2023 - 2024

|  |
| --- |
| ShopMore app |
| Pentest ShopMore webapp |
| Daan Sterckx |

Inhoud

**1. beschrijving app**

**2. onderdelen app**

**3. observaties en** **tips**

**4. gevonden zwakheden**

**5. geteste onderdelen**

**6. ernst van gevonden zwakheden**

**7. privacy**

**8. conclusie**

# Beschrijving app

ShopMore is een webapplicatie die een eenvoudige en gestroomlijnde benadering biedt voor online winkelen. Met minimale styling en een beperkt aantal pagina's richt het zich op essentiële functionaliteiten om gebruikers een directe en heldere winkelervaring te bieden.

De webapp bevat basispagina's zoals een startpagina, productpagina’s en een login pagina. De nadruk ligt op functionaliteit boven visuele aantrekkingskracht, met een gebruiksvriendelijke interface om producten te zoeken en te bekijken.

**Onderdelen app**

Home: De homepagina heeft niet veel inhoud daarom zijn hier ook niet veel zwakheden te vinden

Privacy: op de privacy pagina word de privacy policy gedeelt op deze pagina zullen er ook niet echt zwakheden gevonden kunnen worden maar het is wel een zeer belangrijke pagina omdat privacyregelementen belangrijker en belanglijker worden in een maatschappij waar er elke dag minder privacy is

Products: op deze pagina vind je een aantal producten die je kan aankopen via de webapp. Dit is hoogstwaarschijnlijk verbonden met een database dus is het belangrijk om op te passen voor sql injection en het vertonen van gevoelige informatie in de database

Catagories: op deze pagina vind je een aantal catageriën van producten. Dit is hoogstwaarschijnlijk verbonden met een database dus is het belangrijk om op te passen voor sql injection en het vertonen van gevoelige informatie in de database

Contact: op deze pagina kunnen mensen de eigenaars van de webapp contacteren. Hier kunnen mogelijke zwakheden in zitten aangezien er input velden zijn er moet dus zeker aandacht besteed worden aan het nakijken wat er in deze input velden ingegeven kan worden

Login: op deze pagina kunnen mensen inloggen of gebruikers accounts aanmaken

### Observaties en tips

Wachtwoorden: De wachtwoorden die gebruikt werden voor alle accounts op de webapp zijn zeer slecht en zouden enorm snel gebruteforced kunnen worden. Ik raad aan deze zo snel mogelijk te veranderen.

Content security headers: Sommige paginas hebben geen content security headers deze headers zorgen ervoor dat er geen data op paginas getoont word die niet getoont mag worden zonder deze header kunnen hackers vaak data injection of xxs attacks uitvoeren.

### Gevonden zwakheden

xxs op contact pagina: wanneer ik bijvoorbeeld </script><script>alert(“hello”)</script> dan kan ik de pagina een alert laten sturen met de text hello. Dit gaat omdat er niet gecontroleert wordt welke tekst er ingegeven word op de contact pagina. Daarom kan ik het script van de website sluiten met </script> en dan mijjn eigen script in geven

risico:

Deze kwetsbaarheid, bekend als Cross-Site Scripting (XSS), vormt een aanzienlijk risico voor de integriteit, vertrouwelijkheid en beschikbaarheid van de gegevens op de website.

Het ontbreken van adequate invoervalidatie en -filtering op de contactpagina stelt kwaadwillende gebruikers in staat om schadelijke scripts in te voeren en uit te voeren in de webbrowser van andere gebruikers die dezelfde pagina bezoeken. Dit maakt het mogelijk om ongewenste code, zoals JavaScript, in te voegen die vervolgens wordt uitgevoerd wanneer andere gebruikers de pagina bezoeken. Een voorbeeld hiervan is het invoeren van een script zoals **</script><script>alert(“hello”)</script>**, waardoor een ongewenste pop-up met de tekst "hello" wordt weergegeven.

Het gevaar van deze kwetsbaarheid is aanzienlijk. Het stelt kwaadwillende actoren in staat om gevoelige informatie te stelen, sessiegegevens te manipuleren, gebruikers om te leiden naar schadelijke websites, of andere acties uit te voeren namens de websitegebruiker. Dit kan leiden tot reputatieschade, verlies van vertrouwen bij gebruikers, financiële schade en het compromitteren van gevoelige gegevens.

Een geslaagde aanval via XSS kan leiden tot de verstoring van de normale werking van de webapplicatie, waardoor de geloofwaardigheid en betrouwbaarheid van de website in het gedrang kunnen komen. Het kan ook resulteren in een afname van de gebruikerservaring en het vertrouwen van klanten in de veiligheid van de website.

