|  |
| --- |
| WTIS\_  Security |

Daniek Roelofs(1675315)

24 januari 2025

INHOUDSOPGAVE

[1 Inleiding 3](#_Toc188643657)

[2 casus 3](#_Toc188643658)

[3 Risico analyse 4](#_Toc188643659)

[3.1 Injection 4](#_Toc188643660)

[3.1.1 Risicotabel 4](#_Toc188643661)

[3.1.2 Veiligheidsdiagram 5](#_Toc188643662)

[3.2 Broken authentication 6](#_Toc188643663)

[3.2.1 Risicotabel 6](#_Toc188643664)

[3.2.2 Authenticatie session hijacking 6](#_Toc188643665)

[3.3 XSS (Cross-Site Scripting) 7](#_Toc188643666)

[3.4 CSRF (Cross-Site Request Forgery) 8](#_Toc188643667)

[3.5 Onvoldoende toegangscontrole 9](#_Toc188643668)

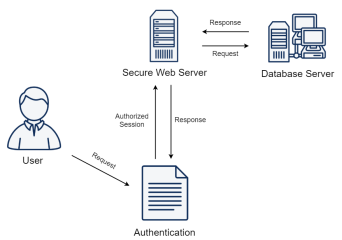
[4 codeVoorbeelden 10](#_Toc188643669)

# Inleiding

Het beroepsproduct voor Web Technology: Implementation & Security (WTIS) bouwt voort op het vorige project (WTUX), dit keer als een volledig functionele webapplicatie. Deze applicatie, ontworpen voor Gelre Airport, moet gebruik maken van HTML, CSS, PHP en een SQL Server.

# casus

Pizzeria Sole Machina wil een webapplicatie waarmee klanten online bestellingen kunnen plaatsen en personeel deze bestellingen efficiënt kan beheren. Klanten kunnen een account aanmaken, het menu bekijken, items aan hun winkelmandje toevoegen en bestellingen plaatsen met een afleveradres. Personeel kan ingelogde bestellingen inzien, de status bijwerken en de voortgang van bestellingen beheren.



# Risico analyse

Gekozen Top 5 OWASP risico's:

* Injection
* Broken authentication
* XSS (Cross-Site Scripting)
* CSRF (Cross-Site Request Forgery)
* Onvoldoende toegangscontrole

## Injection

### Risicotabel

|  |  |  |
| --- | --- | --- |
| Risico | Gevolgen bij doorbraak | Beveiligingmaatregelen |
| Injection | Ongeautoriseerde toegang tot gegevens, manipulatie van database | Gebruik van prepared statements en parameterized queries om SQL-injecties te voorkomen |

Gevolgen bij doorbraak:

Bij een succesvolle injectie-aanval kan een aanvaller ongeautoriseerde toegang krijgen tot de database, gegevens manipuleren of zelfs volledige controle over de server verkrijgen.

Beveiligingsmaatregelen:

De applicatie gebruikt prepared statements en parameterized queries om het risico op SQL-injecties te minimaliseren. Zie hoofdstuk voorbeelden.

### Veiligheidsdiagram

Afbeelding met tekst, schermopname, diagram, ontwerp

Automatisch gegenereerde beschrijving

## Broken authentication

### Risicotabel

|  |  |  |
| --- | --- | --- |
| Risico | Gevolgen bij doorbraak | Beveiligingmaatregelen |
| Injection | Ongeautoriseerde toegang tot gegevens, manipulatie van database | Gebruik van prepared statements en parameterized queries om SQL-injecties te voorkomen |

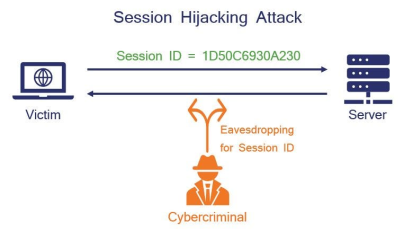
Gevolgen bij doorbraak:

Ongeautoriseerde toegang tot gebruikersaccounts kan leiden tot gegevensdiefstal, manipulatie en verlies van vertrouwen van gebruikers.

Beveiligingsmaatregelen:

De applicatie implementeert sterke wachtwoordbeleid. Wachtwoorden worden gehasht opgeslagen met php password hash (password\_default).

