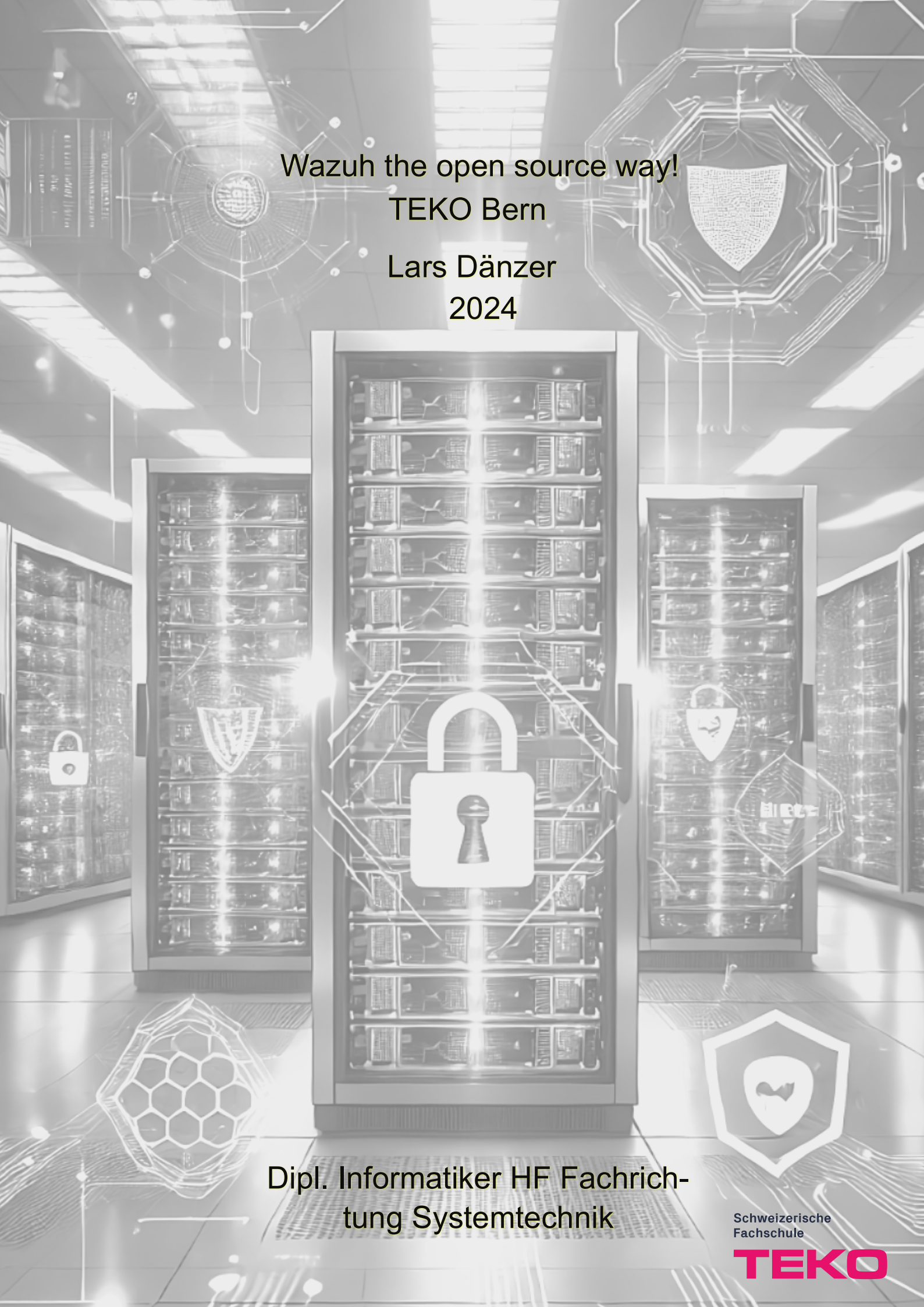

Wazuh the open source way!

TEKO Bern

Lars Dänzer

2024

Dipl. Informatiker HF Fachrichtung Systemtechnik

Schweizerische Fachschule TEKO

1 Inhaltsverzeichnis

2 Management Summary

Diese Diplomarbeit befasst sich mit der Implementierung und Evaluation von Wazuh, einer Open-Source-Lösung zur Bedrohungserkennung und Sicherheitsüberwachung, im Rahmen eines Proof of Concept. Ziel ist es, die Machbarkeit und Effektivität von Wazuh zu untersuchen und dabei den praktischen Nutzen für Unternehmen zu evaluieren, die eine kosteneffiziente und skalierbare Lösung für ihre Sicherheitsanforderungen suchen.

2.1 Themenbeschreibung und Ausgangslage

Die zunehmende Bedrohungslage im Bereich der IT-Sicherheit stellt Unternehmen vor grosse Herausforderungen. Die Notwendigkeit, sowohl interne als auch externe Bedrohungen frühzeitig zu erkennen, ist in der heutigen digitalen Ära von höchster Priorität. Viele Unternehmen stehen vor der Wahl, entweder teure proprietäre Systeme zu nutzen oder auf flexible Open-Source-Alternativen wie Wazuh zurückzugreifen. Die vorliegende Arbeit widmet sich der Frage, ob Wazuh als Open-Source-Alternative eine angemessene Lösung für die Bedrohungserkennung und Überwachung darstellen kann, und wie gut sich diese Lösung in einer Unternehmensumgebung bewährt.

2.2 Proof of Concept

Der Schwerpunkt der Arbeit liegt auf der praktischen Implementierung eines Wazuh-Systems in einer Cloud-Umgebung. Der PoC umfasst die Installation und Konfiguration des Wazuh-Servers sowie die Anbindung von Agenten auf verschiedenen Endgeräten, darunter Windows, Linux und macOS. Dies ermöglicht eine praxisnahe Simulation einer Unternehmensinfrastruktur. Im Anschluss werden benutzerdefinierte Regeln und Alerts entwickelt, um spezifische Bedrohungsszenarien zu adressieren. Ein weiteres Ziel ist die Erstellung benutzerfreundlicher Dashboards zur Überwachung der Sicherheitslage in Echtzeit.

2.3 Erfolgskriterien

Der Erfolg dieses Projekts wird anhand mehrerer Schlüsselkriterien gemessen. Dazu gehörten die problemlose Installation und Konfiguration der Wazuh-Plattform sowie die reibungslose Anbindung der Agenten an den zentralen Wazuh-Server. Ein weiteres Kriterium ist die Funktionalität der entwickelten Regeln und Alerts: Diese müssen nicht nur präzise Bedrohungen erkennen, sondern auch eine minimale Rate an Fehlalarmen aufweisen. Die Leistungsfähigkeit des Systems wird durch die Evaluation der Skalierbarkeit und Benutzerfreundlichkeit beurteilt, insbesondere in Bezug auf die Bedienbarkeit für IT-Administratoren.

2.4 Wirtschaftlichkeit und Nutzen

Wazuh bietet als Open-Source-Lösung den Vorteil, keine Lizenzkosten zu verursachen, was besonders für kleine und mittelständische Unternehmen attraktiv ist. Im Gegensatz zu proprietären SIEM-Systemen ist die Anpassungsfähigkeit von Wazuh ein weiterer Pluspunkt: Benutzerdefinierte Regeln und Dashboards können individuell auf die spezifischen Sicherheitsanforderungen eines Unternehmens zugeschnitten werden. Zusätzlich ermöglicht Wazuh durch seine Echtzeit-Überwachung und Bedrohungserkennung eine frühzeitige Reaktion auf Sicherheitsvorfälle, was potenziell hohe Kosten für Datenverluste oder Systemausfälle verhindert.

2.5 Schlussfolgerungen

Die vorläufigen Ergebnisse des Proof of Concept zeigen, dass Wazuh eine flexible, kosteneffiziente und leistungsstarke Plattform zur Sicherheitsüberwachung darstellt. Die zentrale Überwachung von Logs, die Anpassbarkeit von Alerts und die Benutzerfreundlichkeit der Dashboards machen Wazuh zu einer praktikablen Lösung für Unternehmen, die eine umfassende Sicherheitsstrategie verfolgen. Die Arbeit zeigt auch, dass Wazuh als Open-Source-Lösung mit proprietären Alternativen mithalten kann, insbesondere was die Flexibilität und die Anpassung an individuelle Bedürfnisse angeht.

Im weiteren Verlauf der Arbeit wird das Potenzial von Wazuh für eine produktive Implementierung in grösseren Unternehmensstrukturen bewertet. Durch die gewonnenen Erkenntnisse wird die Grundlage geschaffen, Wazuh als Grundlage für zukünftige Sicherheitslösungen weiterzuentwickeln und den Nutzen von Open-Source-Technologien in der IT-Sicherheitsbranche weiter zu demonstrieren.

3 Lebenslauf

Abbildung 1: Portrait Lars Dänzer

Lars Dänzer
Security Engineer, Inseya AG

Stegfeldweg 7b, 3550 Langnau i.E.

lars@daenzer.io

Profil

Nachdem ich den Bereich der Entwicklung und des Prototypenbaus verlassen habe, arbeite ich nun in der Informatik als Administrator, Application Manager und technischer Projektleiter. Ich halte mein Fachwissen stets auf dem neuesten Stand und passe IT-Lösungen gezielt an die Bedürfnisse meiner Kunden an. Dank meines analytischen, strategischen und zukunftsorientierten Denkens erkenne ich schnell Potenziale und bin stets offen für innovative Lösungen. Als Bindeglied zwischen Technologie und Menschen nutze ich meine strategischen Fähigkeiten, um Herausforderungen effizient zu meistern und neue Wege zu finden.

Berufserfahrung

Security Engineer, Inseya AG, Bern
Sept. 2024 bis Heute

Tech Lead Access Management, Securix AG, Olten
Jan. 2024 bis August 2024

Senior Security Consultant, United Security Providers AG, Bern
Dez. 2022 bis Dez. 2023

Senior Consultant, Information Consulting Group AG, Bern
- Führungsunterstützungsbasis des Bundes (Projekt ICAM)

Feb. 2021 bis Nov. 2022

Application Manager, Adnovum AG, Bern
- Führungsunterstützungsbasis des Bundes (Projekt ICAM)

Aug. 2019 bis Jan. 2021

System Engineer, HaslerRail AG, Bern
Okt. 2018 bis Juli 2019

Electronics Engineer, ICU Tech GmbH, Signau
März 2017 bis Sept. 2018

Ausbildung

CAS - Digital Forensics & Cyber Investigation Spezialist 2, Berner Fachhochschule, Bern 2022 bis 2023

CAS - Digital Forensics & Cyber Investigation Spezialist 1, Berner Fachhochschule, Bern 2022 bis 2021

CAS - Digital Forensics & Cyber Investigation Advanced, Berner Fachhochschule, Bern 2021 bis 2022

CAS - Digital Forensics & Cyber Investigation Fundamentals, Berner Fachhochschule, Bern 2021 bis 2021

Elektroniker EFZ, Kern AG, Konolfingen
2012 – 2016

Zertifikate

CCSK, Cloud Security Alliance
Sept. 2024

SASE Expert Level 2, Cato Networks
Sept. 2024

SASE Expert Level 1, Cato Networks
Sept. 2024

SSE Expert, Cato Networks
Sept. 2024

4 Aufgabenstellung

Die vorliegende Diplomarbeit hat zum Ziel, einen Proof of Concept für die Implementierung von Wazuh, einer Open-Source-Lösung zur Bedrohungserkennung und Sicherheitsüberwachung, durchzuführen. Der Fokus liegt dabei auf der Evaluierung der Praktikabilität und des Potenzials von Wazuh als zentrale Sicherheitsplattform in einer Unternehmensumgebung. Die Aufgabenstellung umfasst mehrere zentrale Bereiche: Installation, Konfiguration, Entwicklung von benutzerdefinierten Sicherheitsregeln, Erstellung von Dashboards zur Echtzeit-Überwachung und Analyse der funktionalen und nicht-funktionalen Anforderungen.

Installation und Konfiguration des Wazuh-Systems:
Die erste Aufgabe besteht in der Installation eines zentralen Wazuh-Servers in einer Cloud-Umgebung. Der Server dient als Hauptkomponente für die Verwaltung und Analyse aller sicherheitsrelevanten Daten. Parallel dazu müssen Wazuh-Agenten auf verschiedenen Endgeräten, wie Windows-, Linux- und macOS-Systemen, installiert und konfiguriert werden. Diese Agenten sammeln sicherheitsrelevante Daten und übertragen diese an den Server zur Analyse. Ziel ist es, ein funktionales System aufzusetzen, das die zentralisierte Überwachung von Betriebssystemversionen, Schwachstellenerkennung und die Erfassung von Log-Daten ermöglicht.

Entwicklung benutzerdefinierter Regeln und Alerts:
Die nächste Aufgabe besteht in der Entwicklung spezifischer, benutzerdefinierter Regeln zur Bedrohungserkennung. Diese Regeln sollen auf typische Bedrohungsszenarien abgestimmt sein, wie etwa die Erkennung von verdächtigen Aktivitäten oder Schwachstellen in Betriebssystemen und Anwendungen. Dabei liegt ein besonderes Augenmerk auf der Balance zwischen Sensitivität und Präzision der Regeln, um sowohl echte Bedrohungen frühzeitig zu erkennen als auch Fehlalarme zu minimieren. Im Rahmen des PoC sollen mindestens drei Regeln implementiert werden, die Sicherheitsbedrohungen spezifisch für das simulierte Unternehmensumfeld adressieren.

Zusätzlich zu den Regeln werden benutzerdefinierte Alerts konfiguriert. Diese Warnmeldungen sollen bei erkannten Bedrohungen automatisch generiert werden und den Administrator über potenzielle Gefahren informieren. Diese Alerts müssen über verschiedene Kanäle (z.B. E-Mail oder Dashboard-Benachrichtigungen) versendet werden können, um eine schnelle und effiziente Reaktion auf Vorfälle zu gewährleisten.

Erstellung von Dashboards zur Sicherheitsüberwachung:
Ein weiterer zentraler Aspekt der Aufgabenstellung ist die Entwicklung benutzerfreundlicher Dashboards zur Überwachung der Sicherheitslage. Diese Dashboards sollen eine Echtzeit-Übersicht über sicherheitsrelevante Aktivitäten und Bedrohungen bieten. Dabei müssen verschiedene Visualisierungsoptionen, wie Diagramme und Metriken, integriert werden, um eine schnelle Analyse und Entscheidungsfindung zu ermöglichen. Das Dashboard soll interaktive Elemente enthalten, die es dem Administrator erlauben, von allgemeinen Sicherheitsinformationen zu detaillierten Berichten über spezifische Bedrohungen zu navigieren.

Evaluation der funktionalen und nicht-funktionalen Anforderungen:
Im Rahmen der Evaluierung des Systems wird die Effektivität der implementierten Regeln und Alerts geprüft. Hierbei liegt der Fokus auf der Genauigkeit der

Bedrohungserkennung sowie der Benutzerfreundlichkeit des Systems. Die Skalierbarkeit des Wazuh-Systems wird ebenfalls untersucht, insbesondere in Bezug auf die Überwachung einer wachsenden Anzahl von Endgeräten und die Verarbeitung steigender Datenmengen.

Neben den funktionalen Anforderungen wird auch die Benutzerfreundlichkeit des Systems bewertet. Hierzu gehören Aspekte wie die einfache Bedienbarkeit der Benutzeroberfläche und die Effizienz der Dashboards. Ziel ist es, ein System zu schaffen, das sowohl leistungsfähig als auch einfach zu bedienen ist, sodass es auch von Administratoren ohne tiefgehende Programmierkenntnisse effektiv genutzt werden kann.

Zusammenfassung der Aufgabenstellung:
Die zentrale Aufgabe dieser Diplomarbeit ist die vollständige Implementierung und Evaluation eines Wazuh-Systems im Rahmen eines Proof of Concept. Dabei sollen sowohl technische als auch benutzerorientierte Aspekte berücksichtigt werden, um eine fundierte Grundlage für die potenzielle Weiterentwicklung von Wazuh als marktfähiges Sicherheitsprodukt zu schaffen. Die Ergebnisse des PoC sollen die Effektivität und Skalierbarkeit von Wazuh aufzeigen und dabei helfen, die Einsatzmöglichkeiten in realen Unternehmensumgebungen zu bewerten.

5 Terminplan

Der Terminplan ist auf eine Laufzeit von 6 Wochen angelegt, basierend auf den Anforderungen der Diplomarbeit und den Meilensteinen des Projekts. Die Arbeit folgt dem angepassten Wasserfallmodell, das eine klare, sequenzielle Bearbeitung sicherstellt. Dabei werden die verschiedenen Phasen des Projekts in sinnvolle Arbeitspakete unterteilt, um die Fortschritte transparent zu verfolgen.

Woche 1: Installation und Grundkonfiguration (16. – 22. September 2024)

Ziele:
Aufsetzen des Wazuh-Servers in einer Cloud-Umgebung (z.B. AWS, Azure, Hetzner).
Installation und Konfiguration von Wazuh-Agenten auf verschiedenen Betriebssystemen (Windows, Linux, macOS).
Verbindung der Agenten mit dem zentralen Wazuh-Server zur Erfassung sicherheitsrelevanter Daten.
Sicherstellung der reibungslosen Kommunikation zwischen Agenten und Server.

Meilenstein:
Erfolgreiche Grundinstallation und Konfiguration des Wazuh-Systems.
Anbindung der Endgeräte und Erfassung erster Log-Daten.

Woche 2 – 3: Entwicklung von Regeln und Alerts (23. September – 06. Oktober 2024)

Ziele:
Analyse typischer Bedrohungsszenarien (z.B. Schwachstellenerkennung, verdächtige Netzwerkaktivitäten).
Entwicklung und Implementierung von mindestens drei benutzerdefinierten Regeln zur Bedrohungserkennung.
Konfiguration und Test von Alerts für erkannte Sicherheitsereignisse.
Feinabstimmung der Regeln und Alerts, um Fehlalarme zu minimieren und die Präzision zu erhöhen.

Meilenstein:
Fertigstellung und erfolgreiche Tests aller benutzerdefinierten Regeln und Alerts.

Woche 4 – 5: Erstellung der Dashboards (07. – 20. Oktober 2024)

Ziele:
Design und Entwicklung benutzerfreundlicher Dashboards.
Integration relevanter Sicherheitsmetriken und -trends (z.B. Anzahl der erkannten Bedrohungen, Zustand der überwachten Systeme).

Meilenstein:
Abschluss der Dashboard-Entwicklung und positive Testbewertungen durch IT-Sicherheitsexperten.

Woche 6: Dokumentation und Abschluss (21. – 27. Oktober 2024)

Ziele:
Zusammenstellung und Analyse der Projektergebnisse.
Erstellung der vollständigen technischen Dokumentation, die die Implementierung, Regeln, Alerts und Dashboards beschreibt.
Schreiben des Abschlussberichts, einschliesslich einer Evaluation der Effektivität und Benutzerfreundlichkeit des Systems.

Meilenstein:
Abschluss der Dokumentationpos

Meilensteine und Abgabetermine:

22. September 2024: Funktionsfähige Grundinstallation des Wazuh-Systems.
6. Oktober 2024: Fertigstellung und Test aller Regeln und Alerts.
20. Oktober 2024: Abschluss der Dashboard-Entwicklung.
28. Oktober 2024: Abgabe der Dokumentation.

Der Terminplan wurde wie beschrieben umgesetzt und es gab keine Abweichungen.

6 Wazuh the open source way!

In diesem Kapitel wird die Open-Source-Lösung Wazuh vorgestellt, beginnend mit einem Überblick über die technischen Hintergründe und die Architektur des Systems. Zudem wird der durchgeführte Proof of Concept im Detail beschrieben, bei dem Wazuh in einer Cloud-Umgebung implementiert und mit verschiedenen Endgeräten verbunden wurde. Die Hauptziele dieses PoC waren, die praktische Einsatzfähigkeit von Wazuh zu bewerten und herauszufinden, ob die Plattform als Grundlage für ein zukünftiges kommerzielles Sicherheitsprodukt dienen könnte.

Neben der Implementierung stehen auch die Anforderungen an das System im Fokus, einschliesslich der funktionalen und nicht-funktionalen Aspekte, die bei der Evaluation berücksichtigt wurden. Dazu gehört die Frage, inwieweit Wazuh in der Lage ist, Bedrohungen zuverlässig zu erkennen, wie gut die Plattform skaliert und wie benutzerfreundlich sie für IT-Administratoren ist.