Het is van cruciaal belang dat deze kwetsbaarheid onmiddellijk wordt aangepakt door het implementeren van strenge invoervalidatie en -filtering op alle invoervelden en door adequate bescherming te bieden tegen XSS-aanvallen. Door deze maatregelen te implementeren, kan de webapplicatie de integriteit en veiligheid van gegevens verbeteren en het risico op schadelijke aanvallen via de contactpagina aanzienlijk verminderen.

advies:

* **Invoer Validatie:**

Zorg ervoor dat alle invoer die wordt ontvangen via formulieren of gebruikersinteracties wordt gevalideerd en gefilterd. Dit betekent het controleren van speciale tekens, HTML-tags en JavaScript-code, en deze indien nodig te neutraliseren of te verwijderen.

* **Gebruik van Escaping Mechanismen:**

Gebruik ontsnappingsmechanismen zoals HTML-encoding om ervoor te zorgen dat invoer van gebruikers als platte tekst wordt weergegeven en niet als uitvoerbare code.

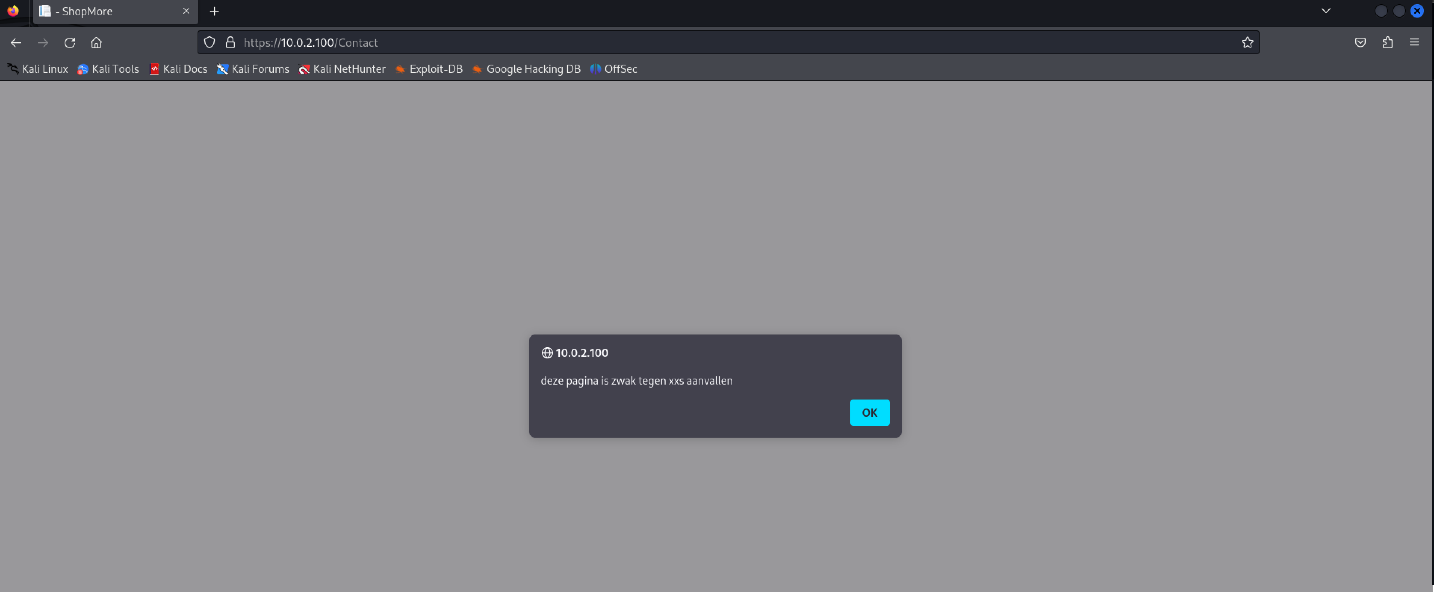
* **Content Security Policy (CSP):**

Implementeer een Content Security Policy die beperkt welke bronnen worden geladen en welk soort code kan worden uitgevoerd, om XSS-aanvallen te voorkomen.

