2024 - 2025

|  |
| --- |
|  |
| Plan Van Aanpak  Security Audit voor: Ordolio |
|  |

Jonas Quintiens  
Student Bachelor in de Toegepaste Informatica – Applicatieontwikkeling

Inhoudsopgave

[Inleiding 2](#_Toc193449160)

[2. Doelstellingen van de pentest 3](#_Toc193449161)

[3. Scope van de Test 3](#_Toc193449162)

[In-scope componenten 3](#_Toc193449163)

[Out-of-Scope Componenten 4](#_Toc193449164)

[De volgende onderdelen worden expliciet uitgesloten van de test om de focus te behouden en onbedoelde verstoringen te voorkomen: 4](#_Toc193449165)

[Infrastructuurbeveiliging 4](#_Toc193449166)

[Netwerkpenetratietests, waaronder firewalls, servers en cloud-infrastructuur. 4](#_Toc193449167)

[Scannen van interne netwerken en privilege-escalatie binnen de bedrijfsomgeving. 4](#_Toc193449168)

[Externe diensten en afhankelijkheden 4](#_Toc193449169)

[SaaS-platformen die geïntegreerd zijn met de applicatie (zoals betalingsgateways en authenticatieproviders), tenzij anders overeengekomen. 4](#_Toc193449170)

[Cloudbeveiliging buiten misconfiguraties die de applicatie beïnvloeden. 4](#_Toc193449171)

[Fysieke beveiliging & Social Engineering 4](#_Toc193449172)

[Geen phishing-aanvallen, tailgating-tests of fysieke toegangstests. 4](#_Toc193449173)

[3.2 Uitgesloten componenten 4](#_Toc193449174)

[3.3 Beperkingen en randvoorwaarden 4](#_Toc193449175)

[4. Testmethodologie 5](#_Toc193449176)

[4.1 Informatie verzamelen en verkennen 5](#_Toc193449177)

[4.2 Geautomatiseerde kwetsbaarheidsscans 5](#_Toc193449178)

[4.3 Handmatige penetratietesten 5](#_Toc193449179)

[5. Exploitatie en Persistentie 6](#_Toc193449180)

[6. Rapportage en Beveiligingsmaatregelen 6](#_Toc193449181)

[7. Conclusie 6](#_Toc193449182)

# Inleiding

In een tijd waarin cyberdreigingen steeds geavanceerder worden, is het essentieel om de beveiliging van webapplicaties grondig te testen. Dit document beschrijft het plan van aanpak voor een penetratietest (pentest) die gericht is op het identificeren, analyseren en mitigeren van kwetsbaarheden binnen een webapplicatie.

Het doel van deze test is om een realistisch inzicht te krijgen in de beveiligingsstatus van de applicatie, mogelijke aanvalsvectoren bloot te leggen en praktische aanbevelingen te doen om de algehele cyberweerbaarheid te verhogen. De test wordt uitgevoerd volgens de methodologieën van de OWASP Top 10 en NIST SP 800-115, en omvat zowel geautomatiseerde scans als diepgaande handmatige tests.

## Doelstellingen van de pentest

De primaire doelstellingen van deze penetratietest zijn als volgt:

* Het identificeren en classificeren van kwetsbaarheden in de webapplicatie, API’s en bijbehorende infrastructuur.
* Het beoordelen van de effectiviteit van bestaande beveiligingsmaatregelen, zoals authenticatie- en autorisatiemechanismen, inputvalidatie en configuratiebeheer.
* Het simuleren van realistische aanvalsscenario’s om de impact en exploitability van kwetsbaarheden te testen.
* Het opstellen van een gedetailleerd rapport met bevindingen, risico-inschattingen en aanbevelingen voor mitigatie.

De test wordt uitgevoerd met respect voor de overeengekomen scope en beperkingen, en zal geen verstoringen veroorzaken in productieomgevingen tenzij anders overeengekomen.

## Scope van de Test

Om een gestructureerde en gecontroleerde penetratietest uit te voeren, wordt de scope vooraf gedefinieerd. Dit voorkomt ongeautoriseerde tests en zorgt ervoor dat de testresultaten representatief en relevant blijven.

### In-scope componenten

De volgende onderdelen vallen expliciet binnen de scope van de penetratietest:

##### Webapplicatie

Alle functionele onderdelen die via de webinterface toegankelijk zijn, inclusief authenticatie, gebruikersrollen en sessiebeheer.

* Bedrijfslogica en mogelijke kwetsbaarheden in gebruikersinteracties.
* API-endpoints die door de webapplicatie worden gebruikt.
* Client-side beveiliging, inclusief JavaScript-beveiliging, lokale opslag en HTTP-beveiligingsheaders.

##### Mobiele Applicatie (iOS & Android)

* Statische en dynamische analyse van de mobiele applicatie.
* Beveiliging van API-communicatie tussen de mobiele app en backend-services.
* Beveiliging van lokale opslag, inclusief versleuteling en gevoelige data-opslag.
* Pogingen tot reverse engineering en manipulatie van de app.

##### API-beveiliging

* Controle van authenticatie- en autorisatiemechanismen bij API-endpoints.
* Testen op inputvalidatie en bedrijfslogica kwetsbaarheden.
* Controle op onbedoelde blootstelling van data, rate limiting en CORS-configuratie.

### Out-of-Scope Componenten

### De volgende onderdelen worden expliciet uitgesloten van de test om de focus te behouden en onbedoelde verstoringen te voorkomen:

##### Infrastructuurbeveiliging

* Netwerkpenetratietests, waaronder firewalls, servers en cloud-infrastructuur.
* Scannen van interne netwerken en privilege-escalatie binnen de bedrijfsomgeving.