Dieses Kapitel führt den Leser durch die wesentlichen Schritte der Implementierung von Wazuh, von der Installation und Konfiguration bis hin zur Entwicklung benutzerdefinierter Regeln und Alerts sowie der Erstellung von Dashboards, die eine zentrale Übersicht über die sicherheitsrelevanten Aktivitäten in der Unternehmensumgebung bieten. Abschliessend wird eine Evaluation der Plattform vorgenommen, um die Stärken und Schwächen von Wazuh im Hinblick auf die spezifischen Anforderungen des Proof of Concept zu analysieren.

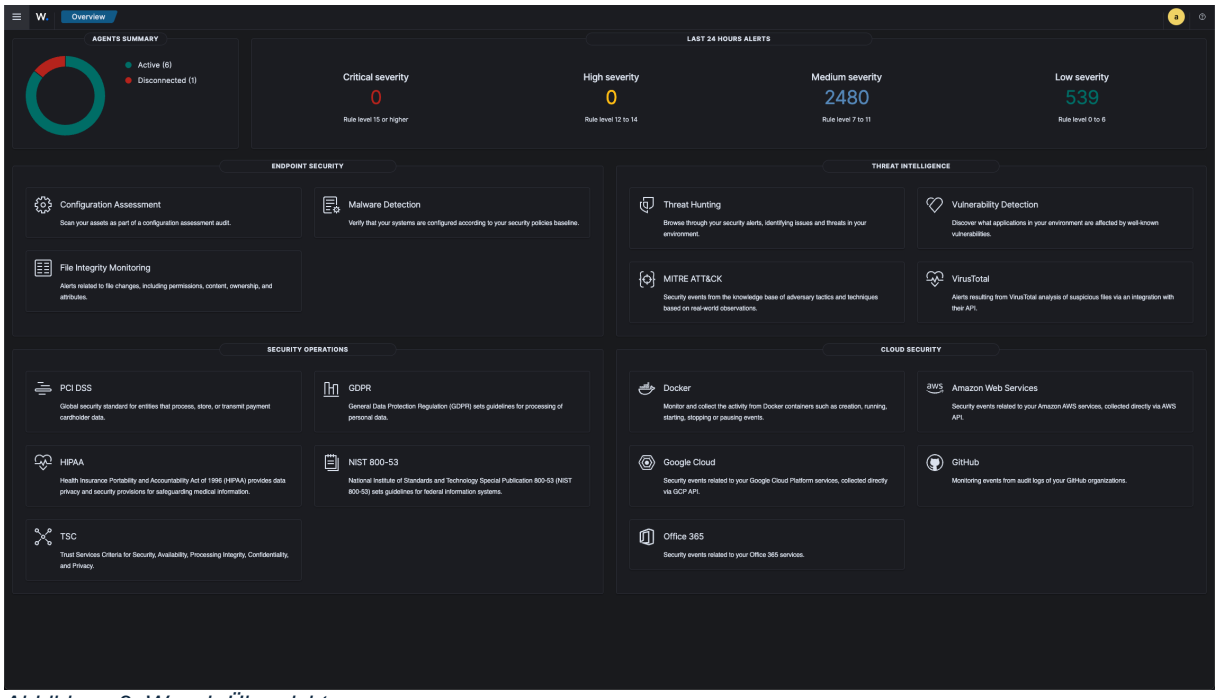

Abbildung 2: Wazuh Übersicht

6.1 Technischer Hintergrund

Die technologische Grundlage von Wazuh basiert auf einer zentralisierten Architektur, die es ermöglicht, Sicherheitsereignisse effizient zu sammeln, zu analysieren und darauf zu reagieren. Als Open-Source-Plattform zur Bedrohungserkennung und Sicherheitsüberwachung kombiniert Wazuh verschiedene Funktionen, die in kommerziellen Sicherheitslösungen häufig nur kostenpflichtig erhältlich sind. Wazuh integriert SIEM- und XDR-Funktionalitäten, die es erlauben, Ereignisse in Echtzeit zu korrelieren und so sowohl interne als auch externe Bedrohungen frühzeitig zu erkennen.

SIEM ist eine Schlüsselkomponente von Wazuh, die es ermöglicht, sicherheitsrelevante Daten von verschiedenen Quellen zu sammeln und zu korrelieren. Diese Daten umfassen Systemprotokolle, Netzwerksicherheitsereignisse und Anwendungsevents, die alle in einer zentralen Datenbank zusammengeführt werden. Die Echtzeit-Analyse dieser Informationen ermöglicht es, verdächtige Aktivitäten schnell zu identifizieren und auf Sicherheitsvorfälle umgehend zu reagieren.

Zusätzlich bietet Wazuh XDR, dass die Erkennungs- und Reaktionsmöglichkeiten über verschiedene Sicherheitsebenen hinweg erweitert. Während klassische SIEM-Systeme oft auf die Analyse von Logs beschränkt sind, ermöglicht XDR die Integration und Überwachung von Sicherheitsdaten aus verschiedensten Quellen, wie Endpunkten, Netzwerken und Cloud-Umgebungen. Dies bietet eine umfassendere Sicht auf mögliche Bedrohungen und steigert die Reaktionsfähigkeit.

Die Architektur von Wazuh setzt auf eine verteilte Infrastruktur, bestehend aus einem zentralen Managementserver und Agenten, die auf überwachten Endgeräten installiert sind. Der Managementserver übernimmt die zentrale Koordination, Datenspeicherung und Analyse, während die Agenten sicherheitsrelevante Daten sammeln und an den Server übermitteln. Diese Architektur erlaubt eine flexible Skalierung und kann sowohl in kleinen als auch in grossen Unternehmen eingesetzt werden. Dabei ist es egal, ob die Endgeräte lokal oder in der Cloud betrieben werden.

In diesem Abschnitt wird die technische Struktur von Wazuh im Detail beschrieben, beginnend mit der Architektur, den verwendeten Technologien sowie den Implementierungsschritten, die für den Aufbau eines funktionalen Wazuh-Systems notwendig sind. Dabei wird auf die Herausforderungen und Besonderheiten eingegangen, die bei der Konfiguration und Integration in eine Cloud-basierte Infrastruktur auftreten. Zusätzlich wird der Proof of Concept erläutert, der es ermöglicht, die Leistungsfähigkeit und Flexibilität der Plattform zu evaluieren.

Dieser technische Hintergrund bildet die Grundlage für das Verständnis der weiteren Kapitel, in denen die Anforderungen an Wazuh, die Implementierung und die Evaluation des Systems im Rahmen des PoC ausführlich behandelt werden.

(Lempa, 2024) (Gupta, 2024) (NetworkChuck, 2023) (Wazuh, 2024)

6.1.1 Ausgangslage

In der heutigen digitalen Welt sehen sich Unternehmen mit einer stetig wachsenden Anzahl an Sicherheitsbedrohungen konfrontiert. Die steigende Vernetzung von Systemen, die zunehmende Verlagerung von Arbeitsprozessen in die Cloud sowie die Nutzung verschiedenster Endgeräte erhöhen die Angriffsfläche für Cyberkriminelle erheblich. Traditionelle Sicherheitslösungen, die auf isolierte Schutzmechanismen setzen, stossen an ihre Grenzen, da moderne Angriffe immer raffinierter und gezielter werden. Gleichzeitig stehen Unternehmen unter dem Druck, nicht nur gesetzliche Anforderungen im Bereich der IT-Sicherheit zu erfüllen, sondern auch ihre sensiblen Daten und Systeme zu schützen, um Reputations- und Geschäftsschäden zu vermeiden.

In diesem Zusammenhang wächst die Bedeutung von SIEM- und XDR-Systemen, die eine zentrale Überwachung und Analyse sicherheitsrelevanter Ereignisse ermöglichen. SIEM-Lösungen sammeln und korrelieren Ereignisse aus unterschiedlichen Quellen, wie Betriebssystemen, Netzwerken und Anwendungen, um Anomalien und potenzielle Bedrohungen frühzeitig zu erkennen. XDR erweitert diesen Ansatz durch eine tiefere Integration verschiedener Sicherheitsfunktionen über Endpunkte, Netzwerke und Cloud-Dienste hinweg und erhöht dadurch die Effizienz der Bedrohungserkennung und -reaktion.

Vor diesem Hintergrund bietet die Open-Source-Plattform Wazuh eine attraktive Alternative zu kostspieligen proprietären Lösungen. Wazuh kombiniert die Leistungsfähigkeit eines SIEM-Systems mit den erweiterten Funktionen eines XDR-Systems und bietet dabei eine kosteneffiziente und flexible Lösung für Unternehmen jeder Grösse. Es ermöglicht die zentrale Überwachung einer heterogenen IT-Landschaft, die sowohl On-Premise als auch Cloud-Systeme umfassen kann.

Die Entscheidung, im Rahmen dieser Diplomarbeit einen PoC durchzuführen, entstand aus dem wachsenden Interesse an Open-Source-Lösungen im IT-Sicherheitssektor. Viele Unternehmen sind bestrebt, lizenzfreie Lösungen zu integrieren, die ihnen nicht nur Flexibilität bieten, sondern auch individuell an die spezifischen Bedürfnisse angepasst werden können. Die Ausgangslage dieses Projekts ist daher die Evaluation von Wazuh als praktikable Lösung zur Bedrohungserkennung und Sicherheitsüberwachung in einem realitätsnahen Unternehmensumfeld. Im Rahmen des PoC soll das Potenzial von Wazuh für eine zukünftige Nutzung überprüft werden, wobei die Flexibilität, Skalierbarkeit und Benutzerfreundlichkeit der Plattform eine zentrale Rolle spielen.

Diese Ausgangslage schafft die Grundlage für die weiteren Schritte des Projekts: die technische Implementierung von Wazuh in einer Cloud-Umgebung, die Anbindung von Endgeräten und die Entwicklung spezifischer Regeln und Dashboards, um eine umfassende Überwachung und Bedrohungserkennung zu gewährleisten. Ziel ist es, die Effizienz der Lösung zu demonstrieren und festzustellen, ob Wazuh als Basis für eine produktive Sicherheitslösung in Unternehmensumgebungen geeignet ist.

6.1.2 Architektur

Die Architektur von Wazuh basiert auf einer zentralisierten, aber dennoch flexiblen Infrastruktur, die es ermöglicht, Sicherheitsereignisse aus einer Vielzahl von Quellen zu sammeln, zu analysieren und darauf zu reagieren. Diese modulare Architektur ist skalierbar und unterstützt den Betrieb sowohl in kleinen Umgebungen mit wenigen Endgeräten als auch in komplexen, gross angelegten IT-Infrastrukturen. Wazuh kombiniert die Funktionen eines SIEM- und XDR-Systems, um eine ganzheitliche Sicherheitsüberwachung zu ermöglichen.

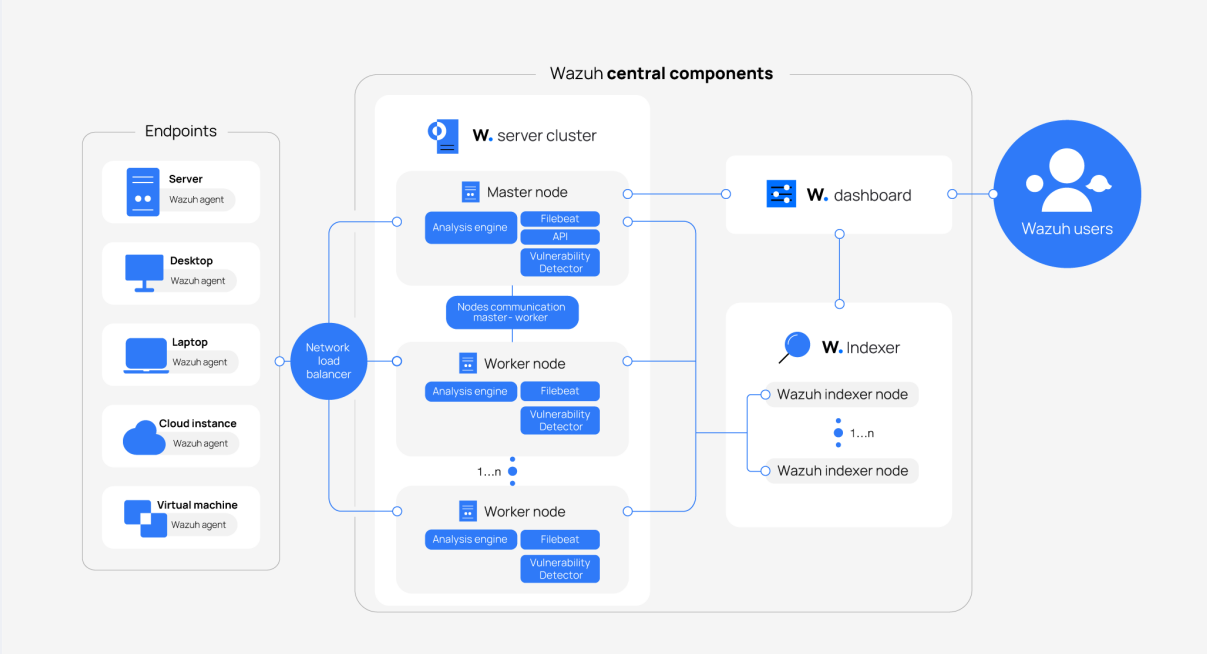

Abbildung 3: Wazuh Architektur

Im Kern besteht die Architektur aus drei Hauptkomponenten:

6.1.2.1 Wazuh Management Server

Der Management Server bildet das Herzstück der Wazuh-Architektur. Er ist verantwortlich für die zentrale Verwaltung und Koordination der gesamten Plattform. Zu seinen Hauptaufgaben gehört das Sammeln, Speichern und Analysieren von sicherheitsrelevanten Daten, die von den verschiedenen Agenten auf den Endgeräten bereitgestellt werden. Der Server führt die Korrelation von Ereignissen durch, analysiert Bedrohungen und initiiert die notwendigen Reaktionen auf sicherheitskritische Vorfälle. Darüber hinaus verwaltet er die Konfiguration der Agenten, definiert Sicherheitsregeln und speichert Logs zur späteren Analyse.
(Wazuh, 2024)

6.1.2.2 Wazuh-Agenten

Die Agenten werden auf den überwachten Endgeräten installiert. Diese Agenten sammeln sicherheitsrelevante Daten wie Systemlogs, Dateiintegritätsprüfungen, Registry-Änderungen (bei Windows-Systemen), Netzwerkaktivitäten und Anwendungsereignisse. Die gesammelten Daten werden dann an den Management Server gesendet, wo sie weiterverarbeitet und analysiert werden. Die Agenten sind leichtgewichtig und kompatibel mit verschiedenen Betriebssystemen, einschliesslich Windows, Linux und macOS. Dies ermöglicht eine breite Abdeckung in heterogenen IT-Umgebungen. (Wazuh, 2024)

6.1.2.3 Elastic Stack (Elasticsearch, Logstash, Kibana)

Wazuh nutzt den Elastic Stack zur Speicherung, Analyse und Visualisierung der gesammelten Daten. Elasticsearch dient als skalierbare Datenbank, die grosse Mengen an Logs und Ereignisdaten speichert und schnelle Abfragen ermöglicht. Logstash wird verwendet, um die Daten aus verschiedenen Quellen zu verarbeiten und an Elasticsearch weiterzuleiten. Kibana ist das Frontend der Plattform und bietet benutzerfreundliche Dashboards zur Visualisierung der sicherheitsrelevanten Ereignisse. Administratoren können über Kibana in Echtzeit auf Bedrohungen reagieren, detaillierte Berichte erstellen und komplexe Abfragen durchführen, um tiefergehende Analysen zu ermöglichen.
(Wazuh, 2024)

6.1.2.4 Datenfluss in der Wazuh-Architektur

Datenaufzeichnung: Die Wazuh-Agenten überwachen kontinuierlich die Endgeräte und sammeln sicherheitsrelevante Daten.

Datenübermittlung: Die Agenten übertragen die gesammelten Informationen an den Management Server. Dies geschieht in Echtzeit oder in regelmässigen Intervallen, je nach Konfiguration.

Datenverarbeitung: Auf dem Management Server werden die empfangenen Daten verarbeitet und analysiert. Hierbei kommen Regeln und Korrelationen zum Einsatz, um verdächtige Aktivitäten zu erkennen.

Speicherung und Analyse: Die verarbeiteten Daten werden im Elastic Stack gespeichert und können durch Administratoren über Kibana visualisiert und ausgewertet werden.

Reaktion auf Vorfälle: Bei der Erkennung von Bedrohungen können benutzerdefinierte Alerts generiert werden, die Administratoren in Echtzeit über sicherheitskritische Vorfälle informieren.

6.1.2.5 Installationsmethoden von Wazuh

Wazuh bietet verschiedene Installationsmethoden, um den individuellen Anforderungen von Unternehmen und IT-Umgebungen gerecht zu werden. Die Wahl der Installationsmethode hängt in der Regel von der Grösse der Infrastruktur, der gewünschten Skalierbarkeit und der Komplexität der zu überwachenden Systeme ab.

Single-Server-Installation:
Bei der Single-Server-Installation wird Wazuh vollständig auf einem einzelnen Server implementiert. Diese Methode eignet sich besonders gut für kleine bis mittelgrosse Umgebungen, in denen eine einfache und schnelle Implementierung gewünscht ist. Alle wesentlichen Komponenten: Management Server, Elasticsearch, Logstash und Kibana, werden auf einem einzigen physischen oder virtuellen Server installiert. Dies bietet den Vorteil einer unkomplizierten Einrichtung und Verwaltung, da alle Funktionen zentral auf einem System verfügbar sind. Diese Architektur ist ideal für Szenarien mit geringer bis mittlerer Last und eignet sich für den PoC, wie er in dieser Arbeit durchgeführt wird. Der Fokus auf die Single-Server-Installation wurde gewählt, um die Implementierung und Konfiguration effizient zu gestalten und dennoch alle Kernfunktionen von Wazuh in einer realitätsnahen Umgebung zu evaluieren.
(Wazuh, 2024)

Distributed Deployment:
Für grössere Infrastrukturen bietet sich eine verteilte Installation an, bei der die verschiedenen Wazuh-Komponenten: Management Server, Elasticsearch und Kibana, auf mehrere Server verteilt werden. Dies ermöglicht eine höhere Skalierbarkeit und Leistung, da die Last auf mehrere Maschinen aufgeteilt wird. Diese Methode wird typischerweise in Umgebungen mit Tausenden von überwachten Endgeräten eingesetzt, wo hohe Verfügbarkeit und Performanz von entscheidender Bedeutung sind.
(Wazuh, 2024)
Cluster-Installation:
Bei der Cluster-Installation wird Elasticsearch in einem Cluster ausgeführt, um Redundanz, Hochverfügbarkeit und Lastverteilung zu gewährleisten. Diese Architektur ist besonders für Unternehmen geeignet, die eine grosse Anzahl von Logs in Echtzeit verarbeiten müssen und eine hohe Ausfallsicherheit verlangen. Auch die Management Server können im Cluster-Modus ausgeführt werden, um die Verfügbarkeit weiter zu erhöhen und Ausfälle zu vermeiden.
(Wazuh, 2024)

Cloud-Installation:
Wazuh kann auch in Cloud-Umgebungen wie AWS, Microsoft Azure oder Google Cloud implementiert werden. Diese Methode bietet eine flexible Skalierbarkeit und ermöglicht es Unternehmen, die Vorteile der Cloud wie elastische Ressourcenverwaltung und Kosteneffizienz zu nutzen. Die Cloud-Installation kann sowohl als Single-Server als auch als verteilte oder Cluster-Installation erfolgen, je nach Anforderungen und Budget.
(Wazuh, 2024)

6.1.2.6 Festlegung auf die Single-Server-Installation

Für die vorliegende Diplomarbeit wurde die Single-Server-Installation gewählt. Diese Entscheidung basiert auf den Anforderungen des Proof of Concept, bei dem eine überschaubare Anzahl von Endgeräten überwacht wird und die einfache Handhabung der Installation im Vordergrund steht. Die Single-Server-Architektur ermöglicht eine schnelle Einrichtung, eine effiziente Konfiguration der Agenten und eine zentrale Verwaltung aller sicherheitsrelevanten Ereignisse. Zudem bietet sie ausreichend Leistung, um die für den PoC notwendigen Funktionen zu testen und zu evaluieren, ohne die Komplexität einer verteilten oder Cloud-basierten Lösung zu haben.