* **Input Sanitization Libraries:**

Gebruik betrouwbare bibliotheken voor invoer-sanitization die specifiek zijn ontworpen om XSS-aanvallen te voorkomen.

bewijs:



Vertical privilege escalation: Ik kan zonder mij in te loggen naar de /admin pagina surfen daar kan ik alle gebruikers hun namen en wachtwoorden zien en kan ik nieuwe gebruikers aanmaken

risico:

Het risico van "Vertical Privilege Escalation" (verticale privilege-escalatie) in een webapplicatie is aanzienlijk en kan leiden tot ernstige inbreuken op de vertrouwelijkheid van gegevens, beveiligingslekken en potentiële misbruiken door ongeautoriseerde gebruikers.

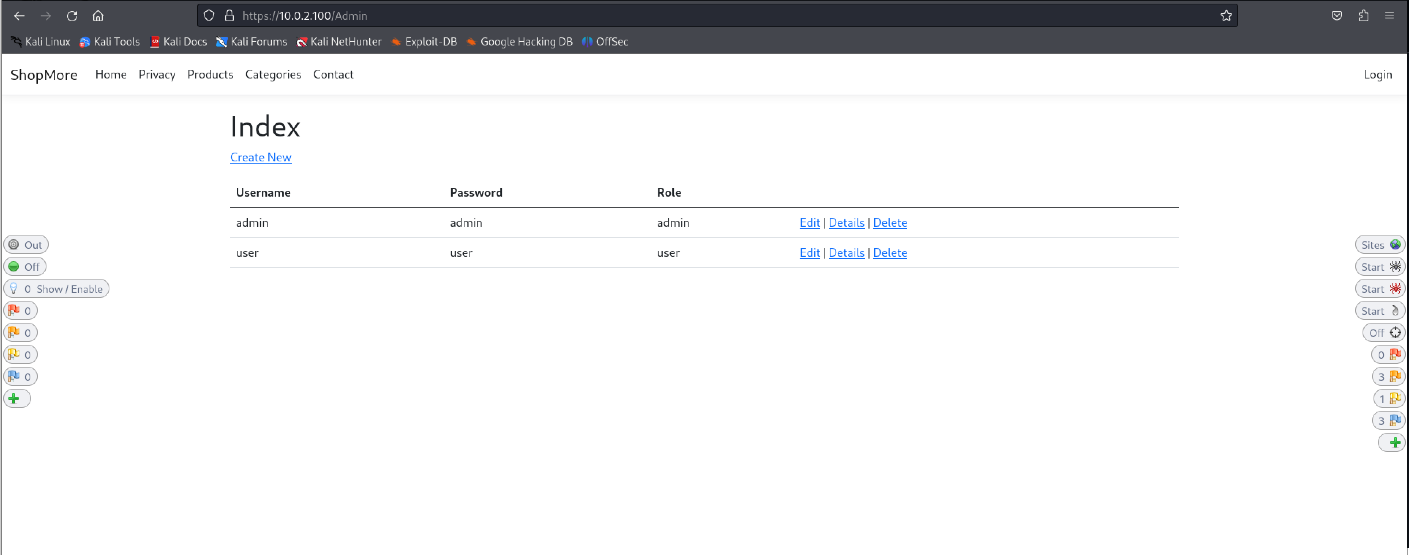
Dit type kwetsbaarheid doet zich voor wanneer een gebruiker, zonder de vereiste bevoegdheden of autorisatie, toegang krijgt tot hoger geplaatste functionaliteiten, zoals beheerders- of administratiefuncties. In dit specifieke scenario geeft de mogelijkheid om zonder inloggen toegang te krijgen tot de "/admin" pagina van de webapplicatie een duidelijk voorbeeld van verticale privilege-escalatie.

Het risico wordt geaccentueerd doordat een niet-geauthenticeerde gebruiker direct naar de admin-pagina kan navigeren, waar normaal gesproken alleen geautoriseerde beheerders of admins toegang zouden moeten hebben. Op deze pagina heeft de gebruiker volledige controle over gebruikersinformatie, inclusief namen, wachtwoorden en mogelijkheden om nieuwe gebruikers aan te maken, wat een aanzienlijke inbreuk vormt op de privacy en beveiliging van het systeem.

advies:

Zorg ervoor dat de admin pagina niet meer toegankelijk is zonder in te loggen als admin.

bewijs:



Sql injection op inlog pagina: wanneer ik “admin’—” ingeef in het username veld van de login pagina(dat is een sql statement om de user genaamd admin op te halen) dan kan ik zonder een wachtwoord in te geven inloggen als admin

risico:

Deze kwetsbaarheid, bekend als SQL-injectie, stelt kwaadwillende gebruikers in staat om kwaadaardige SQL-code in te voeren in invoervelden zoals het gebruikersnaamveld van het inlogformulier.

