Hochschule Worms

Fachbereich Informatik

Studiengang Angewandte Informatik B.Sc.

TBD

Bachelorarbeit xxx

Bruno Macedo da Silva
676839
inf3645@hs-worms.de
Bebelstraße 22 Z10
67549 Worms

Betreuer	Prof. Dr. Zdravko Bozakov
Bearbeitungszeitraum:	Sommersemester 2023
Abgabedatum:	xx. xxx 2023
Sperrvermerk:	Ja/Nein

Inhaltsverzeichnis

Abstract

XXXXXXXXXXXXXXXXXXX

Abbildungsverzeichnis

Glossar

Abfragesprache *Query Language* funktioniert wie einen Filter fir die Suche nach spezifischen Daten in einem Datenbank (at, 2022). 18

Application Programming Interface (API) beziehen sich auf Coden und Regel, die die Kommunikation zwischen verschiedenen Anwendungen ermöglichen. In diesem Fall kann eine Anwendung eine Anfrage einer anderen Anwendung schicken, um Daten zu holen oder zu schicken(IBM, 2020). 31, 32

Brute-Force Angriffe systematischer Versuch, Credentials oder andere sensitive Daten zu raten, indem verschiedene Buchstaben, Ziefern und Symbolen kombiniert werden (Sowmya et al., 2012). 8, 23, 33

Container funktionieren ähnlich wie vm, mit dem Unterschied, dass Container ab dem Software-Ebene entstehen. Containers werden meisten für einzelne Anwendungen verwendet. Sie benutzen die Ressource der Host-Maschine, haben aber eine isolierte Umgebung mit den notwendigen Ressource für den Lauf der ausgewählten Anwendung (Douglis and Nieh, 2019). In dieser Arbeit benutzen wir Docker Container. 20, 24, 29, 31

Cyberangriff - Angriffe, die über Cyberspace stattfinden. Solche Angriffe zielen auf Unternehmen und deren Infrastrukturen, um sie zu zerstören, zu lähmen, zu kontrollieren oder die Integrität ihren Daten zu stehlen oder zu dominieren (NIST, 2020b). 1, 2, 5, 20, 21, 23

Confidentiality, Integrity and Availability (CIA) beschreiben die drei wichtigsten Schutzziele der IT-Sicherheit: Vertraulichkeit, Integrität und Verfügbarkeit (Wendzel, 2018). 7

Cyber Kill Chain®(CKC®) auch *Cyberattack Lifecycle* genannt, bezieht sich auf ein Sicherheitsmodell für die Identifizierung, Analysis und Unterbrechung von fort-

geschrittenen Cyberangriffen. Dieses Model hat sieben festgelegte Phasen: *Recoinnaissance*, *Weaponization*, *Delivery*, *Exploitation*, *Installation*, *Command & Control (C2)* und *Actions on Objectives* (Martin, 2018). 8, 21

Cybersicherheit - Diese Domäne umfasst Kenntnisse und Methoden für den Schutz, Prävention, Wiederherstellung von elektronischen Kommunikationsmittel und deren Inhalt. Es konzentriert sich auf ihrer Verfügbarkeit, Integrität, Authentizität, Vertraulichkeit und Verbindlichkeit (NIST, 2020b). 21

Endpoint bezieht sich auf Geräte oder Systeme, die mit dem Netzwerk verbunden sind. Diese können z.B. Handys, Servers, Computers, Sensoren sein (Microsoft Security, 2022). 4, 5, 16, 18, 31, 32, 34, 37

Hashwerte sind Zeichen, die aus der Ergebnis einer mathematischen Funktion (Hashfunktion) auf einem Text oder auf einer Datei entstehen. Die Wiederherstellung des originallen Objekts soll aber schwierig sein (Wendzel, 2018). 23

Hypertext Transfer Protocol (HTTP) ist der Basis unseres Internets. Dieser Protokol definiert die Regel für die Übertragung von Texten und Datein im Internet. Dieser Protokol verwendet acht Methode, um die Kommunikation zwischen Kunden und Server herzustellen: *GET, POST, HEAD, DELETE, CONNECT, OPTIONS* und *TRACE* (Chai and Ferguson, 2021) and (tutorialspoint, 2009). 31