Mit dieser Architektur kann das volle Potenzial von Wazuh demonstriert werden, während die Ressourcennutzung überschaubar bleibt. Die Erfahrungen aus dieser Installation liefern wertvolle Erkenntnisse für eine mögliche spätere Skalierung auf eine verteilte oder Cloud-basierte Architektur, sollten grössere Umgebungen überwacht werden müssen.

6.1.3 PoC

Der Proof of Concept in dieser Diplomarbeit dient dazu, die praktische Machbarkeit und Leistungsfähigkeit der Wazuh-Plattform als zentrale Lösung für Bedrohungserkennung und Sicherheitsüberwachung zu evaluieren. Ziel dieses PoC ist es, die verschiedenen Kernkomponenten von Wazuh zu installieren, zu konfigurieren und unter realitätsnahen Bedingungen zu testen, um deren Effizienz und Skalierbarkeit in einer Unternehmensumgebung zu bewerten. Durch diesen praktischen Ansatz soll ermittelt werden, ob Wazuh als langfristige Lösung für die IT-Sicherheitsüberwachung verwendet werden kann.

6.1.3.1 Ziele des PoC:

Die zentralen Ziele des PoC sind:

Installation und Konfiguration: Eine voll funktionsfähige Installation und Konfiguration des Wazuh-Systems auf einem einzigen Server (Single-Server-Installation), der sowohl die Management- als auch die Überwachungsfunktionen übernimmt.

Anbindung von Endgeräten: Die Installation von Wazuh-Agenten auf verschiedenen Endgeräten (Windows, Linux, macOS), um ein heterogenes Unternehmensumfeld zu simulieren. Diese Endgeräte werden kontinuierlich überwacht, wobei sicherheitsrelevante Ereignisse wie Log-Dateien, Systemintegrität und Schwachstellen erkannt und analysiert werden.

Entwicklung benutzerdefinierter Regeln: Die Implementierung spezifischer, auf das Unternehmen zugeschnittener Sicherheitsregeln, um Bedrohungen wie Malware, unbefugte Zugriffsversuche und Systemschwachstellen frühzeitig zu erkennen. Es wird dabei angestrebt, mindestens drei Regeln zu implementieren, die sicherheitsrelevante Bedrohungsszenarien in der simulierten Umgebung abdecken.

Generierung von Alerts: Die Konfiguration benutzerdefinierter Alerts, die bei der Erkennung von sicherheitskritischen Ereignissen automatisch ausgelöst werden. Die Alerts sollen Administratoren in Echtzeit benachrichtigen, um eine schnelle Reaktion auf Bedrohungen zu ermöglichen.

Erstellung von Dashboards: Die Entwicklung benutzerfreundlicher Dashboards zur Visualisierung der Sicherheitslage in Echtzeit. Diese Dashboards sollen sicherheitsrelevante Informationen wie die Anzahl der erkannten Bedrohungen, den Systemstatus und die Verteilung von Sicherheitsvorfällen übersichtlich darstellen.

6.1.3.2 Durchführung des PoC:

Der PoC umfasst folgende Schritte:

Phase 1: Einrichtung der Infrastruktur
Zunächst wird der Wazuh-Management-Server auf einem einzigen Server in einer Cloud-Umgebung installiert und konfiguriert. Diese Single-Server-Installation dient dazu, die Grundlage für alle weiteren Schritte des Projekts zu legen. Dabei werden auch die notwendigen Komponenten des Elastic Stack (Elasticsearch, Kibana) eingerichtet, um die Verarbeitung und Visualisierung der gesammelten sicherheitsrelevanten Daten zu gewährleisten.

Phase 2: Installation und Anbindung der Agenten
Nach der erfolgreichen Einrichtung des Management-Servers werden Wazuh-Agenten auf verschiedenen Endgeräten installiert. Die Agenten werden auf gängigen Betriebssystemen (Windows, Linux, macOS) konfiguriert und mit dem zentralen Server verbunden. Diese Phase dient der Überwachung eines realistischen Unternehmensumfelds, in dem die Endgeräte regelmässig sicherheitsrelevante Daten an den Management-Server senden.

Phase 3: Entwicklung von Sicherheitsregeln und Alerts
In dieser Phase werden spezifische Sicherheitsregeln entwickelt, die auf typische Bedrohungsszenarien wie fehlgeschlagene Anmeldeversuche, verdächtige Dateiaktivitäten oder Schwachstellen in Betriebssystemen abzielen. Die entwickelten Regeln werden getestet, um ihre Wirksamkeit zu validieren und eine ausgewogene Balance zwischen der Erkennung realer Bedrohungen und der Minimierung von Fehlalarmen (False Positives) zu gewährleisten. Darüber hinaus werden benutzerdefinierte Alerts konfiguriert, die Administratoren über sicherheitsrelevante Ereignisse informieren.

Phase 4: Entwicklung von Dashboards
In dieser Phase werden benutzerfreundliche Dashboards in Kibana erstellt, die eine zentrale Übersicht über alle sicherheitsrelevanten Aktivitäten und Trends bieten. Diese Dashboards sollen interaktive Visualisierungen enthalten, die eine schnelle Analyse der Bedrohungslage ermöglichen und Administratoren dabei unterstützen, tiefere Einblicke in sicherheitsrelevante Ereignisse zu gewinnen.

Phase 5: Tests und Evaluierung
Abschliessend werden alle implementierten Komponenten getestet, um die Funktionalität, Skalierbarkeit und Benutzerfreundlichkeit der Wazuh-Plattform zu bewerten. Dabei wird insbesondere darauf geachtet, wie effizient die Plattform sicherheitsrelevante Ereignisse erfasst, analysiert und meldet.

6.1.3.3 Erwartete Ergebnisse

Der PoC soll zeigen, dass Wazuh als flexible und leistungsfähige Open-Source-Sicherheitslösung für den Einsatz in Unternehmensumgebungen geeignet ist. Der Fokus liegt dabei auf der Evaluierung der folgenden Aspekte:

Effektivität der Bedrohungserkennung: Die Plattform muss in der Lage sein, sicherheitsrelevante Ereignisse präzise und in Echtzeit zu erkennen und zu melden.

Benutzerfreundlichkeit: Wazuh muss sowohl für IT-Administratoren als auch für Sicherheitsspezialisten leicht zu bedienen sein, insbesondere in Bezug auf die Konfiguration von Regeln und die Nutzung der Dashboards.

Skalierbarkeit: Der PoC wird ebenfalls die Fähigkeit von Wazuh testen, mit einer wachsenden Anzahl von Endgeräten und steigenden Datenmengen effizient umzugehen.

Dieser Proof of Concept dient somit als Grundlage für die Bewertung, ob Wazuh als langfristige Lösung zur Bedrohungserkennung und Sicherheitsüberwachung in Unternehmen eingesetzt werden kann. Die im PoC gewonnenen Erkenntnisse sollen zudem dazu beitragen, die Entscheidungsfindung hinsichtlich der Implementierung einer produktiven Umgebung zu unterstützen.

6.2 Anforderungen

Um den erfolgreichen Einsatz von Wazuh als Sicherheitslösung im Rahmen des Proof of Concept zu gewährleisten, ist es entscheidend, klare Anforderungen zu definieren. Diese Anforderungen unterteilen sich in funktionale und nicht-funktionale Aspekte, die sowohl die technischen Fähigkeiten der Plattform als auch ihre Handhabung und Integration in eine Unternehmensumgebung betreffen.

Die funktionalen Anforderungen legen den Schwerpunkt auf die zentralen Sicherheitsfunktionen von Wazuh, wie die Überwachung von Systemen und Netzwerken, die Erkennung und Meldung von Bedrohungen sowie die zentrale Verwaltung von Logs und sicherheitsrelevanten Daten. Dazu gehören die Einrichtung benutzerdefinierter Sicherheitsregeln, die Generierung von Alerts bei sicherheitsrelevanten Ereignissen und die Erstellung von Dashboards zur Echtzeit-Überwachung.

Die nicht-funktionalen Anforderungen befassen sich mit der Leistung, Skalierbarkeit und Benutzerfreundlichkeit der Plattform. Auch die Flexibilität der Plattform, insbesondere hinsichtlich der Anpassbarkeit von Regeln und Dashboards an spezifische Unternehmensanforderungen, ist ein wichtiger Bestandteil der nicht-funktionalen Anforderungen.

In diesem Kapitel werden sowohl die funktionalen als auch die nicht-funktionalen Anforderungen an das Wazuh-System detailliert beschrieben, um die Zielsetzungen des PoC klar zu definieren und die Bewertungsgrundlage für die spätere Evaluation der Plattform zu schaffen. Diese Anforderungen bilden die Grundlage für die Implementierung, die im Rahmen des PoC überprüft und validiert wird.

6.2.1 Funktionale Anforderungen

Die funktionalen Anforderungen an das Wazuh-System umfassen die zentralen Sicherheitsfunktionen, die im Rahmen des Proof of Concept (PoC) implementiert und getestet werden. Diese Anforderungen definieren, welche konkreten Sicherheitsaufgaben das System erfüllen muss, um die Überwachung und Bedrohungserkennung effektiv zu gewährleisten. Die folgenden Aspekte stehen im Mittelpunkt:

6.2.1.1 Zentrale Überwachung von Logs

Das Wazuh-System muss in der Lage sein, Logs von allen überwachten Endgeräten zentral zu sammeln, zu speichern und zu verarbeiten. Diese Logs umfassen Systemlogs, Anwendungslogs und sicherheitsrelevante Ereignisse, die von den installierten Wazuh-Agenten auf den Endgeräten gesammelt werden. Es muss eine Echtzeit-Verarbeitung der Logs erfolgen, um sicherzustellen, dass sicherheitskritische Ereignisse schnell erkannt und gemeldet werden. Die Logs müssen durchsuchbar sein, sodass Administratoren bei Bedarf spezifische Informationen schnell abfragen können.

6.2.1.2 Erkennung von Sicherheitsbedrohungen

Eine der Kernfunktionen von Wazuh besteht darin, sicherheitsrelevante Bedrohungen zu erkennen. Dies erfolgt durch die Anwendung vordefinierter und benutzerdefinierter Regeln, die die gesammelten Logs und Systemereignisse analysieren. Die Regeln müssen sicherheitskritische Aktivitäten wie Anmeldeversuche, verdächtige Netzwerkverbindungen, Änderungen an Systemdateien und andere potenzielle Bedrohungen identifizieren. Die Funktionalität der Bedrohungserkennung ist der Schlüssel zur Effizienz des Systems und wird massgeblich durch die Präzision der entwickelten Regeln bestimmt.

6.2.1.3 Alerts bei sicherheitsrelevanten Ereignissen

Das System muss in der Lage sein, automatische Alerts zu generieren, wenn sicherheitsrelevante Ereignisse erkannt werden. Diese Alerts müssen an Administratoren oder Sicherheitsexperten weitergeleitet werden, um eine schnelle Reaktion auf Bedrohungen zu ermöglichen. Die Alerts sollen konfigurierbar sein, sodass verschiedene Bedrohungstypen mit unterschiedlichen Prioritäten versehen und über verschiedene Kommunikationskanäle (z.B. E-Mail, SMS oder ein Dashboard) übermittelt werden können. Ein effektives Alert-System minimiert Fehlalarme (False Positives), um die Arbeitsbelastung für Administratoren zu reduzieren und sicherzustellen, dass nur relevante Ereignisse gemeldet werden.

6.2.1.4 Erstellung benutzerdefinierter Regeln

Um den spezifischen Anforderungen eines Unternehmens gerecht zu werden, muss das Wazuh-System die Möglichkeit bieten, benutzerdefinierte Sicherheitsregeln zu erstellen. Diese Regeln sollen flexibel und anpassbar sein, sodass Unternehmen ihre eigenen Bedrohungsszenarien definieren und überwachen können. Mindestens drei benutzerdefinierte Regeln sollen im Rahmen des PoC entwickelt werden, um sicherzustellen, dass das System auf die speziellen Bedürfnisse der überwachten Umgebung eingeht.

6.2.1.5 Dashboards zur Sicherheitsüberwachung

Die Funktionalität der Dashboards ist ein entscheidender Aspekt des Wazuh-Systems. Das System muss die Möglichkeit bieten, benutzerdefinierte Dashboards zu erstellen, die eine zentrale Übersicht über die sicherheitsrelevanten Ereignisse und Bedrohungen bieten. Die Dashboards sollen interaktive Visualisierungen enthalten, die es Administratoren ermöglichen, sicherheitsrelevante Daten in Echtzeit zu überwachen und auf kritische Ereignisse schnell zu reagieren. Zu den visualisierten Informationen gehören unter anderem die Anzahl der erkannten Bedrohungen, der Zustand der überwachten Systeme sowie die Verteilung der Sicherheitsereignisse nach Priorität. Die Dashboards müssen flexibel genug sein, um den spezifischen Anforderungen der Nutzer gerecht zu werden.

6.2.1.6 Integration mit bestehenden Systemen

Eine weitere wichtige funktionale Anforderung ist die Fähigkeit von Wazuh, sich nahtlos in bestehende IT- und Sicherheitssysteme zu integrieren. Dies betrifft insbesondere die Verwendung von APIs zur Kommunikation mit anderen Sicherheitslösungen oder Systemen im Unternehmen, um Synergien zu schaffen und die Effizienz der Sicherheitsüberwachung zu maximieren.

6.2.2 Nicht-funktionale Anforderungen

Neben den funktionalen Anforderungen, die die zentralen Sicherheitsaufgaben des Wazuh-Systems definieren, spielen die nicht-funktionalen Anforderungen eine wesentliche Rolle bei der Bewertung der Leistungsfähigkeit und Benutzerfreundlichkeit des Systems. Diese Anforderungen befassen sich mit Aspekten wie Skalierbarkeit, Benutzerfreundlichkeit, Leistung und Flexibilität und sind entscheidend, um sicherzustellen, dass das System auch in realen Unternehmensumgebungen effizient arbeitet. Im Rahmen des PoC werden die folgenden nicht-funktionalen Anforderungen besonders betrachtet:

6.2.2.1 Skalierbarkeit

Ein entscheidender Faktor für den Erfolg von Wazuh als Sicherheitslösung ist seine Fähigkeit, mit einer wachsenden Anzahl von überwachten Endgeräten und steigenden Datenmengen umzugehen, ohne dass die Leistung beeinträchtigt wird. Das System muss so ausgelegt sein, dass es sowohl kleine Umgebungen mit wenigen Endgeräten als auch grosse, komplexe Infrastrukturen überwachen kann. Insbesondere muss das Wazuh-System in der Lage sein, durch das Hinzufügen weiterer Ressourcen (z.B. zusätzliche Server oder Speicher) zu skalieren, um wachsende Anforderungen zu erfüllen. Für den PoC wird getestet, wie gut das System in einer überschaubaren Umgebung skaliert und wie es auf erhöhte Datenmengen und neue Agenten reagiert.

6.2.2.2 Leistung und Effizienz

Die Leistung des Systems ist ein wesentlicher Aspekt, insbesondere in Bezug auf die Verarbeitungsgeschwindigkeit von Logs, die Echtzeit-Erkennung von Bedrohungen und die Generierung von Alerts. Wazuh muss in der Lage sein, sicherheitsrelevante Ereignisse in Echtzeit zu verarbeiten, ohne die Systemleistung zu beeinträchtigen oder Verzögerungen bei der Bedrohungserkennung zu verursachen. Darüber hinaus wird im Rahmen des PoC untersucht, wie effizient das System in Bezug auf die Nutzung von Rechen- und Speicherressourcen arbeitet. Eine hohe Leistung bei minimalem

Ressourcenverbrauch ist besonders wichtig, um den reibungslosen Betrieb in einer produktiven Umgebung sicherzustellen.

6.2.2.3 Benutzerfreundlichkeit

Die Benutzerfreundlichkeit des Wazuh-Systems ist von zentraler Bedeutung, insbesondere für IT-Administratoren und Sicherheitsspezialisten, die das System täglich nutzen werden. Eine intuitive Benutzeroberfläche, die es ermöglicht, Sicherheitsregeln und Alerts ohne tiefgehende technische Kenntnisse zu konfigurieren, ist ein entscheidender Faktor für den Erfolg der Plattform. Die Dashboards und Visualisierungstools müssen übersichtlich und leicht verständlich sein, um eine effiziente Sicherheitsüberwachung zu ermöglichen. Im Rahmen des PoC wird geprüft, wie benutzerfreundlich das System für Administratoren ist, die keine tiefgehenden Kenntnisse in der Programmierung oder Konfiguration haben.

6.2.2.4 Zuverlässigkeit und Verfügbarkeit

Die Zuverlässigkeit des Wazuh-Systems ist ein weiterer wichtiger nicht-funktionaler Aspekt. Das System muss unter verschiedenen Bedingungen stabil und fehlerfrei funktionieren, unabhängig davon, wie viele Endgeräte überwacht werden oder wie viele Daten verarbeitet werden. Insbesondere wird die Uptime des Systems im Rahmen des PoC überwacht, um sicherzustellen, dass keine Ausfallzeiten auftreten und das System eine zuverlässige Erkennung und Meldung von Bedrohungen gewährleistet. Eine hohe Verfügbarkeit ist in Unternehmensumgebungen von entscheidender Bedeutung, da jede Verzögerung oder jeder Ausfall der Sicherheitsüberwachung zu schwerwiegenden Sicherheitsvorfällen führen könnte.

6.2.2.5 Flexibilität und Anpassungsfähigkeit

Wazuh muss die Möglichkeit bieten, sich flexibel an die individuellen Anforderungen eines Unternehmens anzupassen. Dies umfasst die Erstellung und Anpassung von Sicherheitsregeln, Alerts und Dashboards. Darüber hinaus muss das System erweiterbar sein, sodass es in der Lage ist, auf neue Bedrohungen zu reagieren, indem es beispielsweise aktuelle Threat Intelligence integriert. Die Fähigkeit zur schnellen Anpassung an sich ändernde Sicherheitsanforderungen ist ein entscheidender Vorteil einer Open-Source-Lösung wie Wazuh und wird im PoC besonders evaluiert.

6.2.2.6 Integrationsfähigkeit

Ein weiteres nicht-funktionales Kriterium ist die Fähigkeit von Wazuh, sich nahtlos in bestehende IT- und Sicherheitssysteme zu integrieren. Dies betrifft insbesondere die Nutzung von APIs und Schnittstellen, um Daten von anderen Systemen zu empfangen oder sicherheitsrelevante Informationen an diese weiterzugeben. Diese Integrationsfähigkeit ist entscheidend, um Synergien mit bereits vorhandenen Sicherheitslösungen zu schaffen und die Gesamtleistung der Sicherheitsarchitektur eines Unternehmens zu verbessern.

6.2.2.7 Wartbarkeit und Erweiterbarkeit

Das System muss einfach wartbar sein und regelmässige Updates oder Patches dürfen den laufenden Betrieb nicht beeinträchtigen. Darüber hinaus sollte Wazuh so aufgebaut sein, dass es bei Bedarf einfach erweitert werden kann, etwa durch neue Sicherheitsmodule oder die Integration zusätzlicher Datenquellen. Dies erfordert eine gut dokumentierte API und eine klare Struktur, die es Administratoren ermöglicht, das System kontinuierlich zu aktualisieren und anzupassen, ohne den Betrieb zu unterbrechen.

6.2.3 Erfolgskriterien

Die Erfolgskriterien des Proof of Concept sind entscheidend, um die Effektivität und Eignung von Wazuh als zentrale Plattform für Bedrohungserkennung und Sicherheitsüberwachung zu bewerten. Diese Kriterien dienen dazu, die Erfüllung der funktionalen und nicht-funktionalen Anforderungen zu überprüfen und festzustellen, ob Wazuh die erwarteten Leistungen in einer realitätsnahen Umgebung erbringen kann. Die folgenden Erfolgskriterien wurden festgelegt:

6.2.3.1 Reibungslose Installation und Konfiguration

Ein wesentlicher Indikator für den Erfolg des PoC ist die problemlose Installation und Konfiguration des Wazuh-Systems. Dies umfasst die erfolgreiche Einrichtung des Management-Servers und die Anbindung der Wazuh-Agenten auf den verschiedenen Endgeräten (Windows, Linux, macOS). Der Installationsprozess muss klar dokumentiert sein und die Konfiguration so durchgeführt werden, dass die Kernfunktionen des Systems sofort einsatzbereit sind. Die Verbindungen zwischen Server und Agenten müssen stabil sein, und die Kommunikation muss kontinuierlich ohne Unterbrechungen ablaufen.