### Authenticatie session hijacking



## XSS (Cross-Site Scripting)

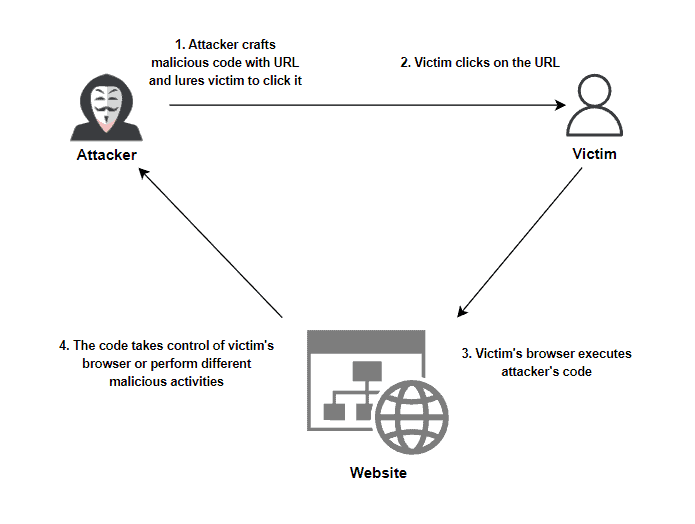
|  |  |
| --- | --- |
| Risico | R2: Cross-Site Scripting (XSS) in invoervelden en databaseweergaven |
| Aanvalstechniek | Invoeren van kwaadaardige JavaScript-code in velden zoals gebruikersnaam, adres of wachtwoord. |
| Kans | Gemiddeld: vaak niet direct zichtbaar voor gebruikers maar kan ernstig misbruikt worden. |
| Gevolg | Hoog: Aanvallers kunnen sessiegegevens stelen, kwaadaardige scripts uitvoeren of gebruikers omleiden naar nepwebsites. |

Gevolgen van een XSS-aanval:

* Gebruikersgegevens kunnen worden gestolen door kwaadaardige scripts.
* Defacements en nepformulieren kunnen op de website worden getoond.
* Sessies kunnen worden overgenomen en misbruikt voor identiteitsdiefstal.

Beveiligingsmaatregelen

* Alle uitvoer naar HTML wordt gesanitized met htmlspecialchars() om kwaadaardige scripts te neutraliseren.



## CSRF (Cross-Site Request Forgery)

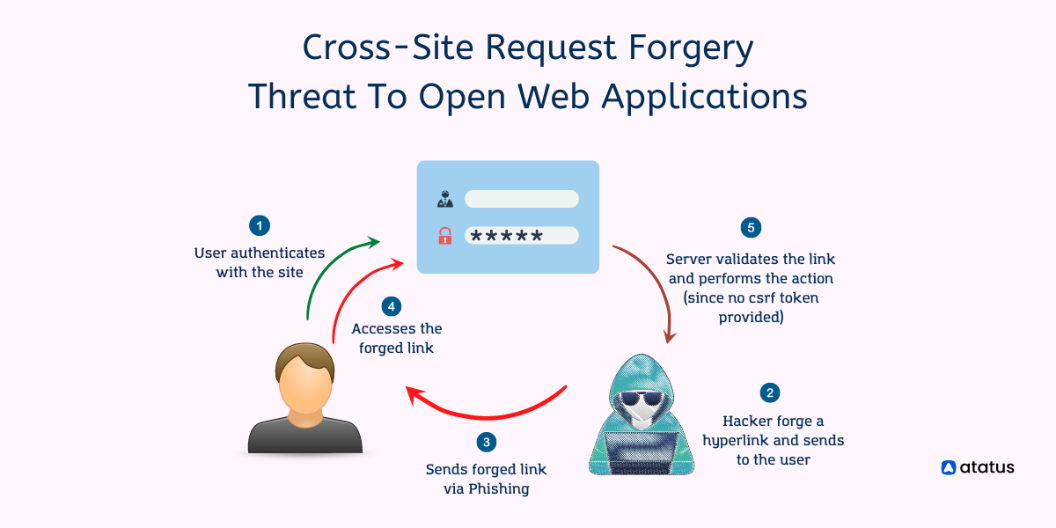
|  |  |
| --- | --- |
| Risico | R3: CSRF-aanvallen op formulieren zoals statusupdates en bestellingen |
| Aanvalstechniek | Vervalsen van verzoeken waardoor een gebruiker zonder zijn medeweten acties uitvoert. |
| Kans | Gemiddeld: vereist vaak sociale engineering, maar kan ernstige gevolgen hebben. |
| Gevolg | Hoog: een aanvaller kan zonder toestemming een bestelling plaatsen of een status wijzigen. |