Falsch Positiv ist eine aus einer fehlerhaften erkannten Verwundbarkeit Warnmeldung (NIST, 2020b). 13

Graphical User Interface (GUI)) ist eine Schnittstelle, die den Nutzer ermöglicht, mithilfe von Symbolen und visuellen Elementen mit der Anwendung zu interagieren (Fu, 2018). 10, 24

Hydra ist eine in Kali Native Open Source Tool für die Entdeckung von Anmeldedaten (Kali, 2022a). 25, 26

Health Insurance Portability and Accountability Act (HIPAA) ist ein US-Bundesgesetz über den Schutz von sensitiven personenbezogenen Gesundheitsdaten (U.S. Department of Health & Human Services, 2016). 7

Kali ist eine Open Source Linux-Distribution, die wegen spezifische Tools im Sicherheitsbereich für die Durchführung von Angriffe und Sicherheitstesten verwendet wird (Kali, 2022b). 24

LogQL ist eine für Grafana Loki entwickelte Abfragesprache. Sie wird verwendet, um Logdateien zu zusammenzustellen (Grafana Labs, 2021c). 34

Multi-Faktor-Authentisierung (MFA) bezeichnet das Authentifizierungsverfahren, indem zwei oder mehrere unabhängige Komponenten zur Authentifizierung verwendet werden, z.B. ein Passwort zusammen mit dem Fingerabdruck, oder eine Karte zusammen mit der Erkennung des Muster der Iris im Auge(Ibrokhimov et al., 2019). 33

Machine Learning (ML) bezieht sich auf die Fähigkeit von Systemen, automatisch Probleme zu lösen und spezifische Aufgabe zu erledigen mithilfe von Datenbeziehung und Bearbeitung (Janiesch et al., 2021). 8, 14

Mitre ATT&CK® Abkürzung für *Adversarial Tactics, Techniques and Common Knowledge*. Es bezieht sich auf eine weltweit zugängliche Wissensbasis mit detaillierter Beschreibung, Klassifizierung und Bekämpfung von verschiedenen Angriffstechniken (MITRE ATT&CK, 2018a). 1, 2, 19–22

National Institute of Standards and Technology (NIST) ist eine US-Behörde, die für die Regelungen, Vereinheitlichung und Weiterentwicklung im Bereich Informationstechnologie zuständig ist (NIST, 2020a). 1

Open Source beschreibt einen Code, an den jeder zugreifen, modifizieren und verbreiten kann, ohne dafür Lizenzen bezahlen zu müssen (Open Source Initiative, 2007). 1,

2, 4, 7, 9, 12, 15, 19, 32

Password Spraying ist ein Angriff gegen Anmeldedaten, indem mögliche Passwörter gegen verschiedenen viele Benutzername verwendet werden. Das Ziel dieses Angriffes ist eine Kontosperrung zu vermeiden, indem wenige Versuche pro Nutzer stattfindet (Swathi, 2022). 23, 24, 27, 28

Password Stuffing ist ein Angriff gegen Passwörtern, indem schon bekannte Anmeldedaten von vorherigen Angriffen verwendet werden. Dieser Angriff basiert sich auf die Idee, dass Nutzer dasselbe Passwort für verschiedenen Systemen verwenden (Ba et al., 2021). 23, 25, 26

Payment Card Industry Data Security Standard (PCDI DSS) sind Sicherheitsstandards, die Unternehmen, die Kreditkarte akzeptieren, bearbeiten, speichern oder übertragen, anwenden müssen (Centers for Disease Control and Prevention, 2016). 7

Network Operations Center (NOC) ist ein zentralisierter Bereich eines Unternehmens und dafür zuständig, Netzwerkaktivitäten zu überwachen und zu verwalten (Mohammed et al., 2021). 15

Plugin sind optionale Software-Komponenten, die weitere Funktionalitäten zu einer Anwendung hinzufügen (IT-Service.Network, 2020). 15, 17

Prometheus ist ein OpenSource Tool von der Firma SoundCloud. Dieses Tool beschäftigt sich mit Überwachung und mit Erstellung von Warnmeldung je nach Regel konfiguriert wurde (Prometheus, 2016). 34