6.2.3.2 Effektive Bedrohungserkennung und minimale Fehlalarme

Die Erkennung von Sicherheitsbedrohungen ist eines der Hauptziele des Wazuh-Systems. Der Erfolg des PoC wird daran gemessen, wie gut das System in der Lage ist, sicherheitsrelevante Ereignisse präzise zu erkennen und rechtzeitig zu melden. Wichtige Bedrohungsszenarien müssen durch die vordefinierten und benutzerdefinierten Sicherheitsregeln abgedeckt werden, und es darf zu keinen signifikanten Fehlalarmen (False Positives) kommen, die die Arbeit der Administratoren unnötig belasten. Die Wirksamkeit der Bedrohungserkennung wird durch simulierte Angriffsszenarien überprüft, bei denen das System mindestens drei Bedrohungsszenarien erfolgreich erkennen und entsprechende Alerts auslösen muss.

6.2.3.3 Benutzerdefinierte Regeln und Alerts

Der Erfolg des PoC wird auch an der Flexibilität und Anpassungsfähigkeit der benutzerdefinierten Regeln und Alerts gemessen. Mindestens drei benutzerdefinierte Regeln sollen entwickelt und getestet werden, um sicherzustellen, dass das System spezifische Bedrohungsszenarien effektiv abdeckt. Die Konfiguration von Alerts muss so durchgeführt werden, dass Administratoren in Echtzeit über sicherheitskritische Ereignisse informiert werden und schnell reagieren können. Die Benutzerfreundlichkeit dieser Regeln und Alerts – sowohl in Bezug auf deren Erstellung als auch ihre Anpassung an spezifische Anforderungen – ist ein weiteres Kriterium für den Erfolg.

6.2.3.4 Zuverlässigkeit und Systemstabilität

Ein weiteres zentrales Erfolgskriterium ist die Zuverlässigkeit und Stabilität des Wazuh-Systems. Während des PoC muss das System eine Uptime von mindestens 99 % aufweisen, was bedeutet, dass es stabil und ohne nennenswerte Ausfälle arbeitet. Dies ist besonders wichtig, da Unterbrechungen in der Sicherheitsüberwachung zu verpassten Bedrohungen führen können. Das System muss auch unter unterschiedlichen Lastbedingungen zuverlässig funktionieren, insbesondere bei der Verarbeitung grosser Mengen an Log-Daten von mehreren Endgeräten.

6.2.3.5 Skalierbarkeit des Systems

Ein wichtiger Faktor für den Erfolg des PoC ist die Skalierbarkeit von Wazuh. Im Rahmen des PoC wird getestet, wie gut das System mit einer wachsenden Anzahl von Endgeräten und steigenden Datenmengen umgehen kann. Die erfolgreiche Implementierung des Systems muss zeigen, dass Wazuh durch das Hinzufügen von Ressourcen (z.B. weitere Server oder Speicher) skaliert werden kann, ohne dass die Leistung oder Stabilität beeinträchtigt wird. Für den PoC wird erwartet, dass das System in einer Umgebung mit mindestens zehn überwachten Endgeräten effizient arbeitet.

6.2.3.6 Erstellung und Nutzung von Dashboards

Ein weiteres Erfolgskriterium ist die Entwicklung und Nutzung von Dashboards zur Visualisierung sicherheitsrelevanter Ereignisse in Echtzeit. Die Dashboards müssen mindestens fünf relevante Visualisierungen enthalten, die klare und verständliche Informationen über die Sicherheitslage liefern. Die Dashboards müssen benutzerfreundlich und anpassbar sein, um den spezifischen Bedürfnissen der Administratoren gerecht zu werden.

6.2.3.7 Benutzerfreundlichkeit und Bedienbarkeit

Die Benutzerfreundlichkeit des Systems ist ein weiteres Kriterium, das den Erfolg des PoC beeinflusst. Die Konfiguration der Agenten, das Erstellen von Regeln und Alerts sowie die Verwaltung von Dashboards müssen intuitiv und ohne tiefgehende technische Kenntnisse möglich sein. Das Feedback von IT-Administratoren und Sicherheitsspezialisten, die das System testen, wird in die Bewertung der Benutzerfreundlichkeit einfliessen. Ein erfolgreiches System zeichnet sich durch eine einfache Bedienbarkeit und eine kurze Einarbeitungszeit für Administratoren aus.

6.2.3.8 Integration mit bestehenden Systemen

Wazuh muss sich erfolgreich in bestehende IT- und Sicherheitssysteme integrieren lassen, um den Anforderungen des Unternehmens gerecht zu werden. Ein Erfolgskriterium ist die nahtlose Integration mit anderen Sicherheitslösungen oder Managementsystemen über standardisierte APIs. Dies ermöglicht Synergien zwischen verschiedenen Systemen und verbessert die Effizienz der Sicherheitsüberwachung.

6.2.3.9 Zusammenfassung der Erfolgskriterien:

Der Erfolg des Proof of Concept wird anhand dieser klar definierten Kriterien gemessen. Diese Kriterien decken sowohl die technischen Anforderungen als auch die Benutzerfreundlichkeit und Flexibilität des Systems ab. Die erfolgreiche Erfüllung dieser Kriterien bestätigt, dass Wazuh als leistungsfähige, skalierbare und anpassbare Lösung zur Bedrohungserkennung und Sicherheitsüberwachung in einer Unternehmensumgebung eingesetzt werden kann.

6.3 Implementierung

Die Implementierung des Wazuh-Systems im Rahmen dieses Proof of Concept stellt den zentralen praktischen Teil der Diplomarbeit dar. In diesem Kapitel wird der technische Aufbau des Systems detailliert beschrieben, beginnend mit der Installation und Konfiguration der Wazuh-Plattform auf einem einzelnen Server. Dabei wird die Single-Server-Architektur genutzt, um die Wazuh-Komponenten effizient zu integrieren und ein funktionierendes System zu schaffen, das alle Anforderungen an die Bedrohungserkennung und Sicherheitsüberwachung erfüllt.

Der Implementierungsprozess gliedert sich in mehrere Phasen, die nacheinander durchgeführt werden. Diese Phasen umfassen die Installation des Management-Servers, die Anbindung von Agenten auf verschiedenen Endgeräten, die Konfiguration von Sicherheitsregeln und Alerts sowie die Erstellung benutzerdefinierter Dashboards für die Sicherheitsüberwachung. Jeder Schritt des Implementierungsprozesses wird detailliert dokumentiert, um eine klare Übersicht über die Vorgehensweise zu geben und sicherzustellen, dass das System optimal konfiguriert ist.

Besonderes Augenmerk liegt auf der Anpassbarkeit des Systems, der Flexibilität der Sicherheitsregeln sowie der Benutzerfreundlichkeit der Dashboards. Ziel der Implementierung ist es, eine voll funktionsfähige Umgebung zu schaffen, die in der Lage ist, sicherheitsrelevante Ereignisse in Echtzeit zu überwachen, Bedrohungen zu erkennen und entsprechende Reaktionen auszulösen. Dieser Implementierungsprozess bildet die Grundlage für die anschliessende Evaluierung des Systems, bei der die Leistungsfähigkeit und Skalierbarkeit von Wazuh umfassend getestet werden.

In den folgenden Abschnitten wird der gesamte Implementierungsprozess Schritt für Schritt beschrieben, um den praktischen Aufbau des Systems transparent zu machen und alle wichtigen Entscheidungen und Konfigurationen zu erläutern.

6.3.1 Installation und Konfiguration

Die Installation und Konfiguration des Wazuh-Systems wurde auf einem Hetzner CX42-Root-Server durchgeführt. Der Server bietet 160 GB SSD-Speicher, 8 vCPUs und 16 GB RAM für 15,90 Euro pro Monat, was angesichts der Ressourcen und Anforderungen des Wazuh-Systems als kostengünstige und leistungsfähige Lösung gewählt wurde. Hetzner war zudem die präferierte Wahl, da bereits andere Services dort gehostet werden, was die Verwaltung vereinfacht, und die Infrastruktur konsolidiert.

6.3.1.1 Wahl des Hosting-Providers und Servers

Der CX42-Server von Hetzner bietet eine solide Grundlage für den Betrieb eines Wazuh-Management-Servers in einer Single-Server-Architektur. Der Vorteil dieses Setups liegt in der geringen Komplexität und den niedrigen Kosten, was für den Proof of Concept (PoC) ideal ist. Die technischen Spezifikationen des Servers – insbesondere der grosse SSD-Speicher und die hohe Anzahl an vCPUs – ermöglichen eine schnelle Verarbeitung von Logs und Sicherheitsereignissen, während die 16 GB RAM genügend Spielraum für die simultane Nutzung von Elasticsearch und Kibana bietet.

6.3.1.2 Installationsprozess

Die Installation folgte dem offiziellen Wazuh Quickstart Guide, was eine strukturierte und bewährte Vorgehensweise sicherstellt. Die wichtigsten Schritte des Installationsprozesses sind unten beschrieben. (Wazuh, 2024)

Schritt 1: Vorbereitung des Servers
Nach der Bereitstellung des Hetzner-Servers wurde Ubuntu 22.04 LTS als Betriebssystem installiert, da es sich um eine gut unterstützte und stabile Linux-Distribution handelt. Vor der eigentlichen Installation von Wazuh wurden einige Vorbereitungen getroffen, um das System auf den neuesten Stand zu bringen:

Systemaktualisierung:
«*sudo apt-get update*» und «*sudo apt-get upgrade*» sorgten dafür, dass alle Pakete auf dem neuesten Stand sind.

Installation von benötigten Paketen: Programme wie «*curl*» und «*wget*» wurden installiert, um den Download und die Installation von Wazuh zu ermöglichen.

Schritt 2: Installation von Wazuh
Die Installation von Wazuh selbst erfolgte über ein automatisiertes Installationsskript, das sowohl den Wazuh-Manager als auch Elasticsearch und Kibana installiert. Das Skript stellt sicher, dass alle Abhängigkeiten auf dem Server korrekt eingerichtet werden und die relevanten Dienste nach der Installation automatisch gestartet werden.

Das Skript wurde mit folgendem Befehl heruntergeladen und ausgeführt:

curl -sO https://packages.wazuh.com/4.9/wazuh-install.sh && sudo bash ./wazuh-install.sh -a

Das Skript führte die Installation der folgenden Komponenten durch:

Wazuh-Manager: Der zentrale Dienst, der die von den Agenten gesammelten Daten verarbeitet und Sicherheitsregeln anwendet.

Elasticsearch: Die Komponente, die für die Indexierung und Speicherung der sicherheitsrelevanten Ereignisdaten verantwortlich ist.

Kibana: Die Benutzeroberfläche, die es Administratoren ermöglicht, sicherheitsrelevante Ereignisse zu visualisieren und Dashboards zu erstellen.

Nachdem das Skript die Installation abgeschlossen hatte, zeigte die Konsole eine Zusammenfassung der Installation und die Anmeldedaten für das Wazuh-Dashboard an:

INFO: --- Summary ---
INFO: You can access the web interface https://<wazuh-dashboard-ip>
User: admin
Password: <ADMIN_PASSWORD>
INFO: Installation finished.

Der Zugriff auf das Wazuh-Dashboard erfolgt über «https://siem.daenzer.swiss», wobei das SSL-Zertifikat beim ersten Besuch eine Warnung erzeugt, da es nicht von einer vertrauenswürdigen Zertifizierungsstelle stammt. Dies ist ein erwartetes Verhalten, und das Zertifikat kann als Ausnahme hinzugefügt werden, oder es kann ein Zertifikat einer vertrauenswürdigen Zertifizierungsstelle eingerichtet werden.

Zugangsdaten: Die Anmeldedaten für das Wazuh-Dashboard (Benutzername und Passwort) wurden in der Datei «wazuh-passwords.txt» gespeichert.

Schritt 3: Firewall-Konfiguration bei Hetzner
Nach der erfolgreichen Installation wurde die Firewall des Root-Servers bei Hetzner entsprechend konfiguriert, um den Zugriff auf das System nur auf die notwendigen Dienste zu beschränken. Diese Sicherheitsmassnahmen sind entscheidend, um unbefugte Zugriffe und potenzielle Bedrohungen zu verhindern.

Folgende Firewall-Regeln wurden festgelegt:

Port 443 (HTTPS): Wazuh Frontend.

Port 1514 und 1515 (für Wazuh-Agenten): Diese Ports sind für die Kommunikation zwischen den Wazuh-Agenten und dem Management-Server erforderlich und wurden entsprechend konfiguriert.

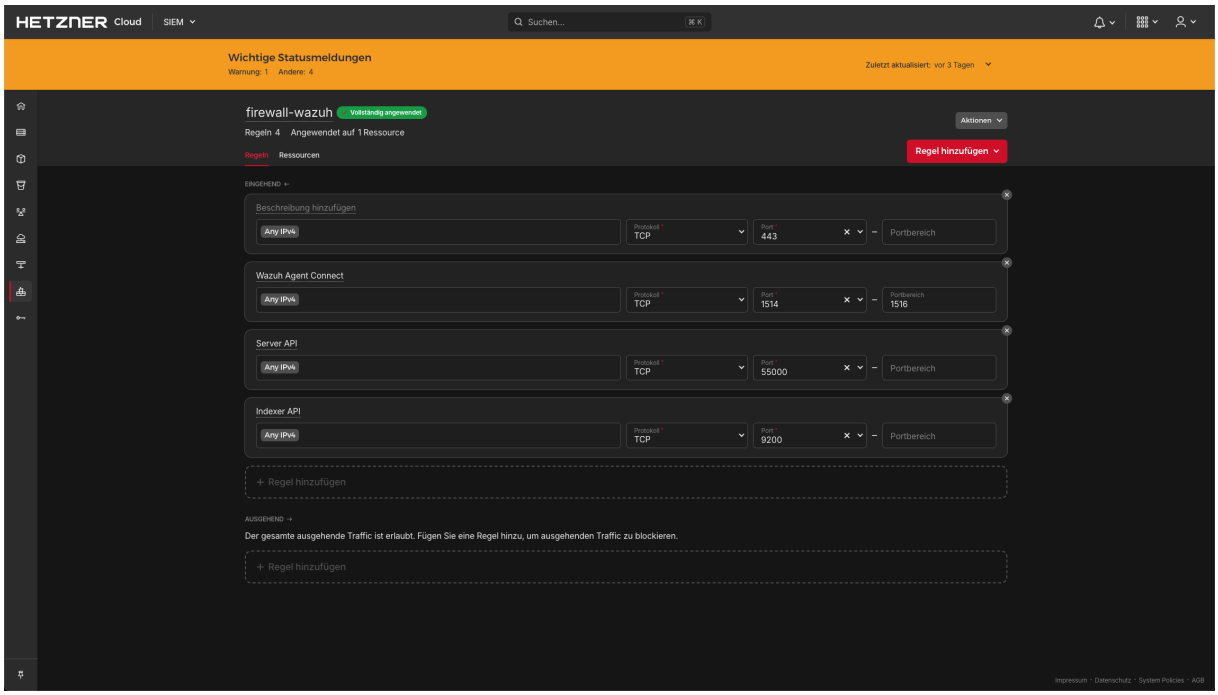

Abbildung 4: Hetzner Firewall Einstellungen

Schritt 4: Absicherung des Zugriffs über Cloudflare
Die Wazuh-Weboberfläche wurde durch die Verwendung von Cloudflare zusätzlich abgesichert. Cloudflare dient als Proxy und bietet Schutz vor verschiedenen Bedrohungen, darunter automatisierte Angriffe und bösartige Bots. Der Zugriff auf das Dashboard erfolgt über die Domain «https://siem.daenzer.swiss», die über Cloudflare geroutet wird.

Folgende Web Application Firewall (WAF)-Regeln wurden festgelegt:

Cloudflare Managed Ruleset: Diese Regeln bieten Schutz vor den häufigsten Sicherheitsbedrohungen wie SQL-Injections, Cross-Site-Scripting (XSS) und anderen Web-Schwachstellen.

Blockierung bekannter Bots: Diese Regel verhindert, dass bekannte bösartige Bots auf die Weboberfläche zugreifen.

Challenge für niedrigen Trustscore: Bei Zugriffen mit einem niedrigen Trustscore (basierend auf Cloudflare-Algorithmen) wird der Nutzer durch ein Captcha verifiziert.

Geoblocking: Der Zugriff auf das Dashboard wurde geografisch eingeschränkt, indem Zugriffe ausserhalb Europas blockiert wurden. Nutzer ausserhalb der Schweiz müssen zusätzlich eine CAPTCHA-Challenge bestehen.

Diese WAF-Regeln bieten eine mehrschichtige Sicherheitsarchitektur, die sowohl automatisierte als auch gezielte Angriffe abwehrt und nur legitimen Nutzern den Zugriff ermöglicht.

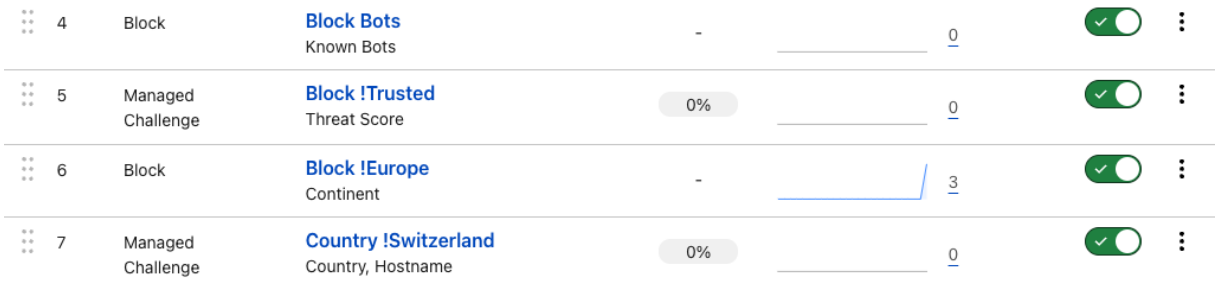

Abbildung 5: Cloudflare WAF Regeln

Schritt 5: Anpassung der Server-Firewall für Cloudflare
Um sicherzustellen, dass alle Zugriffe auf das Dashboard nur über Cloudflare erfolgen, wurde die Server-Firewall entsprechend angepasst. Der Zugriff auf Port 443 ist nun ausschliesslich für Cloudflare-IP-Ranges erlaubt. Dies verhindert direkte Zugriffe auf den Server und stellt sicher, dass alle Anfragen durch die Schutzmechanismen von Cloudflare gefiltert werden.

Diese Konfiguration stellt eine zusätzliche Sicherheitsebene dar, die den direkten Zugriff auf den Wazuh-Server verhindert und gleichzeitig Schutz vor DDoS-Angriffen bietet.

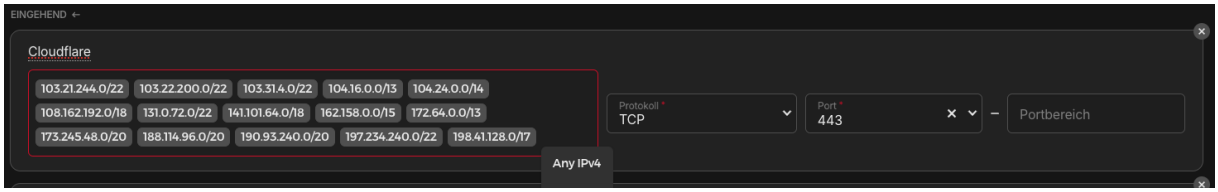

Abbildung 6: Hetzner Firewall Cloudflare

Zusammenfassung der Installation und Konfiguration
Nach Abschluss dieser Schritte war das Wazuh-System vollständig einsatzbereit. Die Installation wurde erfolgreich durchgeführt, und das System ist in der Lage, sicherheitsrelevante Daten zu erfassen, zu verarbeiten und visuell in Dashboards darzustellen. Dank der umfassenden Sicherheitskonfiguration, einschliesslich Firewall-Regeln und Cloudflare-WAF, ist der Zugriff auf das System geschützt und nur für autorisierte Nutzer möglich.

6.3.2 Entwicklung von Regeln und Alerts

Die Entwicklung von Sicherheitsregeln und Alerts ist ein zentraler Bestandteil eines jeden SIEM-Systems, und auch bei der Implementierung von Wazuh spielt dieser Aspekt eine entscheidende Rolle. Die Hauptaufgabe dieses Kapitels ist es, zu beschreiben, wie benutzerdefinierte Regeln und Alerts entwickelt wurden, um Bedrohungen zu erkennen und sicherzustellen, dass das Wazuh-System effektiv auf sicherheitsrelevante Ereignisse reagiert.