##### Externe diensten en afhankelijkheden

* SaaS-platformen die geïntegreerd zijn met de applicatie (zoals betalingsgateways en authenticatieproviders), tenzij anders overeengekomen.
* Cloudbeveiliging buiten misconfiguraties die de applicatie beïnvloeden.

##### Fysieke beveiliging & Social Engineering

* Geen phishing-aanvallen, tailgating-tests of fysieke toegangstests.

### Uitgesloten componenten

Om operationele verstoringen te minimaliseren, worden bepaalde onderdelen expliciet uitgesloten van de test, tenzij anders overeengekomen. Dit kunnen bijvoorbeeld interne netwerken, derde-partij services of productiegegevens zijn.

### Beperkingen en randvoorwaarden

* De test zal worden uitgevoerd binnen een vooraf bepaalde tijdsperiode om impact op de bedrijfsvoering te minimaliseren.
* Er wordt een vooraf goedgekeurde test- of lokale omgeving gebruikt om verstoringen in productiesystemen te voorkomen, tenzij anders overeengekomen.

## Testmethodologie

De penetratietest wordt uitgevoerd volgens een gelaagde aanpak, waarbij zowel automatische als handmatige technieken worden ingezet.

#### Informatie verzamelen en verkennen

In de eerste fase wordt het aanvalsvlak in kaart gebracht. Dit omvat het verzamelen van openbare en semi-openbare informatie die een aanvaller kan gebruiken om potentiële aanvalsvectoren te identificeren.

* Subdomein- en DNS-enumeratie via tools zoals Amass en Sublist3r.
* Identificeren van publieke kwetsbaarheden via OSINT-technieken en Google Dorking.
* Analyseren van configuratiefouten en potentieel gelekte gegevens in code repositories zoals GitHub.

#### Geautomatiseerde kwetsbaarheidsscans

Na de verkenningsfase worden verschillende geautomatiseerde tools ingezet om snel bekende kwetsbaarheden te detecteren.

* **Statische code-analyse (SAST):** Snyk en dependaBot worden gebruikt om kwetsbaarheden in de broncode en dependencies op te sporen.
* **Dynamische applicatiescans (DAST):** OWASP ZAP, Burp Suite en Nikto worden ingezet om de webapplicatie te scannen op bekende OWASP Top 10 kwetsbaarheden.
* **SSL CONFIG en Domain security posture:** Ik zal met verschillende ssl en header scans beoordelen of de domein configuratie veilig is.

#### Handmatige penetratietesten

Geautomatiseerde tools kunnen niet alle logische kwetsbaarheden of geavanceerde aanvalstechnieken detecteren. Daarom wordt in deze fase een handmatige beoordeling uitgevoerd.

* **Authenticatie en autorisatie testen:** Onderzoeken of accounts kunnen worden overgenomen via zwakke wachtwoorden, sessie-hijacking of Multi-Factor Authenticatie (MFA) bypass technieken.
* **Injectieaanvallen:** Testen op SQL Injection (SQLi), Cross-Site Scripting (XSS), Server-Side Request Forgery (SSRF) en andere invoervalidatieproblemen.
* **API beveiligingstesten:** Verificatie van API authenticatie, autorisatie en mogelijke data-exfiltratie via onjuist geconfigureerde endpoints.
* **Business logic flaws:** Testen op onbedoelde functionaliteit, zoals onjuiste validatie van transactiebedragen of ongeautoriseerde toegang tot resources.

## Exploitatie en Persistentie

Indien kwetsbaarheden worden geïdentificeerd, worden gecontroleerde exploit-scenario’s uitgevoerd om de impact te beoordelen. Dit gebeurt zonder destructieve acties en met volledige logging en rapportage.

* **Privilege escalation testen:** Pogingen om van een normale gebruiker over te stappen naar een beheerdersrol.
* **Persistente toegang verkrijgen:** Controleren of kwetsbaarheden kunnen worden gebruikt om langdurige toegang te behouden, zoals via verborgen backdoors.

## Rapportage en Beveiligingsmaatregelen

Na voltooiing van de test wordt een gedetailleerd rapport opgesteld waarin de bevindingen, risico-inschattingen en aanbevolen mitigatiestrategieën worden gepresenteerd.

* **Risico-classificatie van kwetsbaarheden:** Op basis van CVSS-scores en potentiële bedrijfsimpact.
* **Technische en strategische aanbevelingen:** Concrete actiepunten om kwetsbaarheden te verhelpen en toekomstige risico’s te minimaliseren.
* **Security hardening richtlijnen:** Best practices voor het verbeteren van applicatie- en infrastructuurbeveiliging.

Tevens wordt een optionele her-test aangeboden om te valideren of de geïdentificeerde kwetsbaarheden adequaat zijn opgelost.

## Conclusie

Deze penetratietest biedt een diepgaand inzicht in de beveiligingsstatus van de webapplicatie en helpt de organisatie om risico’s proactief te mitigeren. Door een combinatie van geautomatiseerde en handmatige technieken worden zowel technische kwetsbaarheden als logische zwakheden blootgelegd, wat een essentiële stap is in het versterken van de algehele cyberweerbaarheid.

Door deze aanpak krijgt de organisatie niet alleen inzicht in potentiële dreigingen, maar ook concrete aanbevelingen voor verbeteringen die bijdragen aan een veiligere digitale omgeving.