Door specifieke SQL-instructies, zoals "admin’—" (een SQL-statement om de gebruiker genaamd 'admin' op te halen), in te voeren in het gebruikersnaamveld, kunnen aanvallers het authenticatieproces omzeilen en zichzelf zonder geldige inloggegevens inloggen als een geprivilegieerde gebruiker, in dit geval 'admin'. Dit betekent dat een aanvaller toegang kan krijgen tot gevoelige delen van de applicatie met de rechten van een beheerder zonder de juiste referenties te verstrekken.

Het gevaar van deze kwetsbaarheid is zeer hoog. Een succesvolle SQL-injectieaanval kan leiden tot ongeautoriseerde toegang tot gevoelige gegevens, wijzigingen in de database-inhoud, diefstal van vertrouwelijke informatie zoals wachtwoorden of persoonlijke gegevens, en zelfs het compromitteren van het volledige systeem.

De impact van een dergelijke aanval kan verwoestend zijn voor de integriteit, vertrouwelijkheid en beschikbaarheid van de gegevens op de webapplicatie. Het kan leiden tot reputatieschade, verlies van vertrouwen bij gebruikers en mogelijk juridische gevolgen door schending van privacywetten.

Om deze kwetsbaarheid effectief te verhelpen, is het essentieel om invoervalidatie en parameterisatie toe te passen bij het verwerken van gebruikersinvoer. Dit omvat het gebruik van parameterized queries, het ontsnappen van speciale tekens en het implementeren van inputvalidatie om ongeautoriseerde SQL-code te voorkomen.

Het is van vitaal belang om deze kwetsbaarheid onmiddellijk aan te pakken door middel van grondige beveiligingsmaatregelen. Door de juiste beveiligingspraktijken te implementeren, zoals het vermijden van directe SQL-queryconstructies met gebruikersinvoer, kan de webapplicatie haar bescherming tegen SQL-injectieaanvallen aanzienlijk versterken en de veiligheid van het inlogproces verbeteren.

advies:

* **Gebruik Parameterized Queries:**

Implementeer parameterized queries (prepared statements) in plaats van dynamisch gegenereerde SQL-queryreeksen. Deze techniek voorkomt dat invoergegevens rechtstreeks worden samengevoegd met SQL-query's en minimaliseert het risico op SQL-injectie.

* **Input Validatie:**

Voer strikte invoervalidatie uit op alle invoervelden op de inlogpagina en controleer of de ingevoerde gegevens overeenkomen met de verwachte indeling en lengte. Wees selectief in het accepteren van gebruikersinvoer en sta alleen veilige karakters toe.

* **Escaping Special Characters:**

Ontsnap speciale karakters zoals enkele aanhalingstekens, dubbele aanhalingstekens en andere speciale SQL-tokens om te voorkomen dat ze worden geïnterpreteerd als SQL-commando's. Dit helpt om potentiële SQL-injectieaanvallen te voorkomen.

bewijs: https://youtu.be/I8r8lTMMZhg

**Geteste onderdelen**

Command injection: Dit heb ik getest door “whoami” in te geven in de contact, categories en login paginas geen van de paginas gaven de hostname van het systeel terug dit betekent dat er dus geen command injection aanwezig was

Sql injection:dit heb ik getest op de login pagina en de categories pagina bij de catagories pagina kreeg was er geen sql injection mogelijk maar bij de login pagina kon ik inloggen als admin door admin’— in te geven als gebruikersnaam zonder wachtwoord

Reverse path traversal: Dit heb ik getest door eerste ../../../etc/passwd toe te voegen aan de url. Daarna heb ik het ook geprobeert met html encoding door %2e%2e%2f in te geven. Dit werkte ook niet dus dan heb ik nog een tool genaamd dotdotpwn gebruikt om automatisch meer van deze encodings en mogelijkheden te proberen dat lukte ook niet dus heb ik geconcludeert dat reverse path traversal niet mogelijk is

Information disclosure:Ik heb ook veel van de code geanalyseert om na te kijken of er persoonlijke info te zien is. Dat was niet het geval

Xxs: Deze zwakheid heb ik getest door </script><script>alert(“test”)</script> in te geven bij de contact en categories paginas. Bij de contact pagina kreeg ik toen een alert met het woord test in dit betekent dus dat er een xxs zwakheid is op de webapp

Vertical privilege escalation:dit heb ik getest door verschillende sleutelwoorden achter de url te zetten zoals /admin of /administrator. Zo zag ik meteen dat ik zonder in te loggen naar de /admin pagina kon surfen

**Ernst gevonden zwakheden**

|  |  |
| --- | --- |
| zwakheid | ernst |
| Sql-injection | **hoog** |
| Vertical privilege escalation | **kritiek** |
| xxs | **hoog** |

**Privacy**

Het respecteren van de privacy van een klant na een penetratietest is van essentieel belang en omvat een reeks cruciale stappen en ethische overwegingen.