6.3.2.1 Herausforderung: Mangel an sicherheitsrelevanten Logs

Eine unerwartete Herausforderung, die sich während der Implementierung des Wazuh-Systems herausstellte, war der Mangel an sicherheitsrelevanten Logs und kritischen Ereignissen. Obwohl ein Agent erfolgreich auf meinem MacBook installiert wurde und das System mehrere Tage lief, blieben Alerts und kritische Log-Einträge aus. Dieser Mangel an Daten stellte sich als Problem heraus.

6.3.2.2 Erste Schritte: Webhook-Integration für Alerts

Da anfänglich keine kritischen Ereignisse oder Alerts auftraten, wurde die Alert-Funktion über einen Webhook konfiguriert, um zumindest sicherzustellen, dass, falls in Zukunft sicherheitsrelevante Ereignisse auftreten, diese an einen Teams-Kanal weitergeleitet würden. Die Idee war, dass bei einem auftretenden sicherheitskritischen Vorfall eine sofortige Benachrichtigung an das Team gesendet wird, um so schnell wie möglich reagieren zu können.

Die Konfiguration der Webhooks in Wazuh ist relativ einfach und bietet eine Möglichkeit, Benachrichtigungen über verschiedene Kanäle zu senden, sobald ein definierter Sicherheitsvorfall auftritt. Dazu wurde ein Webhook erstellt, der mit einem Teams-Kanal verbunden ist, sodass jede potenzielle Bedrohung in Echtzeit an das Team gemeldet wird.

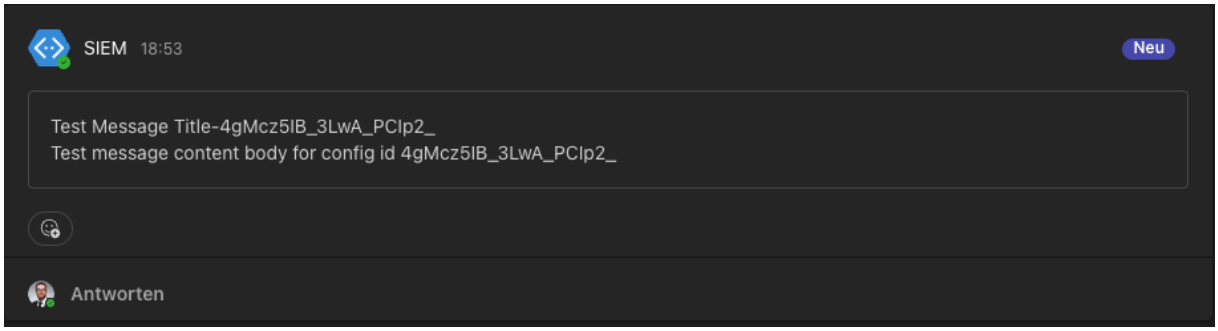

Abbildung 7: Teams Test Alert

6.3.2.3 Ausrollen der Wazuh-Agenten auf produktive Cloud-Server

Um die Datenlage zu verbessern und mehr sicherheitsrelevante Informationen zu sammeln, wurden Wazuh-Agenten auf allen produktiven Cloud-Servern ausgerollt. Diese Server sind für den Betrieb geschäftskritischer Anwendungen verantwortlich und laufen in einer Vielzahl von Konfigurationen, die von Webservern über Datenbanken bis hin zu Netzwerkdiensten reichen. Der Gedanke dahinter war, dass diese produktiven Systeme, obwohl sie ebenfalls gehärtet und gut abgesichert sind, möglicherweise mehr Log-Einträge generieren würden, da sie einer grösseren Anzahl externer Zugriffe ausgesetzt sind.

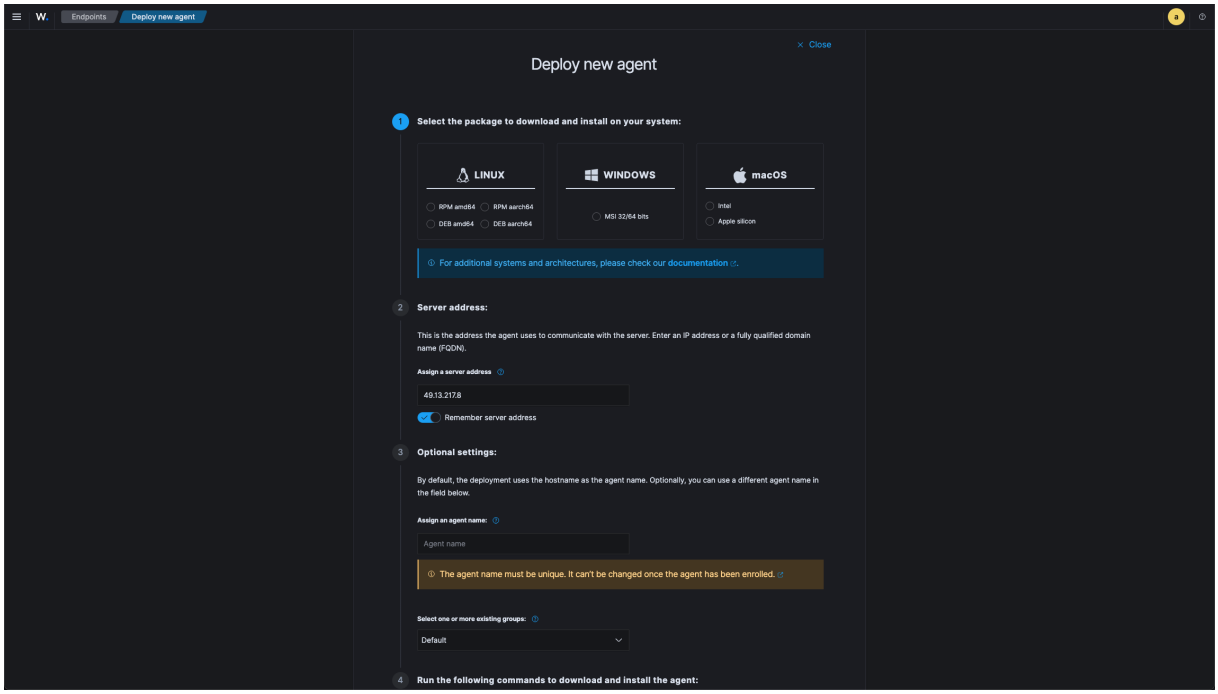

Abbildung 8: Wazuh Agent hinzufügen

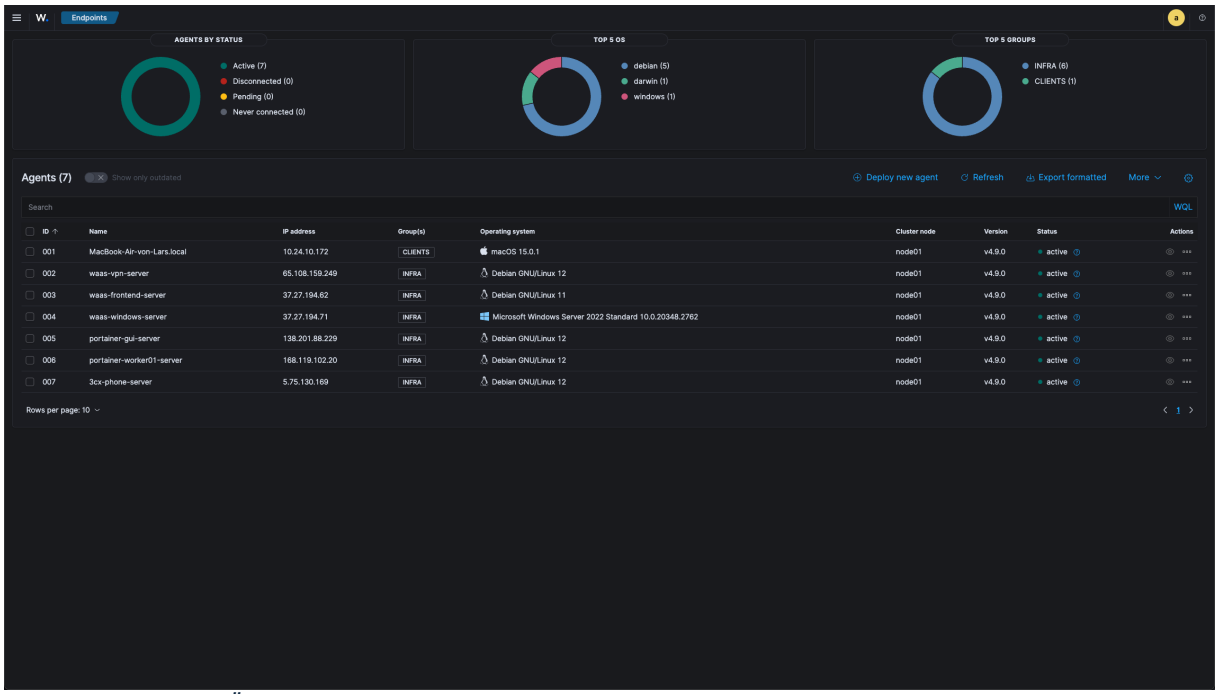

Abbildung 9: Wazuh Übersicht Agenten

6.3.2.4 Beobachtung der Standard-Dashboards

Nachdem die Wazuh-Agenten auf den produktiven Servern installiert wurden, begannen die Standard-Dashboards von Wazuh, langsam sicherheitsrelevante Informationen zu sammeln. Diese Dashboards visualisieren Ereignisse wie Login-Versuche, Netzwerkverbindungen und die Überwachung von Systemintegrität und Schwachstellen. Nun begannen sich die Dashboards allmählich mit Informationen zu füllen, da mehr und mehr Events von den verschiedenen Servern aufgezeichnet wurden.

Die Ereignisdaten, die von den Agenten gesammelt wurden, umfassen eine Vielzahl von sicherheitsrelevanten Informationen, darunter:

Login-Versuche: Erfolgreiche und fehlgeschlagene Anmeldungen auf den Servern.

Netzwerkverbindungen: Verbindungen zu den Servern, einschliesslich eingehender und ausgehender Verbindungen.

Dateiänderungen: Überwachung kritischer Systemdateien auf unautorisierte Änderungen.

Schwachstellenüberprüfung: Identifizierung potenzieller Schwachstellen in den installierten Paketen und Betriebssystemkomponenten.

Die Tatsache, dass die Dashboards sich langsam füllen, zeigt, dass das System die Ereignisse wie erwartet sammelt und verarbeitet. Allerdings bleibt das Problem bestehen: Da die Server alle gehärtet sind und durch zusätzliche Sicherheitsmechanismen wie Cloudflare geschützt werden, treten keine kritischen Events oder rote Felder in den Dashboards auf.

6.3.2.5 Warum gehärtete Systeme ein Problem darstellen

Das Problem mit gehärteten Systemen besteht darin, dass sie so konfiguriert sind, dass sie von vornherein nur minimale Angriffspunkte bieten. Massnahmen wie die Deaktivierung unnötiger Dienste, die Verschlüsselung sensibler Daten und die Verwendung von Firewalls und Web Application Firewalls verhindern, dass viele Angriffe erfolgreich durchgeführt werden können. Im Fall des PoC verhindern diese Massnahmen, dass sicherheitsrelevante Ereignisse überhaupt entstehen, was natürlich aus sicherheitstechnischer Sicht positiv ist, aber den Test der Alert-Funktionalität und der Entwicklung von Regeln erschwert.

Die Server hinter Cloudflare befinden sich zusätzlich in einem besonders geschützten Netzwerksegment, das durch die WAF automatisch Angriffe blockiert, bevor sie überhaupt den Server erreichen. Dies führt zu einer Situation, in der das Wazuh-System zwar Events aufzeichnet, diese aber fast ausschliesslich aus harmlosen Ereignissen bestehen, die keine sicherheitskritischen Situationen darstellen.

6.3.2.6 Entwicklung von benutzerdefinierten Alerts

Trotz des Mangels an sicherheitskritischen Ereignissen wurde dennoch versucht, spezifische benutzerdefinierte Alerts zu entwickeln, die auf bestimmten Ereignissen basieren. Da sich jedoch keine kritischen Sicherheitsereignisse in den Logs fanden, konzentrierte sich die Entwicklung auf potenziell interessante Ereignisse wie:

Wiederholte fehlgeschlagene Login-Versuche: Ein Anzeichen für Brute-Force-Angriffe, die zwar bei gehärteten Systemen selten vorkamen, aber dennoch, als potenzielle Bedrohung betrachtet wurden.

Ungewöhnliche Netzwerkverbindungen: Ein ungewöhnliches Muster an eingehenden oder ausgehenden Verbindungen, das möglicherweise auf unautorisierte Aktivitäten hinweist.

Änderungen an kritischen Dateien: Obwohl keine unautorisierten Änderungen festgestellt wurden, wurde eine Regel erstellt, um alle Änderungen an sensiblen Systemdateien zu überwachen.

Diese benutzerdefinierten Alerts wurden mit Hilfe der Wazuh-Richtlinien entwickelt und in das bestehende Regelwerk integriert. Die Herausforderung bestand jedoch weiterhin darin, dass keine sicherheitsrelevanten Vorfälle auftraten, um diese Regeln auszulösen.

Langsame Füllung der Dashboards:
Mit der Zeit begannen die Standard-Dashboards von Wazuh, mehr Events zu verarbeiten, insbesondere von den produktiven Cloud-Servern, die durchgehend Anfragen und Aktivitäten verarbeiten. Es war beruhigend zu sehen, dass das System in der Lage war, Logs zu sammeln und sicherheitsrelevante Ereignisse in Echtzeit zu überwachen. Trotzdem blieben rote Felder und kritische Ereignisse aus. Diese Abwesenheit von sicherheitskritischen Ereignissen ist ein Indikator dafür, dass die getroffenen Sicherheitsmassnahmen effektiv sind, stellt jedoch auch eine Herausforderung dar, da die eigentliche Funktionalität der entwickelten Alerts und Regeln nicht vollständig getestet werden konnte.

6.3.2.7 Schlussfolgerung

Der Mangel an kritischen Ereignissen verdeutlicht eine wichtige Lektion: In Umgebungen mit gehärteten Systemen und umfassenden Sicherheitsmechanismen wie Cloudflare kann es schwer sein, kritische Sicherheitsereignisse zu generieren, die für Tests oder PoCs erforderlich sind. Dies stellt zwar aus Sicht der Sicherheitsstrategie eine Erfolgsgeschichte dar, bedeutet aber auch, dass die Tests der Wazuh-Alert-Funktionalität unter realen Bedingungen eingeschränkt waren.

Die Integration der Webhook-Alerts in den Teams-Kanal bleibt ein wichtiger Bestandteil des Sicherheitskonzepts. Obwohl derzeit keine kritischen Ereignisse auftreten, stellt dies sicher, dass das Team im Fall eines Vorfalls sofort benachrichtigt wird.

6.3.3 Erstellen von Dashboards

Die Erstellung von Dashboards stellt eine zentrale Komponente der Sicherheitsüberwachung in Wazuh dar. Sie ermöglicht es Administratoren, sicherheitsrelevante Daten effizient zu visualisieren, um Bedrohungen und Schwachstellen frühzeitig zu erkennen und zu priorisieren. Dashboards bieten einen strukturierten Überblick über sicherheitskritische Ereignisse und erlauben es, auf Basis von Daten fundierte Entscheidungen zu treffen. In diesem Abschnitt wird die Implementierung und Konfiguration der Dashboards innerhalb der Wazuh-Plattform erläutert, einschliesslich der spezifischen Herausforderungen, die in der betrachteten IT-Infrastruktur aufgetreten sind.

6.3.3.1 Herausforderungen durch gehärtete Systeme und Cloudflare

Ein wesentliches Problem bei der Implementierung der Dashboards war der Mangel an sicherheitskritischen Ereignissen. Diese Herausforderung ergab sich primär aufgrund der hohen Sicherheitsmassnahmen, die auf den Systemen bereits implementiert waren. Insbesondere die Kombination aus gehärteten Systemen und dem Einsatz von Cloudflare führte dazu, dass viele potenzielle Bedrohungen im Vorfeld blockiert wurden, bevor sie sicherheitsrelevante Log-Einträge oder Ereignisse in Wazuh generieren konnten.

Infolge dieser Sicherheitsvorkehrungen blieben viele der Standard-Dashboards, die auf sicherheitskritische Ereignisse wie Netzwerkangriffe oder unbefugte Zugriffsversuche ausgerichtet sind, leer. Dies schränkte die Möglichkeit ein, sicherheitskritische Vorfälle zu visualisieren und detailliert zu analysieren. Dennoch konnten durch die gezielte Aktivierung von Funktionen wie der Überwachung von Schwachstellen und der Erfassung von Login-Versuchen nützliche Daten generiert und visuell aufbereitet werden.

6.3.3.2 Überwachung von Schwachstellen und CVE-Bewertungen

Eine bedeutende Funktion, die zur Bereitstellung von verwertbaren Daten in den Dashboards aktiviert wurde, war die Schwachstellenüberwachung. Diese Funktionalität erlaubt es, Server auf bekannte Sicherheitslücken zu scannen, welche anschliessend nach dem CVE-Score kategorisiert werden. Wazuh bietet durch die Integration der Schwachstellenüberwachung die Möglichkeit, die erkannten Sicherheitslücken nach ihrem Schweregrad zu klassifizieren und diese Informationen in Dashboards darzustellen.

Nach der Aktivierung der Schwachstellenüberwachung wurde eine erhebliche Menge an Daten generiert. Diese Daten spiegelten die potenziellen Sicherheitslücken wider, die in den verschiedenen Systemen erkannt wurden. Die Schwachstellen wurden auf Basis ihrer Kritikalität nach Critical, High, Medium und Low kategorisiert, was es ermöglichte, eine sofortige visuelle Priorisierung der Bedrohungen vorzunehmen. Kritische Schwachstellen, die rot hervorgehoben wurden, gaben einen schnellen Überblick über die gravierendsten Sicherheitsrisiken.

6.3.3.3 Filterung der Schwachstellen

Ein zentrales Problem bei der Analyse der Schwachstellen war die fehlende Möglichkeit, zwischen Schwachstellen, die extern über das Internet zugänglich sind, und solchen, die nur intern relevant sind, zu unterscheiden. Da vor allem die von aussen zugänglichen Schwachstellen aus sicherheitstechnischer Sicht von höherer Relevanz sind, ist es von grosser Bedeutung, eine Möglichkeit zur Filterung dieser Informationen zu entwickeln. Derzeit zeigt das Schwachstellen-Dashboard alle erkannten Sicherheitslücken an, ohne eine klare Unterscheidung zwischen externen und internen Risiken zu ermöglichen. Diese Unterscheidung ist jedoch entscheidend, da interne Schwachstellen in vielen Fällen ein geringeres Risiko darstellen als solche, die potenziell von externen Angreifern ausgenutzt werden können.

Abbildung 10: Wazuh Schwachstellen Übersicht

Die derzeitige Herausforderung besteht darin, die Darstellung der Schwachstellen so zu optimieren, dass Schwachstellen, die für Angriffe von aussen exponiert sind, priorisiert und hervorgehoben werden. Diese Funktion würde die Effizienz der Bedrohungserkennung erheblich verbessern, indem sie es ermöglicht, auf einen Blick zu erkennen, welche Sicherheitslücken die höchste Dringlichkeit besitzen.

6.3.3.4 Heatmap für fehlgeschlagene Login-Versuche

Ein weiteres bedeutendes Dashboard, das erstellt wurde, visualisiert fehlgeschlagene Login-Versuche auf Grundlage ihrer geografischen Herkunft. Ziel dieser Visualisierung war es, potenzielle Angriffsversuche zu identifizieren und dabei zu erkennen, aus welchen geografischen Regionen diese stammen. Diese Art der Visualisierung trägt zur Verbesserung der Sicherheitslage bei, da Angriffe oft aus spezifischen Regionen ausgehen und entsprechende Gegenmassnahmen, wie z.B. Geoblocking, ergriffen werden können.

6.3.3.5 Unerwartete Ergebnisse

Obwohl davon ausgegangen wurde, dass keine Daten zu fehlgeschlagenen Login-Versuchen erfasst würden, da die Systeme durch zusätzliche Schutzmechanismen wie Cloudflare abgesichert sind, ergaben sich überraschende Erkenntnisse. Die Heatmap zeigte, dass auf der Wazuh-Instanz selbst mehrere fehlgeschlagene Anmeldeversuche registriert wurden. Diese Beobachtung führte zu einer eingehenderen Untersuchung der Konfiguration des Root-Servers, bei der festgestellt wurde, dass der Port 22 falsch konfiguriert war. Dieser war so eingestellt, dass er Verbindungen von allen IP-Adressen zuliess.

