Ivan Ip

FONTYS ICT SEMESTER 3

Research document

Software development S3 Security

Table of Contents

[Inleiding 2](#_Toc155141730)

[Wat kan een SQL-injectie doen met de applicatie 3](#_Toc155141731)

[Hoe werken aanvallen met SQL-Injectie 4](#_Toc155141732)

[Impact van SQL-injectie aanvallen 5](#_Toc155141733)

[Hoe voorkom je een SQL-injectie aanval 6](#_Toc155141734)

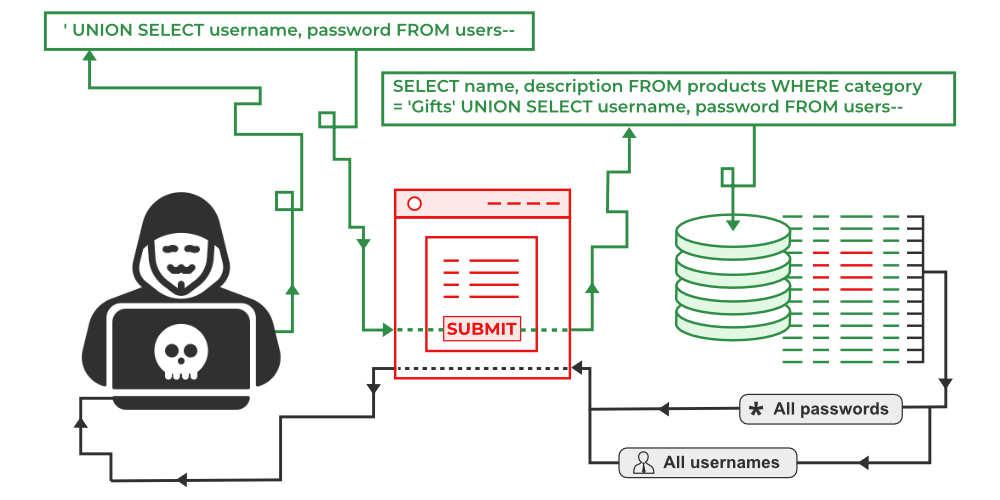
[Bronnen 6](#_Toc155141735)

# Inleiding

Voor de individuele project maak ik gebruik van Osu! API. Met de api kan ik de speler statistieken ophalen en ook de gespeelde matches terugkijken. De bedoeling hiervan is dat spelers gemakkelijk hun statistieken kunnen terug zien en ook de gespeelde matches. Er is namelijk ook geen mogelijkheid om je games terug te zien vandaar dat ik dit project maak. Ook zorg ik ervoor dat ik niet alleen de api zal gebruiken. De data die ik van de api ophaal zal ik continu opslaan in mijn eigen database zodat ik niet afhankelijk ben van de api. Natuurlijk moet mijn website ook beveiligd zijn tegen aanvallen. In dit onderzoek ga ik het over SQL-injectie hebben. Wat is SQL-injectie en hoe kan ik mijn website beveiligen tegen SQL-injecties. Dit soort onderwerpen komen voor in dit onderzoek.

# Wat kan een SQL-injectie doen met de applicatie

Een SQL-injectie, soms afgekort als SLQi is een soort cyberaanval waarbij een hacker een SQL-codefragment (Structured Query Language) gebruikt om een database te manipuleren en toegang te verkrijgen tot mogelijk waardevolle informatie. In mijn project maak ik gebruik van een SQL database waardoor je SQL injectie toe kan passen. Dit heeft erge gevolgen als ik mijn data niet goed bescherm. Het kan zijn dat mijn data wordt gemanipuleerd of zelfs wordt gewist.



# Hoe werken aanvallen met SQL-Injectie

Om SQL-injectie te begrijpen, is het belangrijk om te weten wat structured query language (SQL) is. SQL is een querytaal die wordt gebruikt bij het programmeren om toegang te krijgen tot gegevens opgeslagen in relationele databases en om deze gegevens te wijzigen en te verwijderen. Aangezien de overgrote meerderheid van websites en web-apps afhankelijk is van SQL-databases, kan een SQL-injectieaanval ernstige gevolgen hebben voor organisaties.

