Onderzoeksvoorstel

Ontwerp van een geautomatiseerd systeem voor security monitoring en incident response in SaaS-omgevingen binnen een DevOps-infrastructuur.

Bachelorproef, 2025-2026

Sem Demoen

E-mail: sem.demoen@student.hogent.be

Samenvatting

Deze bachelorproef onderzoekt hoe een geautomatiseerd systeem voor security monitoring en incident response kan worden ontworpen om securitydata binnen SaaS-omgevingen effectief te centraliseren, analyseren en rapporteren in een DevOps-infrastructuur. Het onderzoek vertrekt vanuit een concrete casus bij Devinity, waarbij huidige uitdagingen zoals verspreide tools, vertraagde incidentdetectie en inconsistente rapportering in kaart worden gebracht. De centrale onderzoeksvraag luidt: "Hoe kan een geautomatiseerd systeem voor security monitoring en incident response worden ontworpen om securitydata binnen SaaS-omgevingen effectief te centraliseren, analyseren en rapporteren in een DevOps-infrastructuur?" Het doel is een proof-of-concept te ontwikkelen dat logs, events en alerts van SaaS-platforms (zoals Microsoft 365, Google Workspace, Okta) en DevOps-omgevingen verzamelt via API's, webhooks en logforwarders, deze gegevens normaliseert en analyseert met rule-based en machine learning-methoden, en automatische rapportages genereert voor technische teams en management. De methodologie omvat een literatuurstudie, analyse van bestaande security monitoring tools, ontwerp en implementatie van een prototype en evaluatie aan de hand van prestatie-indicatoren zoals detectiesnelheid, foutpositieven en responstijd. Het verwachte resultaat is een functioneel proof-of-concept dat operationele efficiëntie verhoogt, sneller incidenten detecteert en consistente, bruikbare rapportages oplevert. De meerwaarde voor de doelgroep bestaat uit een concreet en herhaalbaar framework voor geautomatiseerde security monitoring binnen SaaS en DevOps, dat organisaties helpt risico's beter te beheersen en hun incident response te optimaliseren.

Keuzerichting: System & Network Administrator

Inhoudsopgave

1 Inleiding 1
2 Literatuurstudie 2
2.1 Security monitoring & SIEM 2
2.2 SaaS & cloud logging 2
2.3 Correlatie, threat intelligence & detectie 2
2.4 Metrics & rapportering 2
3 Methodologie 2
4 Verwacht resultaat, conclusie 3
Referenties 3

1. Inleiding

Deze bachelorproef onderzoekt hoe een geautomatiseerd systeem voor security monitoring en incident response kan worden ontworpen om securitydata binnen SaaS-omgevingen effectief te centraliseren, analyseren en rapporteren in een DevOps-infrastructuur.

Het onderzoek vertrekt vanuit een concrete casus bij Devinity.eu, een SaaS-bedrijf dat momenteel meerdere tools gebruikt voor security monitoring en incidentdetectie. Hierdoor ontstaan problemen zoals vertraagde detectie, onvolledige rapporten en inconsistentie in incidentafhandeling. De doelgroep bestaat uit DevOps- en securityteams binnen SaaS-bedrijven, die baat hebben bij een gecentraliseerd en geautomatiseerd systeem dat de operationele efficiëntie verhoogt en risico's beter beheersbaar maakt.

De centrale onderzoeksvraag luidt: **Hoe kan een geautomatiseerd systeem voor security monitoring en incident response worden ontworpen om securitydata binnen SaaS-omgevingen effectief te centraliseren, analyseren en rapporteren in een DevOps-infrastructuur?**

De doelstelling van deze bachelorproef is het ontwikkelen van een proof-of-concept van een gecentraliseerd systeem dat:

- Securitydata uit SaaS-applicaties (zoals Microsoft 365, Google Workspace, Okta) en DevOps-omgevingen verzamelt via API's, webhooks en logforwarders;
- Data normaliseert en analyseert om incidenten sneller en accurater te detecteren;
- Automatische rapportages genereert voor technische teams en management;

- KPI's levert om de effectiviteit van monitoring en incident response te meten.

Het concrete eindresultaat is een werkend prototype dat de voordelen van centralisatie, snelle detectie en consistente rapportage aantoont.

2. Literatuurstudie

Deze literatuurstudie behandelt de state-of-the-art van security monitoring, incident response en data-analyse binnen SaaS-omgevingen en DevOps. Het doel is een theoretische basis te leggen voor het ontwerp van het proof-of-concept en de selectie van technologieën.