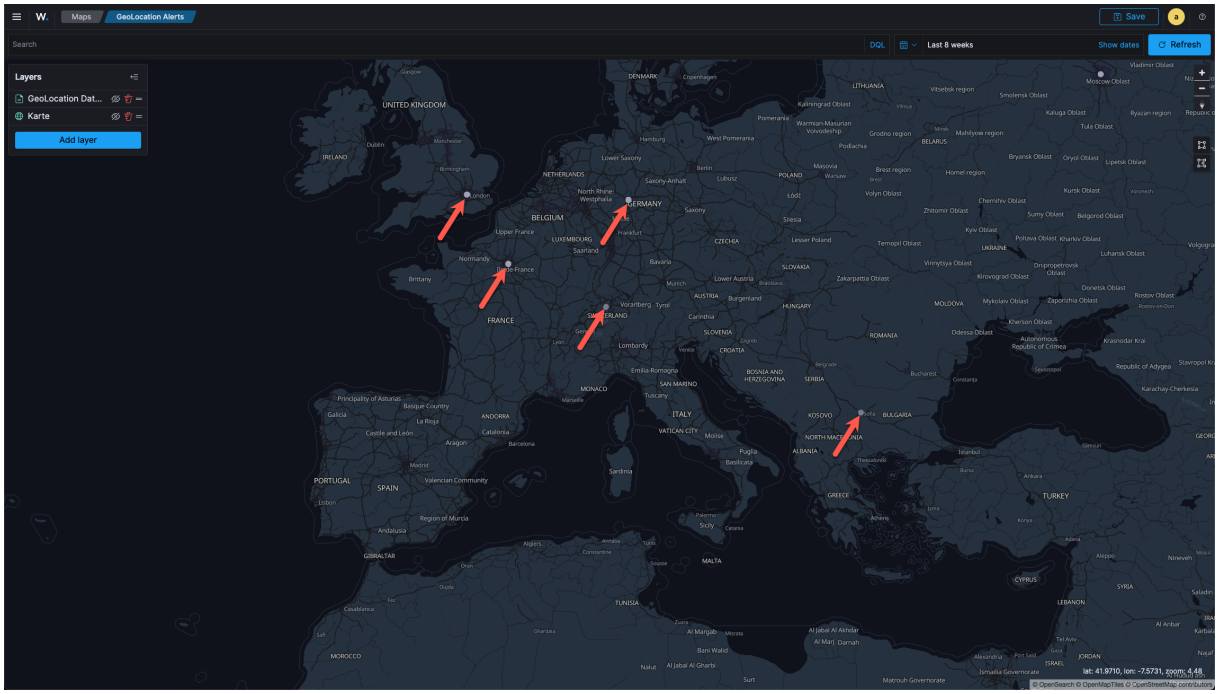

Abbildung 11: Wazuh fehlgeschlagene Logins

Diese Entdeckung ermöglichte es, die Konfiguration des Servers zu korrigieren und den Zugriff auf Port 22 auf vertrauenswürdige IP-Adressen zu beschränken. Diese Massnahme führte zu einer signifikanten Reduktion der Angriffsfläche und demonstrierte den Mehrwert der Visualisierungsfunktionen von Wazuh bei der Erkennung und Behebung von Sicherheitslücken.

6.3.3.6 Erkenntnis durch Datenvisualisierung

Die Entdeckung der falschen SSH-Konfiguration zeigte auf, wie wertvoll Dashboards als Mittel zur Verbesserung der Systemsicherheit sein können. Durch die Visualisierung der fehlgeschlagenen Anmeldeversuche in einer geografischen Heatmap konnte nicht nur ein sicherheitsrelevantes Problem identifiziert, sondern auch eine direkte Korrekturmassnahme ergriffen werden, um die Sicherheit der Infrastruktur zu erhöhen. Dies unterstreicht den Nutzen von Wazuh als Instrument zur kontinuierlichen Sicherheitsüberwachung und zur Unterstützung des Prozesses der Systemhärtung.

Im Fall der Schwachstellenvisualisierung kann der Benutzer beispielsweise auf eine Kategorie klicken (z.B. “Critical Vulnerabilities”), um detaillierte Informationen über die betroffenen Systeme, die Art der Schwachstelle und den CVE-Score zu erhalten. Diese interaktiven Elemente erlauben eine tiefgehende Analyse der Sicherheitslücken und unterstützen den Administrator bei der Entscheidung, welche Massnahmen zur Behebung der Schwachstellen ergriffen werden sollten.

6.3.3.7 Zukunftsaussichten und Optimierungsmöglichkeiten

Obwohl die aktuell implementierten Dashboards bereits wertvolle Informationen liefern, bestehen weiterhin Optimierungsmöglichkeiten. Ein zentrales Ziel für die Zukunft ist die Verbesserung der Filterfunktionen innerhalb der Schwachstellenvisualisierung, um eine klare Unterscheidung zwischen externen und internen Risiken zu ermöglichen. Dies würde es ermöglichen, Schwachstellen, die dem Internet ausgesetzt sind, klar zu identifizieren und gezielt zu priorisieren.

Darüber hinaus besteht Potenzial für die Erweiterung der Heatmap-Visualisierungen, um Angriffsversuche noch genauer zu analysieren und spezifische Muster oder Regionen zu identifizieren, aus denen verstärkt Angriffe erfolgen. Diese Informationen könnten genutzt werden, um proaktive Massnahmen wie Geoblocking oder zusätzliche Sicherheitsregeln zu implementieren.

6.3.3.8 Schlussfolgerungen

Die Erstellung von Dashboards in Wazuh hat sich als unverzichtbares Werkzeug zur Überwachung und Verbesserung der Systemsicherheit erwiesen. Trotz der Herausforderungen, die sich aus der Absicherung der Systeme durch Cloudflare und deren Härtung ergeben, konnten nützliche Einblicke gewonnen werden, insbesondere durch die Aktivierung der Schwachstellenüberwachung und die Visualisierung von Login-Versuchen.

Die Dashboards bieten einen zentralen Überblick über die sicherheitsrelevanten Ereignisse und Schwachstellen der Systeme. Besonders die Schwachstellenvisualisierung ermöglicht es, kritische Sicherheitslücken zu identifizieren und ihre Behebung priorisiert anzugehen. Gleichzeitig hat die Heatmap-Visualisierung geholfen, potenzielle Angriffsflächen zu identifizieren und Konfigurationsfehler zu korrigieren.

Insgesamt haben die Dashboards dazu beigetragen, die Systemsicherheit zu erhöhen und eine bessere Visualisierung der sicherheitsrelevanten Daten zu gewährleisten.

6.4 Evaluation

Die Evaluation des im Rahmen dieses Projekts implementierten Wazuh-Systems ist ein zentraler Bestandteil, um die Wirksamkeit und Effizienz der Plattform in einer realen Unternehmensumgebung zu bewerten. Ziel der Evaluation ist es, die Funktionalität des Systems in Bezug auf Bedrohungserkennung, Sicherheitsüberwachung und Benutzerfreundlichkeit zu analysieren. Dabei werden sowohl die implementierten Sicherheitsregeln und Alerts als auch die Dashboards auf ihre Effizienz und Nützlichkeit hin geprüft. Zusätzlich wird untersucht, inwieweit das System den Anforderungen an Skalierbarkeit, Leistung und Integrationsfähigkeit gerecht wird.

6.4.1 Besonderes Augenmerk wird auf die folgenden Aspekte gelegt

Effektivität der Bedrohungserkennung: Wie zuverlässig erkennt das System sicherheitskritische Ereignisse und wie genau können diese klassifiziert und priorisiert werden?

Benutzerfreundlichkeit und Handhabung: Ist die Benutzeroberfläche intuitiv und die Konfiguration von Regeln und Dashboards einfach genug, um auch von Administratoren ohne tiefgehende Programmierkenntnisse genutzt zu werden?

Skalierbarkeit und Leistung: Wie gut skaliert das System, wenn die Anzahl der überwachten Endgeräte und die Menge an sicherheitsrelevanten Daten steigt?

Relevanz der gesammelten Daten: Wie nützlich sind die gesammelten Informationen für die tägliche Sicherheitsüberwachung, und können Administratoren damit fundierte Entscheidungen treffen?

Die Evaluation erfolgt auf Grundlage der in den vorherigen Kapiteln implementierten Funktionen, einschliesslich der Überwachung von Schwachstellen, der Visualisierung von sicherheitsrelevanten Ereignissen und der generierten Alerts. Diese Untersuchung wird die Grundlage dafür liefern, ob und wie Wazuh langfristig als Sicherheitslösung in einer produktiven Umgebung eingesetzt werden kann.

6.4.2 Effektivität

Die Effektivität eines SIEM-Systems wie Wazuh hängt massgeblich davon ab, wie gut es sicherheitsrelevante Ereignisse erkennt, analysiert und aufbereitet. Für die Evaluation der Effektivität im Rahmen dieses Projekts wurden die Konfigurationen von Sicherheitsregeln, Alerts und Dashboards intensiv beobachtet, um zu bewerten, ob die gewünschten Ziele erreicht wurden.

6.4.2.1 Erkennung von sicherheitsrelevanten Ereignissen

Einer der Schwerpunkte der Evaluation war die Erkennung von sicherheitskritischen Ereignissen auf den überwachten Systemen. Hierbei stellte sich aufgrund der gehärteten Systeme und des umfassenden Einsatzes von Cloudflare ein erstes Problem dar: Das Fehlen sicherheitsrelevanter Logs. Diese Herausforderung ergab sich aus der Tatsache, dass viele Bedrohungen durch die Härtung der Systeme bereits auf einer höheren Schutzebene abgefangen wurden, bevor sie das Wazuh-System erreichten. Die Systemsicherheit an sich war dadurch gegeben, jedoch fehlten die

sicherheitskritischen Ereignisse, die zur Bewertung der Bedrohungserkennung herangezogen werden konnten.

Um diesem Problem entgegenzuwirken, wurden verschiedene Schwachstellen-Scans und spezifische Konfigurationen durchgeführt. Diese zeigten, dass die Funktion der Schwachstellenüberwachung in Wazuh ein wertvolles Werkzeug darstellt, um potenzielle Risiken zu erkennen. Besonders die Visualisierung der Schwachstellen nach CVE-Scores ermöglichte eine schnelle Identifikation kritischer Sicherheitslücken. Dies lieferte hilfreiche Daten zur Analyse, obwohl konkrete Bedrohungen in Form von Angriffsversuchen weitgehend ausblieben.

6.4.2.2 Überraschung durch fehlgeschlagene Login-Versuche

Ein unerwartetes Ereignis verdeutlichte jedoch die Effektivität von Wazuh in der Bedrohungserkennung. Während der Erstellung eines Dashboards zur Visualisierung von fehlgeschlagenen Login-Versuchen, die nach geografischen Regionen gefiltert wurden, stellte sich heraus, dass es trotz der gehärteten Systeme zu mehreren fehlgeschlagenen Anmeldeversuchen auf der Wazuh-Instanz selbst kam. Dies führte zu der Entdeckung, dass der Zugriff über Port 22 auf dem Root-Server fälschlicherweise nicht eingeschränkt wurde, was bedeutet, dass der Port für alle IP-Adressen offenstand.

Diese Konfigurationslücke stellte eine erhebliche Angriffsfläche dar, die durch das Dashboard sichtbar wurde. Wazuh erkannte die wiederholten Login-Versuche und visualisierte sie, was dazu führte, dass die Serverkonfiguration korrigiert werden konnte. Durch das Anpassen der Firewall-Regeln auf vertrauenswürdige IP-Adressen wurde die Angriffsfläche deutlich reduziert. Dieses Beispiel zeigt, wie Wazuh durch das Sammeln und Analysieren von sicherheitsrelevanten Daten dazu beitragen kann, sicherheitskritische Konfigurationsfehler aufzudecken und proaktiv zu beheben.

6.4.2.3 Alerts und Bedrohungspriorisierung

Obwohl das System nur wenige kritische Ereignisse generierte, konnte die Effektivität der Alert-Funktion dennoch getestet werden. Die Konfiguration der Alerts über einen Webhook ermöglichte es, sicherheitsrelevante Vorfälle in Echtzeit an einen Teams-Kanal weiterzuleiten. Trotz des Mangels an realen Bedrohungen funktionierte die Alert-Mechanik einwandfrei, was die Fähigkeit von Wazuh unterstrich, sicherheitskritische Ereignisse sofort zu melden. Besonders hilfreich war die Integration der Schwachstellenüberwachung in die Alerts, die es ermöglichte, kritische Schwachstellen umgehend zu identifizieren und entsprechend zu priorisieren.

Die Schwachstellen wurden nach Schweregrad klassifiziert und in den Dashboards visualisiert. Dies ermöglichte eine klare Priorisierung der Sicherheitslücken auf Grundlage des CVE-Scores. Besonders wertvoll war die farbliche Hervorhebung der kritischen Schwachstellen, die sofort ins Auge fielen und eine schnelle Reaktion ermöglichten. Die Darstellung und Kategorisierung nach „Critical“, „High“, „Medium“ und „Low“ war dabei äusserst hilfreich, um eine fundierte Entscheidungsfindung zu unterstützen.

Schwachstellenmonitoring und Datenflut
Die Schwachstellenüberwachung lieferte eine grosse Menge an sicherheitsrelevanten Daten. Diese Fülle an Informationen machte die Filterung und Priorisierung der Schwachstellen zu einer Herausforderung. Die Kategorisierung nach CVE-Scores erleichterte zwar die Sichtung der wichtigsten Schwachstellen, jedoch war es schwierig, zwischen Schwachstellen, die dem Internet ausgesetzt waren, und solchen, die nur intern relevant sind, zu unterscheiden. Diese Filterung ist von entscheidender Bedeutung, da vor allem öffentlich zugängliche Schwachstellen ein grösseres Risiko darstellen.

Dennoch zeigte die Schwachstellenüberwachung eine Vielzahl von Sicherheitslücken auf, die bislang unentdeckt geblieben wären. Dies demonstriert die Effektivität des Systems bei der proaktiven Identifikation potenzieller Sicherheitsrisiken, auch wenn diese nicht immer sofort kritisch erscheinen. Langfristig bietet die Schwachstellenüberwachung eine wertvolle Basis für die kontinuierliche Verbesserung der Sicherheit der überwachten Systeme.

6.4.2.4 Fazit zur Effektivität

Die Effektivität von Wazuh konnte trotz der Herausforderungen durch gehärtete Systeme und den zusätzlichen Schutz durch Cloudflare unter Beweis gestellt werden. Insbesondere die Entdeckung der offenen SSH-Konfiguration durch die Visualisierung der fehlgeschlagenen Login-Versuche zeigte, wie das System auch in scheinbar sicheren Umgebungen zur Identifikation von Schwachstellen beitragen kann.

Die Fähigkeit, Schwachstellen zu kategorisieren, sicherheitsrelevante Ereignisse zu erfassen und Alerts in Echtzeit zu generieren, macht Wazuh zu einem effektiven Werkzeug für die Bedrohungserkennung. Die Schwachstellenüberwachung stellte sich als ein besonders leistungsfähiges Instrument heraus, um potenzielle Risiken zu erkennen und Massnahmen zur Behebung zu priorisieren. Trotz der Schwierigkeiten bei der Unterscheidung zwischen internen und externen Schwachstellen bleibt Wazuh ein wertvolles Werkzeug zur proaktiven Sicherheitsüberwachung und Risikominimierung.

6.4.3 Benutzerfreundlichkeit

Die Benutzerfreundlichkeit eines SIEM-Systems ist von entscheidender Bedeutung, um die Effizienz und Effektivität der täglichen Arbeit von Administratoren zu gewährleisten. Im Fall von Wazuh zeigt sich, dass das System in Bezug auf die Leistungsfähigkeit und Flexibilität beeindruckt, jedoch im Bereich der Benutzerfreundlichkeit Verbesserungsbedarf besteht.

6.4.3.1 Leistungsfähigkeit und Flexibilität von Wazuh

Wazuh bietet eine Vielzahl von Funktionen, die es zu einem äusserst mächtigen Werkzeug für die Sicherheitsüberwachung machen. Mit der richtigen Konfiguration und ausreichend Recherche lassen sich nahezu alle Aspekte der Überwachung und Bedrohungserkennung individuell anpassen. Die Plattform erlaubt eine flexible Gestaltung von Regeln, Alerts und Dashboards, wodurch spezifische Sicherheitsanforderungen erfüllt werden können. Besonders hervorzuheben ist, dass Wazuh durch die Integration verschiedener Sicherheitsfunktionen eine umfassende Überwachung ermöglicht, die sowohl die Schwachstellenanalyse als auch die Bedrohungserkennung in Echtzeit umfasst.

Die Möglichkeit, durch Experimentieren und Einarbeitung in die umfangreiche Dokumentation nahezu jede denkbare Überwachungsaufgabe zu realisieren, macht Wazuh zu einem vielseitigen Instrument, das in unterschiedlichen IT-Umgebungen eingesetzt werden kann. Dies eröffnet IT-Administratoren einen grossen Handlungsspielraum, um massgeschneiderte Sicherheitslösungen zu entwickeln. Wazuh bietet zudem eine hohe Anpassungsfähigkeit, was insbesondere in komplexen und dynamischen IT-Landschaften von Vorteil ist.

6.4.3.2 Herausforderungen in der Benutzerfreundlichkeit

Trotz der hohen Flexibilität weist die Benutzeroberfläche von Wazuh deutliche Schwächen auf, die die Benutzerfreundlichkeit einschränken. Besonders auffällig ist, dass die Oberfläche nicht immer intuitiv aufgebaut ist und Administratoren häufig auf die externe Dokumentation angewiesen sind, um bestimmte Konfigurationen vorzunehmen. Viele Funktionen und Optionen sind nicht auf den ersten Blick verständlich, was eine Herausforderung ist, insbesondere für weniger erfahrene Benutzer.

Ein konkretes Beispiel hierfür ist die Erstellung von benutzerdefinierten Dashboards. Obwohl Wazuh eine grosse Bandbreite an Visualisierungsmöglichkeiten bietet, ist die Implementierung spezifischer Filter oder die Anpassung der Dashboards an individuelle Anforderungen nicht immer unmittelbar ersichtlich. Die Konfiguration von Dashboards zur Darstellung von Schwachstellen oder Login-Versuchen erfordert oftmals ein tiefes Verständnis der Funktionsweise von Wazuh sowie Zeit für Experimentieren und Anpassung. Diese Komplexität kann die Benutzerfreundlichkeit beeinträchtigen, da Benutzer, die keine tiefgehenden Kenntnisse über das System haben, Schwierigkeiten haben könnten, schnell zu den gewünschten Ergebnissen zu gelangen.

6.4.3.3 Erstellung von Dashboards und Regeln

Die Erstellung und Anpassung von Dashboards und Sicherheitsregeln ist eine der zentralen Stärken von Wazuh, gleichzeitig aber auch ein Bereich, der die Benutzerfreundlichkeit herausfordert. Die Dashboards bieten umfassende Visualisierungsmöglichkeiten, die jedoch ohne ausreichende Einarbeitung nicht immer leicht zugänglich sind. Beispielsweise war es in meinem Fall notwendig, mit verschiedenen Filtern zu experimentieren, um Schwachstellen nach ihrer Kritikalität zu sortieren und darzustellen. Obwohl Wazuh die technischen Möglichkeiten für diese Visualisierungen bietet, ist die Benutzeroberfläche nicht immer intuitiv genug, um diese Konfigurationen schnell und effizient vorzunehmen.

Auch die Erstellung benutzerdefinierter Sicherheitsregeln erwies sich als leistungsfähig, aber anspruchsvoll in der Anwendung. Administratoren können spezifische Regeln entwickeln, um bestimmte sicherheitsrelevante Ereignisse zu erkennen und darauf zu reagieren. Die Konfiguration dieser Regeln setzt jedoch voraus, dass der Benutzer ein detailliertes Verständnis der zugrunde liegenden Logik und Syntax besitzt. Für weniger erfahrene Benutzer oder solche, die sich nicht intensiv mit der Wazuh-Dokumentation auseinandersetzen, kann dies eine erhebliche Herausforderung darstellen.

6.4.3.4 Einarbeitung notwendig und Zeitaufwand

Ein wesentlicher Aspekt, der die Benutzerfreundlichkeit von Wazuh beeinträchtigt, ist die Einarbeitung, die erforderlich ist, um das volle Potenzial des Systems auszuschöpfen. Während erfahrene Administratoren in der Lage sind, mit den zahlreichen Konfigurationsmöglichkeiten umzugehen, könnte dies für weniger versierte Benutzer zu Schwierigkeiten führen. Das System bietet eine grosse Vielfalt an Optionen, die jedoch ohne gründliche Einarbeitung und Experimentieren schwer zu durchschauen sind.