Een SQL query is een verzoek dat naar een database - een archief vol informatie op de computer - wordt verstuurd om informatie te verzamelen of om een SQL-code uit te voeren. Een voorbeeld hiervan is inloginformatie die via een webformulier wordt ingestuurd om een gebruiker toegang te geven tot een website. Deze webformulieren zijn meestal ontworpen om alleen specifieke gegevens zoals een naam en/of wachtwoord te accepteren. Als de informatie is ingevoerd, wordt deze gecontroleerd. De gebruiker krijgt toegang als de informatie overeenkomt met de database. Zo niet, wordt hen de toegang ontzegd.

De problemen ontstaan omdat het bij veel webformulieren niet mogelijk is om te voorkomen dat meer informatie wordt ingevoerd. Aanvallers kunnen hier gebruik van maken en invoervelden van het formulier gebruiken om hun eigen verzoeken naar de database te sturen. Dit kan er voor zorgen dat ze verschillende misdadige activiteiten kunnen uitvoeren zoals het stelen van gevoelige gegevens of informatie in de database manipuleren voor hun eigen doeleinden.

# Impact van SQL-injectie aanvallen

Een succesvolle SQL-injectieaanval kan ernstige gevolgen hebben voor een bedrijf. Dit komt omdat een SQL-injectieaanval het volgende kan doen:

* Gevoelige gegevens blootgeven. Aanvallers kunnen gegevens ophalen, waardoor gevoelige gegevens die op de SQL-server opgeslagen zijn, het risico lopen om blootgesteld te worden.
* Gegevensintegriteit beïnvloeden. Aanvallers kunnen informatie van je systeem wijzigen of verwijderen.
* Gebruikersprivacy beïnvloeden. Afhankelijk van de gegevens die op de SQL-server zijn opgeslagen, kan een aanval gevoelige gebruikersinformatie blootleggen, zoals adressen, telefoonnummers en creditcardgegevens.
* Een aanvaller beheerderstoegang geven tot je systeem. Als een databasegebruiker beheerdersrechten heeft, kan een aanvaller toegang krijgen tot het systeem met behulp van kwaadaardige code.

Een aanvaller algemene toegang geven tot je systeem. Als je zwakke SQL-opdrachten gebruikt om gebruikersnamen en wachtwoorden te controleren, kan een aanvaller toegang krijgen tot je systeem zonder de inloggegevens van een gebruiker te kennen. Van hieruit kan een aanvaller grote schade aanrichten met toegang te krijgen tot gevoelige informatie en deze informatie te manipuleren. Ook kan de aanvaller de inhoud van je database te lezen, te wijzigen of zelfs te verwijderen. Dit kan grote gevolgen hebben van mijn applicatie.

# Hoe voorkom je een SQL-injectie aanval

Er is goed nieuws: je kunt vandaag al maatregelen nemen om een SQL injectie aanval te voorkomen. Het is namelijk zo dat het merendeel van de SQL injectie kwetsbaarheden snel kan gevonden worden.

* Een kwetsbaarheidsscanner gebruiken

Een optie is om een ‘web kwetsbaarheden scanner’ te gebruiken. Je kunt deze scanner gebruiken door tests op te zetten tegen elke ingang van de applicatie. Via deze weg kun je zoeken naar fouten, systematische verschillen en de antwoorden die je op de applicatie krijgt.

* Geparameteriseerde query’s gebruiken

SQL-injectie aanvallen komen voor doordat door de gebruiker geleverde invoer wordt gebruikt in SQL-queries zonder dat deze eerst wordt gefilterd. Een manier om dit te voorkomen is het gebruik van queries met parameters - dat wil zeggen queries die variabelen gebruiken in plaats van constanten in de querystring. In mijn applicatie heb ik geparameteriseerde query’s gebruikt zodat mijn applicatie beveiligd is tegen SQL-injecties.

# SQL injection in zijn werk

# Bronnen

<https://www.kaspersky.nl/resource-center/definitions/sql-injection>

<https://www.one.com/nl/website-beveiliging/wat-is-sql-injection>

<https://nordvpn.com/nl/blog/sql-injectie/>

<https://www.avg.com/nl/signal/sql-injection>