Gevolgen van onvoldoende toegangscontrole:

* Een klant kan andermans bestellingen bekijken of wijzigen.
* Een personeelslid kan onbedoelde wijzigingen uitvoeren zonder de juiste rechten.
* Een niet-ingelogde gebruiker kan zonder account acties uitvoeren.

Beveiligingsmaatregelen:

* Controle van sessies en rollen op elke pagina.
* Personeel heeft geen toegang tot klantpagina’s en andersom.



## Onvoldoende toegangscontrole

|  |  |  |
| --- | --- | --- |
| Risico | R4: Onvoldoende toegangscontrole voor bestellingen en adminpagina’s | |
| Aanvalstechniek | | Manipuleren van pagina-aanroepen door URL's te wijzigen (bijv. naar bestellingen van andere klanten). |
| Kans | Hoog: veel webapplicaties vertrouwen te veel op client-side beveiliging. | |
| Gevolg | Hoog: onbevoegde toegang tot bestellingen en klantgegevens. | |

Gevolgen van onvoldoende toegangscontrole:

* Een klant kan andermans bestellingen bekijken of wijzigen.
* Een personeelslid kan onbedoelde wijzigingen uitvoeren zonder de juiste rechten.
* Een niet-ingelogde gebruiker kan zonder account acties uitvoeren.

Beveiligingsmaatregelen:

* Controle van sessies en rollen op elke pagina.
* Personeel heeft geen toegang tot klantpagina’s en andersom.

# codeVoorbeelden

Afbeelding met tekst, schermopname, software

Automatisch gegenereerde beschrijving

Deze code voert het inlogproces uit voor medewerkers:

1. Inlogverzoek: Controleert of het inlogformulier is ingediend.
2. Databaseverbinding: Maakt verbinding met de database.
3. Inloggegevens: Haalt gebruikersnaam en wachtwoord op uit het formulier.
4. SQL-query: Voert een voorbereide query uit om de gebruiker op te zoeken.
5. Resultaat verifiëren: Controleert het wachtwoord met password\_verify.
6. Succesvolle login: Stelt de sessie in en stuurt door naar de Homepagina.
7. Mislukte login: Toont een foutmelding bij een fout wachtwoord.

Afbeelding met tekst, schermopname, Lettertype, lijn

Automatisch gegenereerde beschrijving

Deze code controleert de aanwezigheid van een URL-parameter:

1. Controle: Controleert of de ingelogd en rol parameter is ingesteld in $\_SESSION.
2. Geen gegevens: Als de parameter ontbreekt, wordt 'geen gegevens' weergegeven.
3. Beëindigen: Het script stopt de uitvoering met exit().



Het gebruikmaken van password\_hash zodat de wachtwoorden veilig worden opgeslagen bij het registreren.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Het gebruikmaken van password\_verify bij de login.

Dit voorkomt dat wachtwoorden als platte tekst in de database worden opgeslagen en maakt brute-force aanvallen moeilijker.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Toegangscontrole in de navbar om te voorkomen dat de gebruikers toegang hebben tot pagina’s die niet voor hun zijn bedoelt.

Afbeelding met tekst, Lettertype, schermopname, Graphics

Automatisch gegenereerde beschrijving

Deze code laadt dynamisch alle pagina's:

1. Controleer URL-parameter: Controleert of de pagina parameter is ingesteld in $\_GET.
2. Actie: Als de parameter is ingesteld, laadt de code de opgegeven pagina met include $\_GET['pagina'] . ".php".
3. Fallback: Als de pagina parameter niet is ingesteld, laadt de code standaard HomeView.php



Htmlspecialchars zorgt ervoor dat speciale HTML-tekens (zoals <, >, &) worden omgezet naar veilige entiteiten, waardoor XSS-aanvallen worden voorkomen.