Ein weiterer Faktor, der die Benutzerfreundlichkeit beeinträchtigt, ist der Zeitaufwand, der für die Konfiguration und Anpassung von Wazuh erforderlich ist. Da viele Funktionen nicht sofort zugänglich oder intuitiv zu bedienen sind, müssen Administratoren oft erheblich Zeit investieren, um die gewünschte Konfiguration zu erreichen. Dies kann insbesondere in Umgebungen, in denen schnelle Anpassungen erforderlich sind, die Effizienz des Systems beeinträchtigen.

6.4.3.5 Dokumentation als zentrale Unterstützung

Ein positiver Aspekt der Benutzerfreundlichkeit von Wazuh ist die umfassende Dokumentation, die eine wertvolle Ressource für Administratoren darstellt. Die Dokumentation bietet detaillierte Anleitungen zu den verschiedenen Funktionen und Konfigurationen von Wazuh und ermöglicht es Benutzern, auch komplexe Aufgaben zu bewältigen. Allerdings ersetzt die Dokumentation nicht die Notwendigkeit einer benutzerfreundlicheren Oberfläche, die es ermöglichen würde, bestimmte Konfigurationen ohne externe Hilfe direkt in der Benutzeroberfläche vorzunehmen.

Die Abhängigkeit von der Dokumentation zeigt, dass Wazuh zwar leistungsfähig, aber nicht unbedingt zugänglich ist. Während erfahrene Benutzer die Flexibilität und die Anpassungsmöglichkeiten des Systems schätzen werden, könnten weniger erfahrene Benutzer Schwierigkeiten haben, das System ohne umfangreiche Einarbeitung zu nutzen.

6.4.3.6 Verbesserungspotenzial

Es gibt mehrere Ansätze, wie die Benutzerfreundlichkeit von Wazuh verbessert werden könnte. Eine stärkere Integration von Hilfsmitteln direkt in die Benutzeroberfläche, wie etwa kontextbezogene Hilfen oder interaktive Anleitungen, könnte die Zugänglichkeit des Systems erheblich steigern. Insbesondere Funktionen wie die Erstellung von Dashboards oder die Konfiguration von Sicherheitsregeln könnten durch vordefinierte Templates oder kontextsensitive Unterstützung vereinfacht werden.

Auch die Einführung von interaktiven Tutorials oder Assistenten, die Benutzer durch den Konfigurationsprozess führen, wäre eine sinnvolle Ergänzung. Dies würde die Einarbeitung deutlich vereinfachen und die Einarbeitungszeit reduzieren, was insbesondere in sicherheitskritischen Umgebungen von Vorteil wäre, in denen schnelle Reaktionszeiten gefordert sind.

6.4.3.7 Fazit zur Benutzerfreundlichkeit

Wazuh ist ein äusserst leistungsfähiges und flexibles Werkzeug zur Sicherheitsüberwachung, das jedoch in Bezug auf die Benutzerfreundlichkeit noch Verbesserungsbedarf aufweist. Die Flexibilität und die Vielzahl an Konfigurationsmöglichkeiten machen das System besonders attraktiv für erfahrene Administratoren, die bereit sind, Zeit in die Einarbeitung und Anpassung des Systems zu investieren. Allerdings erfordert die komplexe Benutzeroberfläche von Wazuh, dass Administratoren sich intensiv mit der Dokumentation auseinandersetzen, um das volle Potenzial des Systems nutzen zu können.

Während die Dokumentation wertvolle Unterstützung bietet, wäre eine benutzerfreundlichere Oberfläche wünschenswert, die es ermöglicht, bestimmte Konfigurationen intuitiver vorzunehmen. Mit der Einführung von interaktiven Hilfsmitteln oder kontextbezogenen Anleitungen könnte die Benutzerfreundlichkeit erheblich gesteigert werden, was Wazuh zu einer noch effektiveren und zugänglicheren Lösung für die Sicherheitsüberwachung machen würde.

6.5 Wirtschaftlichkeit und Nutzen

Die Wirtschaftlichkeit und der Nutzen eines SIEM-Systems sind entscheidende Faktoren bei der Bewertung der langfristigen Implementierung und des Einsatzes in einem Unternehmen. Wazuh, eine Open-Source-Sicherheitsplattform, bietet im Vergleich zu proprietären SIEM-Lösungen signifikante Kostenvorteile, während es gleichzeitig ein breites Spektrum an Funktionalitäten für die Bedrohungserkennung, Schwachstellenanalyse und Sicherheitsüberwachung bereitstellt. In diesem Abschnitt wird die wirtschaftliche Effizienz von Wazuh analysiert und dessen Nutzen in Bezug auf Sicherheitsvorteile und Kosteneinsparungen untersucht.

6.5.1 Kostenvorteile durch Open-Source-Natur

Ein zentraler Aspekt der Wirtschaftlichkeit von Wazuh ist dessen Open-Source-Natur, was bedeutet, dass keine Lizenzgebühren anfallen. Dies stellt einen erheblichen Vorteil gegenüber proprietären SIEM-Systemen dar, die oft mit hohen Lizenzkosten verbunden sind. In vielen Fällen steigen diese Kosten proportional zur Anzahl der überwachten Endpunkte, wodurch Unternehmen mit einer wachsenden IT-Infrastruktur signifikante Ausgaben tragen müssen. Durch den Wegfall der Lizenzgebühren ermöglicht Wazuh eine signifikante Reduktion der Gesamtbetriebskosten und schafft so eine attraktive Alternative für Unternehmen, die ihre Sicherheitsinfrastruktur kosteneffizient gestalten möchten.

Im vorliegenden Fall wurde Wazuh auf einem Hetzner CX42-Root-Server implementiert, der zu einem Preis von 15,90 Euro pro Monat bereitgestellt wird. Diese Infrastruktur stellt eine kostengünstige und leistungsfähige Basis für den Betrieb der Wazuh-Plattform sowie der zugehörigen Komponenten wie Elasticsearch und Kibana dar. Da Wazuh auf gängigen Serverplattformen betrieben werden kann, entstehen keine zusätzlichen Investitionskosten für spezielle Hardware, was einen weiteren Kostenvorteil im Vergleich zu proprietären Lösungen darstellt, die oft spezialisierte Hardware erfordern.

6.5.2 Initiale Investitionskosten

Die anfänglichen Investitionskosten für die Implementierung von Wazuh umfassen im Wesentlichen die Infrastrukturkosten sowie den zeitlichen Aufwand für die Installation und Konfiguration. Der Einsatz eines kostengünstigen, aber leistungsfähigen Servers stellte sicher, dass das System effizient arbeiten konnte, ohne unnötige Ressourcen zu beanspruchen. Da Wazuh selbst keine Lizenzgebühren verursacht, waren die einzigen initialen Kosten diejenigen, die mit der Hardware und der Arbeitszeit zur Einrichtung verbunden waren.

Die Zeit für die Konfiguration und Anpassung des Systems war aufgrund der Flexibilität und der umfangreichen Möglichkeiten von Wazuh vergleichsweise hoch. Dies betrifft insbesondere die Erstellung benutzerdefinierter Regeln, die Konfiguration von Dashboards und die Integration der Schwachstellenüberwachung. Trotz des anfänglichen Konfigurationsaufwands sind die laufenden Betriebskosten nach der Einrichtung minimal, da die Überwachungs- und Analyseprozesse weitgehend automatisiert ablaufen.

6.5.3 Langfristige Kosteneffizienz

Wazuh bietet durch den Verzicht auf Lizenzgebühren und die Flexibilität in der Infrastruktur eine langfristige Kosteneffizienz. Die einzigen wiederkehrenden Kosten sind die laufenden Serverkosten sowie der notwendige personelle Aufwand für die Systemadministration. Im Vergleich zu proprietären SIEM-Lösungen, die häufig auf nutzungsabhängigen Preismodellen basieren, ist Wazuh somit besonders attraktiv, da die fortlaufenden Kosten relativ stabil und kalkulierbar sind.

Neben den direkten Kosteneinsparungen durch den Verzicht auf Lizenzgebühren trägt Wazuh durch seine umfassenden Funktionen zur Bedrohungserkennung und Schwachstellenüberwachung dazu bei, potenzielle Sicherheitsvorfälle frühzeitig zu identifizieren. Dies hilft, die Risiken von finanziellen Verlusten durch Sicherheitsverletzungen, wie Datenverlust, Systemausfälle oder Cyberangriffe, zu minimieren. Die Fähigkeit von Wazuh, Sicherheitsrisiken präventiv zu erkennen, trägt somit nicht nur zur Reduktion der Betriebskosten bei, sondern bietet auch einen wichtigen Beitrag zur Sicherung der Geschäftsabläufe.

6.5.4 Nutzen durch Bedrohungserkennung

Wazuh bietet neben der wirtschaftlichen Effizienz auch einen erheblichen praktischen Nutzen durch seine Funktionen zur Echtzeit-Bedrohungserkennung und Schwachstellenüberwachung. Die Möglichkeit, sicherheitsrelevante Ereignisse in Echtzeit zu überwachen und sofort auf potenzielle Bedrohungen zu reagieren, ist ein zentraler Vorteil der Plattform. Besonders die Konfiguration von Alerts und die Überwachung von Schwachstellen ermöglichen eine schnelle Erkennung von Sicherheitsvorfällen und eine direkte Reaktion darauf. Ein praktisches Beispiel war die Erkennung von fehlgeschlagenen Login-Versuchen auf dem Wazuh-Server, was zur Korrektur einer fehlerhaften Firewall-Regel führte. Diese proaktive Bedrohungserkennung trug zur Reduzierung der potenziellen Angriffsfläche bei.

Die Schwachstellenüberwachung in Wazuh bietet eine umfassende Analyse potenzieller Schwachstellen in den überwachten Systemen und kategorisiert diese nach dem CVE-Score. Durch diese Funktion können Unternehmen Sicherheitslücken identifizieren, priorisieren und gezielt beheben, bevor sie von potenziellen Angreifern ausgenutzt

werden. Die Möglichkeit, Schwachstellen auf Basis ihrer Kritikalität zu priorisieren, ermöglicht eine zielgerichtete Allokation von Sicherheitsressourcen und trägt zur Minimierung des Risikos von Sicherheitsverletzungen bei.

6.5.5 Produktivitätssteigerung und Entlastung des IT-Teams

Ein zusätzlicher Nutzen von Wazuh liegt in der Automatisierung vieler Sicherheitsaufgaben, die das IT-Team entlastet und gleichzeitig die Effizienz steigert. Durch die automatische Log-Überwachung und Alert-Generierung werden Administratoren nur bei sicherheitskritischen Ereignissen benachrichtigt, was den Aufwand für die manuelle Analyse und Prüfung von Logs erheblich reduziert. Dies erlaubt es den IT-Verantwortlichen, sich auf relevante Sicherheitsvorfälle zu konzentrieren und gleichzeitig die tägliche Arbeitslast zu verringern.

Darüber hinaus bieten die in Wazuh integrierten Dashboards eine effektive Möglichkeit, sicherheitsrelevante Daten in Echtzeit zu visualisieren. Dies unterstützt das IT-Team dabei, fundierte Entscheidungen zu treffen und sofortige Massnahmen zur Gefahrenabwehr einzuleiten. Die Möglichkeit, benutzerdefinierte Dashboards zu erstellen, die den spezifischen Sicherheitsanforderungen eines Unternehmens entsprechen, erhöht die Effizienz und den praktischen Nutzen des Systems weiter.

6.5.6 Risikominimierung durch präventive Sicherheitsüberwachung

Ein weiterer wesentlicher Nutzen von Wazuh ist die Fähigkeit, potenzielle Sicherheitsbedrohungen proaktiv zu identifizieren und Schwachstellen zu beheben, bevor sie ausgenutzt werden können. Durch die kontinuierliche Überwachung sicherheitskritischer Systeme und die Integration von Threat Intelligence Feeds bietet Wazuh eine umfassende Bedrohungserkennung und die Möglichkeit, auf neue und unbekannte Bedrohungen zeitnah zu reagieren. Dies trägt entscheidend zur Minimierung des Sicherheitsrisikos bei, da potenzielle Schwachstellen schnell erkannt und behoben werden können, bevor sie zu einem ernsthaften Problem werden.

Diese proaktive Überwachung ermöglicht es Unternehmen, potenziell kostspielige Sicherheitsvorfälle zu verhindern und somit ihre finanzielle Sicherheit und geschäftliche Kontinuität zu gewährleisten. Durch die frühzeitige Erkennung und Behebung von Schwachstellen trägt Wazuh dazu bei, das Risiko von Cyberangriffen, Datenverlusten oder Systemausfällen deutlich zu verringern.

6.5.7 Fazit zur Wirtschaftlichkeit und Nutzen

Insgesamt zeigt sich, dass Wazuh ein äusserst wirtschaftlich effizientes und gleichzeitig nützliches Werkzeug zur IT-Sicherheitsüberwachung darstellt. Durch den Verzicht auf Lizenzgebühren und die flexible Implementierung auf kostengünstiger Infrastruktur ist Wazuh eine attraktive Lösung für Unternehmen jeder Grösse, die ihre Sicherheitsinfrastruktur kosteneffizient gestalten möchten. Die langfristigen Kosteneinsparungen durch die Open-Source-Natur der Plattform und die Möglichkeit, Sicherheitsvorfälle präventiv zu verhindern, bieten einen signifikanten wirtschaftlichen Vorteil.

Darüber hinaus liefert Wazuh durch seine umfassenden Funktionen zur Bedrohungserkennung, Schwachstellenanalyse und Automatisierung einen erheblichen praktischen Nutzen. Die Plattform verbessert die Effizienz der Sicherheitsüberwachung,

entlastet das IT-Team und trägt zur Risikominimierung bei. Zusammenfassend lässt sich sagen, dass Wazuh nicht nur wirtschaftlich sinnvoll ist, sondern auch eine leistungsstarke und flexible Lösung zur Sicherung von IT-Infrastrukturen darstellt.

6.6 Schlussfolgerungen und Ausblick

In diesem abschliessenden Kapitel werden die Ergebnisse der Implementierung und Evaluation des Wazuh-Systems zusammengefasst. Die Schlussfolgerungen basieren auf den zuvor durchgeführten Analysen in den Bereichen Bedrohungserkennung, Schwachstellenüberwachung, Benutzerfreundlichkeit und Wirtschaftlichkeit. Neben der Bewertung des aktuellen Zustands der Implementierung wird ein Ausblick auf mögliche Verbesserungen und zukünftige Entwicklungen gegeben, die das Potenzial von Wazuh in der langfristigen Nutzung weiter steigern könnten.

Ziel dieses Kapitels ist es, die wichtigsten Erkenntnisse zusammenzufassen und eine Perspektive zu bieten, wie Wazuh als SIEM-Lösung in der Praxis weiter optimiert und genutzt werden kann. Dabei wird insbesondere auf die Bereiche eingegangen, in denen das System bereits sehr gut funktioniert, sowie auf die Herausforderungen, die für eine effektive Nutzung noch gelöst werden müssen. Der Ausblick wird zudem potenzielle Erweiterungen und zukünftige Entwicklungen der Plattform beleuchten, die ihre Einsatzmöglichkeiten in einem dynamischen IT-Umfeld weiter stärken könnten.

6.6.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde die Implementierung und Evaluation der Open-Source-Sicherheitsplattform Wazuh durchgeführt, mit dem Ziel, ihre Eignung als SIEM-Lösung in einer produktiven Umgebung zu bewerten. Die Arbeit begann mit der Definition der Anforderungen an das System, gefolgt von der technischen Implementierung, der Konfiguration von Regeln, Alerts und Dashboards, sowie einer umfassenden Evaluation der Plattform in Bezug auf Effektivität, Benutzerfreundlichkeit und Wirtschaftlichkeit.

6.6.1.1 Anforderungen

Zu den grundlegenden Anforderungen an das Wazuh-System gehörten die Echtzeitüberwachung sicherheitsrelevanter Ereignisse, die Schwachstellenüberwachung und die Fähigkeit, auf potenzielle Bedrohungen durch Alerts in Echtzeit zu reagieren. Weiterhin wurde eine intuitive Benutzeroberfläche gefordert, die es ermöglicht, Regeln und Dashboards ohne tiefgehende technische Kenntnisse zu konfigurieren. Neben der Funktionalität war auch die Wirtschaftlichkeit ein entscheidender Faktor, da eine kostenfreie Open-Source-Lösung evaluiert wurde, die dennoch einen hohen praktischen Nutzen bieten sollte.

6.6.1.2 Implementierung und Effektivität

Die Implementierung des Wazuh-Systems erfolgte auf einem Hetzner CX42-Root-Server, der kostengünstig und leistungsfähig genug war, um die notwendige Infrastruktur für das System bereitzustellen. Dabei wurden der Wazuh-Manager, Elasticsearch und Kibana erfolgreich installiert und konfiguriert. Ein zentrales Element der Implementierung war die Konfiguration von Regeln zur Bedrohungserkennung und Alerts, die über Webhooks an einen Teams-Kanal gesendet wurden, um sicherheitsrelevante Vorfälle in Echtzeit zu melden.

Ein besonders wichtiges Beispiel für die Effektivität des Systems zeigte sich in der Entdeckung einer Fehlkonfiguration des Ports 22 auf der Wazuh-Instanz. Durch die Visualisierung von fehlgeschlagenen Login-Versuchen in einer Heatmap wurden sicherheitskritische Ereignisse sichtbar, die darauf hinwiesen, dass der Zugriff auf den Port fälschlicherweise nicht eingeschränkt war und somit allen IP-Adressen offenstand. Dieser Konfigurationsfehler wurde durch die Funktionalität von Wazuh erkannt und konnte somit sofort behobenwerdn, was zu einer signifikanten Verringerung der Angriffsfläche führte. Dieses Beispiel verdeutlichte die Fähigkeit von Wazuh, Bedrohungen proaktiv zu erkennen und zum Schutz der IT-Systeme beizutragen.

6.6.1.3 Schwachstellenüberwachung und CVE-Scores

Ein weiterer zentraler Aspekt der Implementierung war die Aktivierung der Schwachstellenüberwachung. Diese Funktionalität generierte eine grosse Menge an sicherheitsrelevanten Daten, indem sie bekannte Schwachstellen auf den überwachten Servern identifizierte und nach ihrem CVE-Score klassifizierte. Das Dashboard zur Schwachstellenüberwachung ermöglichte es, Schwachstellen nach Kritikalität (Critical, High, Medium, Low) zu filtern und darzustellen. Besonders hilfreich war die Visualisierung der Critical Vulnerabilities, die sofort ins Auge fielen und Administratoren eine Priorisierung der Behebung von Sicherheitslücken ermöglichten.

Eine der grössten Herausforderungen bei der Nutzung dieser Funktion war jedoch die Filterung der Schwachstellen, um jene zu identifizieren, die extern (über das Internet) zugänglich sind. Obwohl das System eine klare Übersicht über alle erkannten Schwachstellen bietet, war es schwierig, zwischen internen und externen Sicherheitslücken zu unterscheiden. Diese Filtermöglichkeit wäre jedoch von grosser Bedeutung, da externe Schwachstellen weitaus kritischer sind und dringender behoben werden müssen.

6.6.1.4 Benutzerfreundlichkeit

Die Benutzerfreundlichkeit des Wazuh-Systems erwies sich als zweischneidiges Schwert. Auf der einen Seite bietet das System eine beeindruckende Flexibilität und Leistungsfähigkeit, die es ermöglicht, nahezu jede Überwachungsaufgabe zu realisieren, wenn genügend Recherche und Experimentieren investiert werden. Auf der anderen Seite ist die Benutzeroberfläche jedoch nicht immer intuitiv gestaltet, was die Konfiguration von Regeln und Dashboards für weniger erfahrene Administratoren erschwert. Viele der Funktionen erfordern ein tiefes Verständnis der zugrunde liegenden Systeme und häufig die Konsultation der Wazuh-Dokumentation, um die gewünschten Konfigurationen erfolgreich umzusetzen.

Während die Dokumentation umfassend und hilfreich ist, wäre eine stärkere Integration von Hilfsmitteln direkt in der Benutzeroberfläche wünschenswert, um die

Einarbeitungszeit zu verkürzen und die Bedienbarkeit zu verbessern. Die Erstellung von interaktiven Tutorials oder kontextbezogenen Anleitungen könnte die Benutzerfreundlichkeit erheblich steigern.

