Cybersecurity & Warfare – Week 2 – Top 10 Owasp

[**A01:2021-Broken Access Control**](https://owasp.org/Top10/A01_2021-Broken_Access_Control/)

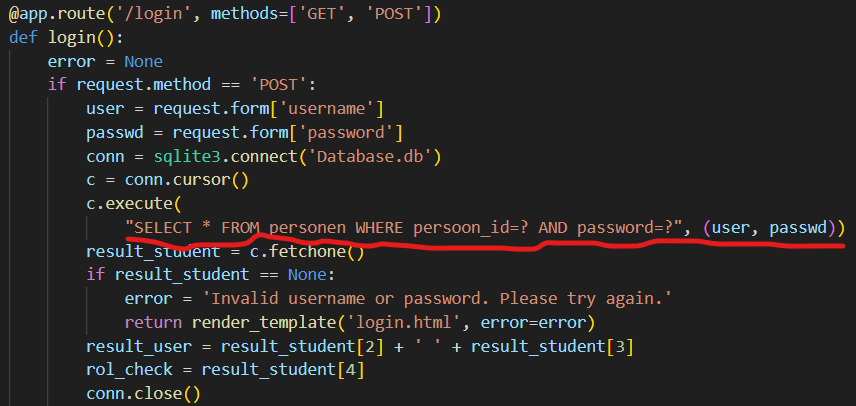
Dit probleem is toentertijd opgelost doormiddel van Approutes die gekoppeld zijn voor elke gebruiker. Eerst wordt er gecheckt wat voor rol de gebruiker heeft wanneer hij inlogt, dit kan een student, teacher en admin zijn. In de onderstaande tabel wordt aangegeven welke rechten elke gebruiker heeft.

|  |  |  |  |
| --- | --- | --- | --- |
| **Toegang** | **Student** | **Teacher** | **Admin** |
| Homestudent (Home page voor de studenten) | ✔ | X | ✔ |
| Calendarstudent (de kalender voor de studenten) | ✔ | X | ✔ |
| Personen opzoeken | ✔ | ✔ | ✔ |
| Personen toevoegen aan de database | X | ✔ | ✔ |
| Personen verwijderen uit de database | X | ✔ | ✔ |
| Hometeacher(Home page voor de docenten) | X | ✔ | ✔ |
| Calendarteacher( de kalender voor de docenten) | X | ✔ | ✔ |
| Evenement aanmaken op de kalenders | X | ✔ | ✔ |

Wanneer een niet geautoriseerd persoon probeert in te loggen op de homepagina zal hij de melding ‘’Invalid username or password. Please try again.’’ En wanneer deze probeert toegang te krijgen doormiddel van het aanpassen van de url wordt de persoon weer teruggestuurd naar het inlog scherm. Dit komt omdat er niet geverifieerd is welke rol deze persoon heeft en hierdoor kan deze niet zonder toestemming op een pagina komen waar normaliter rechten voor nodig zijn.

[**A03:2021-Injection**](https://owasp.org/Top10/A03_2021-Injection/)

Binnen werkplaats 3 hebben wij gebruik gemaakt van prepared statements in onze query’s



i.p.v. de inlog gegevens te hard coden haalt hij de data uit onze database doormiddel van de parameters die gegeven zijn in de query.

Ook is er gebruikt gemaakt van een beveiligde sessieopslag. Dit is gedaan door het attribuut  dit zorgt ervoor dat de cookies goed zijn beveiligd hierdoor kunnen aanvallers niet zomaar sessiecookies vervalsen.

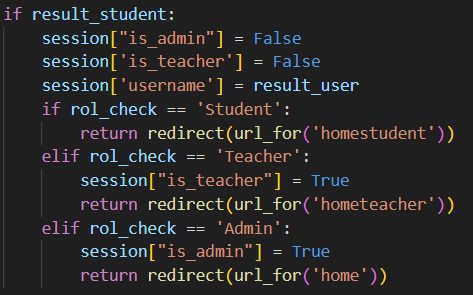
Zoals eerder vermeld bij Broken Access Control wordt gecheckt wat voor inlog rol de gebruiker heeft dit wordt geverifieert doormiddel van de Rol die het account heeft in de database. Hiermee wordt er dus voor gezorgd dat elke gebruiker alleen bij de functies kan die hem zijn toegewezen.



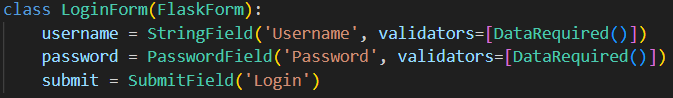


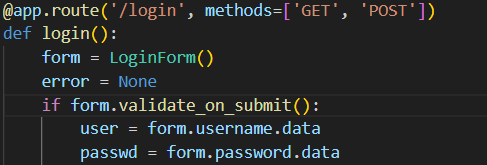


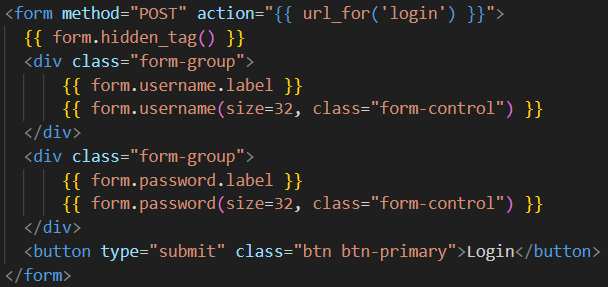




Echter bevat de code geen goede invoervalidatie. Dit heb ik opgelost doormiddel van ‘WTForms’ toe te passen. Hierdoor wordt er bij de input van de gebruiker gekeken of het gegevenstype ook overeenkomt met het daadwerkelijke verwachte gegevenstype. Hierdoor kunnen SQL injecties die gebruikmaken van verkeerde gegevenstype tegen worden gehouden.



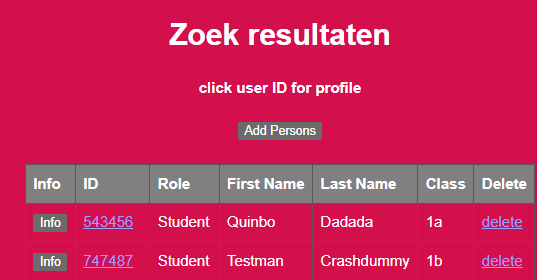




[**A06:2021-Vulnerable and Outdated Components**](https://owasp.org/Top10/A06_2021-Vulnerable_and_Outdated_Components/)

Na het grondig doorzoeken van de code werkplaats 3 ziet het er naar uit dat de search pagina nogal kwetsbaar is. De ‘’Add persons’’ en de ‘info’ buttons functioneren allebei zoals het hoort.

Echter de ‘’Go Back’’, ‘’Back to Home’’ en ‘’delete’’ button zijn verkeerd gerouted.





Wanneer deze buttons worden gebruikt stuurt hij je door naar de add persons pagina. Dit geeft gebruikers met de rol studenten de mogelijkheid om nieuwe studenten toe te voegen aan de database en ook te verwijderen terwijl dit niet binnen hun bevoegdheid valt.

Dit is op te lossen doormiddel van betere routing en meerdere html search pagina’s te maken voor elke rol.