**Gegevensbeheer en vertrouwelijkheid:** Na het voltooien van een penetratietest zou ik ervoor zorgen dat alle gevoelige gegevens die tijdens de beoordeling zijn verkregen, met uiterste zorg worden behandeld. Dit omvat het veilig opslaan en versleutelen van alle gegevens die zijn verkregen tijdens de test. Alleen geautoriseerd personeel zou toegang moeten hebben tot deze informatie, strikt voor de analyse en rapportage.

**Geheimhoudings- en vertrouwelijkheidsovereenkomsten:** Voordat de penetratietest wordt uitgevoerd, is het essentieel om een duidelijk begrip te vestigen via geheimhoudingsovereenkomsten of vertrouwelijkheidsovereenkomsten met de klant. Hierin worden de voorwaarden vastgelegd waarbinnen gegevens worden verzameld en wordt ervoor gezorgd dat gevoelige informatie niet wordt gedeeld of vrijgegeven zonder expliciete toestemming van de klant.

**Beperkte toegang tot testresultaten:** De gedetailleerde resultaten en bevindingen van de penetratietest moeten alleen worden gedeeld met personen of teams die rechtstreeks betrokken zijn bij het analyseren van de beveiligingslekken en het opstellen van rapporten. Deze informatie mag niet worden verspreid naar ongeautoriseerde partijen.

**Anonimisering van gegevens:** In rapporten of communicatie over de penetratietest moet persoonlijk identificeerbare informatie (PII) of gevoelige gegevens die rechtstreeks aan de klant kunnen worden gekoppeld, worden geanonimiseerd of weggelaten. Dit helpt de identiteit en gegevens van de klant te beschermen tegen onbedoelde blootstelling.

**Veilige verwijdering van gegevens:** Zodra de penetratietest is voltooid en de rapporten zijn gegenereerd, moeten alle verzamelde gegevens die niet langer nodig zijn, veilig worden verwijderd of vernietigd volgens de beste praktijken in de branche. Dit omvat het wissen van tijdelijke opslagmedia en ervoor zorgen dat er geen resterende gegevens achterblijven.

**Communicatie met de klant:** Open en transparante communicatie met de klant is essentieel. Het bespreken van de omvang van gegevensverzameling, wat er met de verkregen informatie zal gebeuren en hoe deze zal worden beschermd, helpt het vertrouwen op te bouwen en toont een toewijding aan hun privacy.

**Nazorg na de beoordeling:** Bied de klant nazorg na de beoordeling door ondersteuning te bieden bij het implementeren van aanbevelingen en het oplossen van geïdentificeerde kwetsbaarheden. Hiermee benadrukt u verder uw toewijding aan hun beveiliging en privacy door hen te helpen hun systemen te versterken.

Door deze principes en praktijken te volgen, toon ik een diep respect voor uw privacy, zelfs nadat ik een penetratietest hebt uitgevoerd. Dit handhaaft niet alleen de vertrouwelijkheid, maar ook een professionele en ethische benadering van cybersecurity-beoordelingen.

**Classificatie van ernst**

1. Kritiek

• Geen uitgebreide kennis vereist; het is gemakkelijk om de kwetsbaarheid te activeren.

• Tools en scripts die de kwetsbaarheid activeren, zijn openbaar beschikbaar.

• De kwetsbaarheid kan worden geactiveerd zonder handmatige interactie van de gebruiker.

• De kwetsbaarheid kan worden geactiveerd door een anonieme actor.

• De klant heeft de materialisatie van deze kwetsbaarheid als een "Doomsday Scenario" vermeld. Tenzij anders overeengekomen, worden bevindingen van deze categorie onmiddellijk gecommuniceerd naar de contactpersonen die in het scopeformulier zijn vermeld.

2. Hoog

• Geen uitgebreide kennis vereist; het is gemakkelijk om de kwetsbaarheid te activeren.

• Tools en scripts die de kwetsbaarheid activeren, zijn openbaar beschikbaar.

• De kwetsbaarheid kan worden geactiveerd zonder handmatige interactie van de gebruiker.