6.6.1.5 Wirtschaftlichkeit und Nutzen

Die wirtschaftliche Analyse von Wazuh zeigte, dass die Plattform aufgrund ihrer Open-Source-Natur signifikante Kostenvorteile bietet. Da keine Lizenzgebühren anfallen, entstehen lediglich Kosten für die Infrastruktur und die Administration. Die Implementierung auf einem kostengünstigen Hetzner-Server ermöglichte den Betrieb eines voll funktionsfähigen SIEM-Systems bei minimalen monatlichen Betriebskosten. Im Vergleich zu proprietären SIEM-Lösungen, die oft hohe Lizenzgebühren verlangen, bietet Wazuh somit eine kosteneffiziente Alternative, die gleichzeitig umfassende Überwachungs- und Bedrohungserkennungsfunktionen bereitstellt.

Darüber hinaus bietet Wazuh durch seine Schwachstellenüberwachung und Automatisierung einen hohen praktischen Nutzen. Die Echtzeit-Bedrohungserkennung, die Möglichkeit zur Priorisierung von Schwachstellen sowie die Entlastung des IT-Teams durch automatisierte Alerts tragen massgeblich zur Risikominimierung und zur Effizienzsteigerung bei. Langfristig bietet Wazuh somit nicht nur wirtschaftliche Vorteile, sondern auch einen signifikanten Beitrag zur Sicherheitsstrategie eines Unternehmens.

6.6.1.6 Fazit zum Schlusswort

Zusammenfassend lässt sich sagen, dass Wazuh eine leistungsfähige und flexible Open-Source-Lösung für die IT-Sicherheitsüberwachung darstellt, die sowohl aus wirtschaftlicher Sicht als auch in Bezug auf den praktischen Nutzen überzeugt. Trotz einiger Herausforderungen in der Benutzerfreundlichkeit zeigt das System eine hohe Effektivität bei der Bedrohungserkennung und Schwachstellenüberwachung. Die Implementierung und Anpassung des Systems erfordert zwar eine gewisse Einarbeitung, jedoch bietet Wazuh mit seinen umfassenden Funktionen eine kosteneffiziente und vielseitige Lösung zur Sicherung von IT-Infrastrukturen.

6.6.2 Weiterentwicklung

Die Implementierung und der Einsatz von Wazuh haben gezeigt, dass die Plattform nicht nur ein leistungsfähiges Werkzeug für die IT-Sicherheitsüberwachung ist, sondern auch Potenzial für eine kontinuierliche Weiterentwicklung bietet. Um die langfristige Wirksamkeit und Sicherheit der Plattform sicherzustellen, ist es entscheidend, die implementierten Funktionen stetig zu erweitern und zu optimieren.

6.6.2.1 Sicherstellung des sicheren Einsatzes

Ein zentraler Aspekt der Weiterentwicklung von Wazuh liegt in der fortlaufenden Sicherstellung des sicheren Einsatzes der Plattform. Das System bietet bereits eine starke Basis für die Erkennung von Bedrohungen und die Überwachung von Schwachstellen, doch die kontinuierliche Anpassung an neue Bedrohungen und die regelmässige Aktualisierung der Sicherheitsregeln sind unerlässlich, um das volle Potenzial der Plattform zu nutzen. Dies erfordert eine regelmässige Wartung und Anpassung des Systems, um sicherzustellen, dass neue Sicherheitsrisiken, Schwachstellen oder Konfigurationsfehler frühzeitig erkannt und adressiert werden können.

Besonders in dynamischen IT-Umgebungen, in denen sich Anforderungen und Sicherheitsbedrohungen schnell ändern, ist es notwendig, das System ständig weiterzuentwickeln. Regelmässige Systemupdates, das Monitoring neuer Threat Intelligence Feeds und die Anpassung der Sicherheitsregeln sollten Teil einer kontinuierlichen Strategie sein, um sicherzustellen, dass Wazuh auch langfristig einen hohen Schutz vor Cyberangriffen bietet.

6.6.2.2 Investitionen in den Ausbau der Funktionalität

Die bisherige Implementierung von Wazuh hat gezeigt, dass das System in der Lage ist, eine Vielzahl von sicherheitsrelevanten Daten zu sammeln und zu visualisieren. Um diese Fähigkeiten weiter auszubauen, sollten zusätzliche Funktionen und Erweiterungen in Betracht gezogen werden. Eine vielversprechende Weiterentwicklung wäre die Implementierung von Konfigurationsüberprüfungen, um mögliche Fehlkonfigurationen systematisch zu identifizieren. So wie das Beispiel der offenen Port 22 Konfiguration auf dem Root-Server gezeigt hat, können solche Fehlkonfigurationen erhebliche Risiken darstellen. Durch die Implementierung von Regeln, die abweichende Konfigurationen erkennen und melden, könnten potenzielle Angriffsflächen weiter reduziert werden.

Ein weiteres Element der Weiterentwicklung könnte die verstärkte Automatisierung bei der Behebung von Schwachstellen sein. Wazuh bietet bereits eine umfassende Schwachstellenüberwachung, jedoch könnten in Zukunft weitere Mechanismen entwickelt werden, die automatisch Handlungsempfehlungen ausgeben oder sogar bestimmte Standardmassnahmen zur Fehlerbehebung vorschlagen. Dies würde den Arbeitsaufwand für Administratoren weiter reduzieren und die Reaktionszeit bei sicherheitskritischen Vorfällen verkürzen.

6.6.2.3 Entdecken weiterer Sicherheitsfunktionen

Neben der Überwachung von Schwachstellen und sicherheitsrelevanten Ereignissen könnte Wazuh durch die Integration zusätzlicher Sicherheitsfunktionen weiter gestärkt werden. Eine Möglichkeit wäre die Erweiterung des Konfigurationsmanagements, um Compliance-Prüfungen für Sicherheitsstandards wie ISO 27001, NIST oder CIS Controls automatisiert durchzuführen. Dies würde Unternehmen dabei unterstützen, ihre Systeme nicht nur sicher, sondern auch konform mit internationalen Sicherheitsstandards zu halten.

Auch die Netzwerküberwachung könnte intensiviert werden, indem zusätzliche Module zur Analyse von Netzwerkaktivitäten und zur Erkennung von anomalen Netzwerkverbindungen integriert werden. Die Identifizierung von ungewöhnlichem Netzwerkverkehr könnte eine noch tiefere Analyse potenzieller Bedrohungen ermöglichen und die Sicherheitsarchitektur auf Netzwerkebene weiter verbessern.

6.6.2.4 Lehren aus gehärteten Systemen und zukünftige Einsichten

Die Implementierung von Wazuh in einer Umgebung mit gehärteten Systemen hat zu wertvollen Erkenntnissen geführt, obwohl sie auch einige Herausforderungen mit sich brachte. Durch die Härtung der Systeme und den Schutz durch Cloudflare traten weniger sicherheitskritische Vorfälle auf, was die Datengrundlage für die Bedrohungserkennung reduzierte. Dennoch haben die eingesetzten Dashboards und die Schwachstellenüberwachung wertvolle Einblicke in die Sicherheitsarchitektur geliefert, die sonst nicht sichtbar gewesen wären.

Rückblickend haben die gesammelten Daten und Erkenntnisse dazu beigetragen, die Sicherheitslage der Systeme besser zu verstehen und weiter zu optimieren. Die Entdeckung der Port 22 Konfiguration und die Analyse von Schwachstellen anhand der CVE-Scores haben gezeigt, wie wichtig es ist, Systeme kontinuierlich zu überwachen, auch wenn sie bereits gehärtet sind. Diese Einsichten bilden die Grundlage für zukünftige Weiterentwicklungen und Sicherheitsoptimierungen, die darauf abzielen, die bereits bestehenden Systeme noch widerstandsfähiger gegen Bedrohungen zu machen.

6.6.2.5 Fazit zur Weiterentwicklung

Die bisherigen Implementierungserfahrungen mit Wazuh verdeutlichen, dass die Plattform eine solide Basis für eine umfassende Sicherheitsüberwachung bietet, aber gleichzeitig noch Raum für Weiterentwicklungen vorhanden ist. Durch den Ausbau der Funktionalität, insbesondere im Bereich der Konfigurationsüberprüfung und der verstärkten Automatisierung von Sicherheitsmassnahmen, kann Wazuh zu einem noch leistungsfähigeren Werkzeug für die IT-Sicherheit werden.

Die fortlaufende Investition in den Ausbau von Wazuh wird dazu beitragen, die Plattform zukunftssicher zu gestalten und sicherzustellen, dass sie den wachsenden Anforderungen einer sich ständig verändernden Bedrohungslandschaft gerecht wird.

7 Literaturverzeichnis

Gupta, R. (2024). *YouTube*. Von https://youtu.be/fvMus-Tc83E?si=ZC_pGZdQPHpiQ7jx abgerufen

Lempa, C. (2024). *YouTube*. Von https://youtu.be/RjvKn0Q3rgg?si=N2M3WqrvW-Ew5XWV abgerufen

NetworkChuck. (2023). *YouTube*. Von https://youtu.be/3CaG2GI1kn0?si=xl7f6YBygp3nWCg2 abgerufen

Wazuh. (2024). *Wazuh Dokumentation*. Von https://documentation.wazuh.com/current/index.html abgerufen

Abschnitte dieser Arbeit wurden mithilfe einer künstlichen Intelligenz (AI) auf Rechtschreibung, Grammatik sowie stilistische und sprachliche Klarheit überprüft und optimiert. Dies diente der Verbesserung des Leseflusses und der Ausdrucksstärke, wobei der inhaltliche Charakter und die wissenschaftliche Integrität der Arbeit gewahrt wurden.

8 Abbildungsverzeichnis

9 Fazit

Im Rahmen dieser Arbeit wurde die Open-Source-Sicherheitsplattform Wazuh umfassend implementiert, getestet und evaluiert, um ihre Eignung als SIEM-Lösung in einer produktiven Umgebung zu beurteilen. Der Fokus lag darauf, Wazuhs Fähigkeit zur Bedrohungserkennung, Schwachstellenüberwachung und Sicherheitsüberwachung zu analysieren, während die Benutzerfreundlichkeit und Wirtschaftlichkeit ebenfalls kritisch bewertet wurden.

9.1 Erfüllung der Anforderungen

Die ursprünglichen Anforderungen an das System, insbesondere die Echtzeitüberwachung von sicherheitsrelevanten Ereignissen und die Schwachstellenanalyse, wurden erfolgreich erfüllt. Durch die Implementierung von Regeln und Alerts konnte eine umfassende Sicherheitsüberwachung realisiert werden, die in der Lage ist, sicherheitskritische Vorfälle zu erkennen und zeitnah darauf zu reagieren. Besonders die Schwachstellenüberwachung, die Schwachstellen nach CVE-Scores kategorisiert, ermöglichte es, Sicherheitslücken effektiv zu priorisieren und gezielt zu beheben.

9.2 Effektivität und Praxisnutzen

Ein besonders wichtiges Beispiel für die Effektivität von Wazuh zeigte sich in der Entdeckung einer Fehlkonfiguration des SSH-Ports 22 auf dem Root-Server, die durch die Analyse von fehlgeschlagenen Login-Versuchen sichtbar wurde. Diese Sicherheitslücke konnte durch Wazuh frühzeitig entdeckt und somit behoben werden, was die potenzielle Angriffsfläche erheblich reduzierte. Dieses Beispiel verdeutlicht die Fähigkeit von Wazuh, proaktiv zur Systemsicherheit beizutragen und Bedrohungen rechtzeitig zu identifizieren.

Zudem zeigte sich, dass Wazuh trotz der Herausforderungen durch gehärtete Systeme und den Schutz durch Cloudflare wertvolle Einblicke in die Sicherheitslage der überwachten Systeme ermöglichte. Trotz des Mangels an sicherheitskritischen Ereignissen lieferten die Schwachstellenüberwachung und die benutzerdefinierten Dashboards wichtige Informationen über potenzielle Sicherheitslücken.

9.3 Benutzerfreundlichkeit

Trotz der Flexibilität und Leistungsfähigkeit von Wazuh erwies sich die Benutzeroberfläche als nicht immer intuitiv. Die Konfiguration von Regeln und Dashboards erforderte eine erhebliche Einarbeitung und Recherche, was die Lernkurve für weniger erfahrene Administratoren erschwerte. Die Dokumentation von Wazuh war zwar hilfreich, jedoch könnte die Benutzerfreundlichkeit durch die Integration von interaktiven Hilfsmitteln und kontextbezogenen Anleitungen weiter verbessert werden.

9.4 Wirtschaftlichkeit

Wazuh hat sich als eine äusserst kosteneffiziente Lösung für die Sicherheitsüberwachung erwiesen, insbesondere im Vergleich zu proprietären SIEM-Systemen. Da keine Lizenzgebühren anfallen, entstehen nur geringe laufende Kosten für die Infrastruktur und den Betrieb des Systems. Dies macht Wazuh zu einer wirtschaftlich attraktiven Lösung, die dennoch umfassende Sicherheitsfunktionen bietet.

9.5 Weiterentwicklung

Die Arbeit zeigt, dass Wazuh trotz seines bereits umfassenden Funktionsumfangs noch weiterentwickelt werden kann. Die Integration von Konfigurationsüberprüfungen, die Optimierung der Schwachstellenüberwachung und die Implementierung zusätzlicher Automatisierungsfunktionen sind potenzielle Bereiche, in denen das System weiter ausgebaut werden kann. Zudem bietet die Plattform die Möglichkeit, zusätzliche Sicherheitsfunktionen wie Compliance-Prüfungen oder Netzwerküberwachung zu integrieren.

9.6 Abschliessende Bewertung

Insgesamt hat sich Wazuh als ein äusserst leistungsfähiges, flexibles und kosteneffizientes Tool für die IT-Sicherheitsüberwachung erwiesen. Die Plattform erfüllt die Anforderungen an eine umfassende Bedrohungserkennung und Schwachstellenüberwachung und bietet durch ihre Open-Source-Natur erhebliche wirtschaftliche Vorteile. Trotz einiger Herausforderungen in der Benutzerfreundlichkeit bleibt Wazuh eine wertvolle Lösung für Unternehmen, die eine skalierbare und kostengünstige Sicherheitsplattform benötigen.

Langfristig betrachtet, bietet Wazuh nicht nur eine kostengünstige Alternative zu proprietären SIEM-Systemen, sondern auch eine Plattform, die durch ihre kontinuierliche Weiterentwicklung zukunftssicher gestaltet werden kann. Die gewonnenen Erkenntnisse und die erfolgreiche Implementierung zeigen, dass Wazuh einen wichtigen Beitrag zur IT-Sicherheitsstrategie eines Unternehmens leisten kann und dabei sowohl wirtschaftliche als auch sicherheitstechnische Vorteile vereint.

10 Schlusswort

Die Auseinandersetzung mit der Implementierung und Evaluation der Open-Source-Sicherheitsplattform Wazuh hat wertvolle Einblicke in die Anforderungen und Herausforderungen moderner IT-Sicherheitsüberwachung gegeben. Im Laufe dieser Arbeit wurde deutlich, dass Wazuh trotz seiner Komplexität und anfänglichen Lernkurve eine äusserst leistungsfähige und flexible Lösung darstellt. Die Möglichkeit, durch gezielte Konfigurationen, Regeln und Dashboards nahezu jede sicherheitsrelevante Anforderung abzudecken, zeigt das Potenzial der Plattform auf.

Besonders beeindruckend war die Entdeckung und Behebung realer Sicherheitslücken, wie etwa der Fehlkonfiguration des SSH-Ports, die durch die Visualisierung von sicherheitsrelevanten Daten ermöglicht wurde. Solche Erfolge verdeutlichen, dass auch in gut gehärteten Systemen weiterhin Raum für Verbesserungen und wertvolle Sicherheitsgewinne besteht.

Die Arbeit hat gezeigt, dass Zeit und Engagement in die Weiterentwicklung und Konfiguration von Wazuh investiert werden müssen, um das volle Potenzial der Plattform zu nutzen. Mit der kontinuierlichen Verbesserung und Anpassung an neue Bedrohungen wird Wazuh zu einem immer wichtigeren Baustein in der Sicherheitsstrategie eines Unternehmens.

Rückblickend war die Implementierung von Wazuh nicht nur eine technische Herausforderung, sondern auch eine Gelegenheit, das Sicherheitsbewusstsein weiter zu schärfen und die Notwendigkeit einer laufenden Überwachung und Anpassung moderner IT-Systeme zu betonen. Die wertvollen Erkenntnisse aus dieser Arbeit werden auch in Zukunft dazu beitragen, die Sicherheit der IT-Infrastruktur weiter zu optimieren und potenzielle Risiken zu minimieren.

11 Glossar und Abkürzungsverzeichnis

Alerts: Automatische Benachrichtigungen, die bei Erkennung sicherheitskritischer Ereignisse durch das SIEM-System ausgelöst werden.

Bedrohungserkennung: Prozess der Identifizierung potenzieller Sicherheitsvorfälle oder Angriffe in einem Netzwerk oder System.

Cloudflare: Ein Cloud-Dienst, der Websites und Netzwerke durch Content Delivery, DDoS-Schutz und Web Application Firewalls (WAF) schützt.

CVE (Common Vulnerabilities and Exposures): Ein öffentliches Verzeichnis von bekannten Sicherheitslücken, das zur Klassifizierung und Bewertung von Schwachstellen in Systemen und Software dient.

Dashboards: Grafische Oberflächen, die sicherheitsrelevante Daten und Ereignisse visualisieren und zur Überwachung und Analyse in Echtzeit verwendet werden.

Elasticsearch: Eine Such- und Analyse-Engine, die grosse Mengen an Daten speichert und schnell durchsucht. Teil des Wazuh-Ökosystems.
Firewall: Ein Netzwerk-Sicherheitsmechanismus, der den ein- und ausgehenden Datenverkehr basierend auf vordefinierten Regeln kontrolliert.

Heatmap: Eine grafische Darstellung, die Daten in einer farblichen Intensität visualisiert, um Muster oder Anomalien anzuzeigen. In der Sicherheitsüberwachung oft verwendet, um geografische Herkunft von Ereignissen zu zeigen.
Konfigurationsüberprüfung: Die Analyse und Validierung von Systemeinstellungen, um potenzielle Fehlkonfigurationen zu erkennen, die zu Sicherheitsrisiken führen könnten.

Kibana: Ein Analyse- und Visualisierungstool, das Daten aus Elasticsearch darstellt. Es bietet Dashboards zur Überwachung von Wazuh-Daten.
Logs: Aufzeichnungen von Ereignissen und Aktivitäten, die in einem System oder Netzwerk stattfinden. Diese Daten werden zur Analyse und Bedrohungserkennung genutzt.

Port 22: Standardport für den SSH-Dienst (Secure Shell), der zur sicheren Fernverwaltung von Computern verwendet wird.

Schwachstellenüberwachung: Die systematische Überprüfung von IT-Systemen auf bekannte Sicherheitslücken, die in der CVE-Datenbank verzeichnet sind.

SIEM: Systeme zur Sammlung, Überwachung und Analyse von sicherheitsrelevanten Daten in Echtzeit.

Webhook: Eine automatisierte Nachrichtenschnittstelle, die in der IT verwendet wird, um Benachrichtigungen zu senden, sobald ein bestimmtes Ereignis auftritt.

Wazuh: Eine Open-Source-Sicherheitsplattform, die Funktionen zur Bedrohungserkennung, Schwachstellenüberwachung und Sicherheitsüberwachung bietet.

API: Application Programming Interface
CVE: Common Vulnerabilities and Exposures
CX: Cloud Server Model bei Hetzner (z.B. CX42)
DDoS: Distributed Denial of Service
ISO: International Organization for Standardization
IT: Information Technology
NIST: National Institute of Standards and Technology
SIEM: Security Information and Event Management
XDR: Extended Detection and Response
SSH: Secure Shell
WAF: Web Application Firewall
PoC: Proof of Concept

12 Eigenständigkeits-Erklärung

Arbeiten, die nachweisbar in vollen Umfang oder in den wesentlichen Teilen unverändert oder ohne korrekte Quellenangabe übernommen werden, gelten als vorfabriziert und werden nicht bewertet.

Ich bestätige, dass ich die vorliegende Diplomarbeit selbstständig verfasst und alle benutzten Quellen gekennzeichnet habe. Diese Arbeit wurde weder in gleicher noch in ähnlicher Form bereits einer Prüfungskommission vorgelegt.

Name / Vorname:
Lars Dänzer

Ort / Datum / Unterschrift:
Langnau i.E., 27.10.2024