepast)
* We gebruiken geen componenten?

# A10:2017-Insufficient Logging & Monitoring

* Zorg dat logs gegenereerd worden zodat deze logs door een gecentraliseerd logbeheersysteem gebruikt kan worden.
* Het systeem zo monitoren dat er een alarm af gaat wanneer verdacht gedrag wordt gedecreteerd
* Zorg dat er een plan van herstel beschikbaar is indien iets fout is