3. Gemiddeld

• Alleen mogelijk wanneer aan verschillende vereisten is voldaan.

• Minder waarschijnlijk om voor te komen vanwege de behoefte aan specifieke vaardigheden.

• Er is weinig tot geen impact op de bedrijfsvoering.

• Kan niet geautomatiseerd worden.

4. Laag

• De kwetsbaarheid is moeilijk te misbruiken (bijv. het vereist dat een tegenstander aanzienlijke hoeveelheden netwerkverkeer vastlegt dat is versleuteld met dezelfde sleutel).

• Minder waarschijnlijk om voor te komen vanwege de behoefte aan gespecialiseerde vaardigheden. • Er is weinig tot geen impact op de bedrijfsvoering.

• Kan alleen worden gebruikt in combinatie met andere kwetsbaarheden.

• Kan niet worden geautomatiseerd.

5. Informatief

Kenmerken Bevindingen in deze categorie kunnen twee verschillende aard hebben.

Ze omvatten een puur theoretische kwetsbaarheid

Op het moment van testen en gedurende de testperiode konden we niet genoeg informatie verzamelen om de impact van deze bevinding te beoordelen. In dit geval kan een informatieve bevinding een effectieve CVSS-beoordeling hebben tussen 0 en 10. We raden aan alle bevindingen van deze categorie te evalueren om te bepalen of ze opnieuw moeten worden beoordeeld. Als de bevinding een puur theoretische kwetsbaarheid omvat, is er ofwel

• Geen realistische impact op vertrouwelijkheid, integriteit noch beschikbaarheid OF

• Een (bijna) onbestaande kans dat het zal materialiseren

**Conclusie**

De uitgevoerde penetratietest heeft een grondige evaluatie van de beveiligingsmaatregelen binnen de webapplicatie "ShopMore" mogelijk gemaakt. De test identificeerde verschillende kwetsbaarheden en risicogebieden die de integriteit, vertrouwelijkheid en beschikbaarheid van gegevens kunnen bedreigen.

Allereerst werd een kritieke kwetsbaarheid ontdekt op de contactpagina, waar een gebrek aan invoervalidatie resulteerde in het risico van Cross-Site Scripting (XSS). Deze zwakte stelt kwaadwillende gebruikers in staat om schadelijke scripts in te voegen en uit te voeren, waardoor de kans op gegevensdiefstal, sessie-manipulatie en andere aanvallen toeneemt. Aanbevelingen omvatten het implementeren van strikte invoervalidatie, het ontsnappen van speciale karakters en het gebruik van Content Security Policy (CSP) om XSS-aanvallen te voorkomen.

Verder bleek de inlogpagina vatbaar voor SQL-injectie, waarbij kwaadwillende gebruikers ongeautoriseerde toegang konden verkrijgen door specifieke SQL-instructies in te voeren in het gebruikersnaamveld. Aanbevolen maatregelen omvatten het gebruik van parameterized queries, invoervalidatie en het vermijden van directe SQL-queryconstructies met gebruikersinvoer om dergelijke aanvallen te voorkomen.

Bovendien werd verticale privilege-escalatie vastgesteld, waarbij niet-geauthenticeerde gebruikers zonder inloggegevens toegang konden krijgen tot de "/admin" pagina, waar normaal gesproken alleen geautoriseerde beheerders toegang zouden moeten hebben. Dit vormde een aanzienlijke bedreiging voor de vertrouwelijkheid van gegevens en de integriteit van het systeem. Aanbevelingen omvatten het implementeren van strikte autorisatiecontrole, beperking van toegangsrechten en sterke authenticatiemechanismen.

Deze bevindingen onderstrepen de noodzaak om proactieve beveiligingsmaatregelen te implementeren binnen de webapplicatie "ShopMore". Door het toepassen van strikte invoervalidatie, het implementeren van parameterized queries, het beheren van toegangsrechten en het versterken van authenticatieprocessen kan de algehele beveiliging worden verbeterd en kunnen potentiële beveiligingsrisico's worden geminimaliseerd.

Het aanpakken van deze kwetsbaarheden is van vitaal belang om de weerbaarheid van de webapplicatie tegen potentiële aanvallen te vergroten en de bescherming van gevoelige gegevens te waarborgen. Voortdurende monitoring, regelmatige beveiligingsaudits en een cultuur van veilige coderingspraktijken zullen essentieel zijn om de veiligheid van de webapplicatie op lange termijn te waarborgen.

|  |
| --- |
| xx |
|  |