2.1. Security monitoring & SIEM

Traditionele intrusion detection- en prevention-systemen (IDPS) vormen de basis van security monitoring. Volgens Scarfone & Mell (2007) zijn detectie, correlatie van events en het beperken van false positives kernfuncties van IDPS-systemen (Scarfone & Mell, 2007). Moderne SIEM-oplossingen zoals Elastic Security en OpenSearch Security combineren log ingestion, correlatie, dashboards en alerting in één platform. Deze systemen zijn goed toepasbaar binnen DevOps-omgevingen en vormen de basis voor geautomatiseerde monitoring.

2.2. SaaS & cloud logging

SaaS-omgevingen bieden logging- en securitymonitoring via API's, webhooks en agents. De Cloud Security Alliance benadrukt het belang van gecentraliseerde logging en governance (Alliance, 2019). Microsoft 365 en Okta documenteren welke logs en events beschikbaar zijn en hoe deze kunnen worden verzameld. Dit toont aan dat centralisatie van securitydata uit SaaS-diensten praktisch haalbaar is.

2.3. Correlatie, threat intelligence & detectie

Het normaliseren van data en toevoegen van context is cruciaal voor effectieve detectie. Mavroeidis & Bromander (2017) beschrijven taxonomieën en modellen voor cyber threat intelligence, wat helpt bij het structureren van alerts (Mavroeidis & Bromander, 2017). Het MITRE ATT&CK Framework ondersteunt het definiëren van detectieregels en analyse van incidenten, waardoor een systematische aanpak mogelijk is.

2.4. Metrics & rapportering

Het meten van prestaties en effectiviteit is essentieel. NIST SP 800-55 (NIST SP 800-55, 2007) en Jaquith (2007) beschrijven KPI's zoals Mean Time To Detect (MTTD), Mean Time To Respond (MTTR) en aantal false positives. Deze metrics zijn bruikbaar voor zowel technische teams als management om de effectiviteit van het systeem te evalueren.

3. Methodologie

Het onderzoek wordt uitgevoerd in vier fasen:

1. **Probleemanalyse:** Samenwerking met Devinity.eu om huidige monitoringtools, workflows en tekortkomingen in kaart te brengen. Analyse van bestaande logs, incidenten en KPI's.
2. **Architectuurontwerp:** Selectie van technologieën zoals Elastic Stack, Wazuh en OpenSearch voor het verzamelen, normaliseren en analyseren van data. Definiëren van datastromen via API's, webhooks en logforwarders.
3. **Proof-of-concept ontwikkeling:** Implementatie van centrale logging, correlatie- en analysemethoden (rule-based en eenvoudige machine learning), en automatische rapportage. Testen binnen een DevOps-pipeline.
4. **Evaluatie en KPI-meting:** Meten van detectiesnelheid, aantal false positives, responstijd en percentage automatisch afgehandelde incidenten. Analyse van resultaten en aanbevelingen voor optimalisatie.

Tijdsplanning:

- Fase 1: 3 weken – requirements-analyse en dataverzameling.
- Fase 2: 2 weken – architectuurontwerp en selectie van tools.
- Fase 3: 6 weken – implementatie van proof-of-concept.
- Fase 4: 2 weken – evaluatie, metingen en rapportering.

De deliverables per fase zijn:

- Fase 1: Analyseverslag van huidige situatie en probleemstelling.
- Fase 2: Ontwerpspecificatie van de architectuur en datastromen.
- Fase 3: Werkend proof-of-concept systeem.
- Fase 4: Evaluatierapport met KPI's, conclusies en aanbevelingen.

4. Verwacht resultaat, conclusie

Het verwachte resultaat is een proof-of-concept van een gecentraliseerd securitymonitoringsysteem dat:

- Securitydata uit meerdere SaaS- en DevOps-bronnen centraliseert en normaliseert;
- Incidenten sneller en betrouwbaarder detecteert via correlatie en analyse;
- Automatische rapportages genereert voor technische teams en management;
- KPI's levert zoals MTTD, MTTR, aantal false positives en responstijd.

De meerwaarde voor de doelgroep (DevOps- en securityteams bij SaaS-bedrijven) ligt in hogere operationele efficiëntie, betere risicoinschatting en snellere incidentafhandeling. Het onderzoek levert zowel praktische inzichten als een concrete demonstratie van een geautomatiseerd monitoring- en incident response-systeem.

Referenties

Alliance, C. S. (2019). Cloud Controls Matrix (CCM). https://cloudsecurityalliance.org/artifacts/cloud-controls-matrix-v4-0-1/

Mavroeidis, V., & Bromander, S. (2017). Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. *Computers & Security*, *72*, 144–157. https://doi.org/10.1016/j.cose.2017.08.001

NIST SP 800-55. (2007). *Performance Measurement Guide for Information Security*.

Scarfone, K., & Mell, P. (2007). *Guide to Intrusion Detection and Prevention Systems (IDPS), NIST SP 800-94*. https://doi.org/10.6028/NIST.SP.800-94