Proprietary bezieht sich auf Software, die einer Firma oder Person gehören. Für die Nutzung ist meisten Kauf einer Lizenz notwendig. In diesem Fall haben Kunden wenig oder kaum Zugang zu einem originellen Code (Nexcess, 2022). 2, 7, 19

Rockyou ist eine Textdatei mit über 8 Milliarden Passwörter im Klartext. Diese Datei entstammt aus einem im Jahr 2009 stattgefundenen Angriff gegen Yahoo und ist

seitdem ständig aktualisiert (Mikalauskas, 2023). 25, 27

Security Operations Center (SOC) ist ein zentralisierter Bereich eines Unternehmens und dafür zuständig, Sicherheitsvorfälle zu überwachen, zu identifizieren, zu bewerten und dazu zu reagieren (Vielberth, 2021). 1, 4

Secure Shell Protocol(SSH) ist ein Netzwerkprotokoll, das eine verschlüsselte Verbindung zwischen den Endpoints anbietet. SHH wird meistens für die Fernadministration von Rechner verwendet. Dieses Protokoll ermöglicht die Erstellung eine sichere Verbindung in einer unsicheren Umgebung (Wendzel, 2018). 25

Ubuntu ist eine Open Source Linux-Distribution, die oft für Servers, Desktops und Internet of Things (IoT) verwendet wird (Ubuntu, 2023b). 24

Use Cases beschreiben die Interaktion zwischen Systemen und Benutzer. Sie dienen zu der Anforderungserhebung eines Systems (Savic et al., 2012). 2, 8, 19

virtuelle Maschine (VM) ist eine Kopie der Hardware-Struktur mit eigener Aufteilung von Ressources und mit eigenen Betriebssystem. In einer physikalischen Maschine, auch Host virtuelle Maschine genannt, kann mehreren von solchen VMs laufen. Sie emulieren ein echtes und unabhängiges System (Tanembaum, 2009). 20, 24

Abkürzungsverzeichnis

API Application Programming Interface.

BSI Bundesamt für Sicherheit in der Informationstechnik.

CIA Confidentiality, Integrity and Availability.

CKC® Cyber Kill Chain.

HTTP Hypertext Transfer Protocol.

IDS Intrusion Detection System.

GUI Graphical user interface.

IPS Intrusion Prevention System.

FPO Fachspezifische Prüfungsordnung.

HIPAA Health Insurance Portability and Accountability Act.

KI Künstliche Intelligenz.

MFA Multi-Faktor-Authentisierung.

ML Machine Learning.

NIST National Institute of Standards and Technology.

OTX Open Threat Exchange.

LML Log Monitoring Lackey.

OSSIM Open Source Security Information Management.

PCDI DSS Payment Card Industry Data Security Standard.

NOC Network Operations Center.

owasp® Open Web Application Security Project®.

RegExp Regular Expression.

SIEM Security Information and Event Management.

SEM Security Event Management.

SIM Security Information Management.

SOC Security Operations Center.

SSH Secure Shell Protocol.

USM Unified Security Management.

VM virtuelle Maschine.

1. Einleitung

Der heutige Netzwerkverkehr fast tausendfach größer als vor 20 Jahre (Roser et al., 2015). Das Internet wird heutzutage für fast alle unseren alltäglichen Tätigkeiten verwendet: Soziale Netzwerke, Video und Audio-Streaming, Einkauf, behördliche Angelegenheiten und viele andere. So viel Verkehr generiert eine unermessliche Menge von Daten, die alle möglichen Inhalte beinhalten, von unschuldigen Anfragen nach einem eigenen Kontostand bis zur Ausführung von beabsichtigten Anfragen, um Systeme lahmzumachen. Um das Erste vom Zweiten zu unterscheiden, verwenden viele Firmen das sogenannte Security Information and Event Management (SIEM) oder die Log Analysis Tools.

Das National Institute of Standards and Technology (NIST) definiert SIEM als Software für die Sammlung, Anpassung, Analyse, Überwachung und Bedrohungserkennung von Sicherheitsdaten aus verschiedenen Quellen, damit das zuständige Security Operations Center (SOC) Maßnahmen ergreifen kann (NIST, 2020b). Die Bewertung dieser Daten spielt eine wesentliche Rolle bei solchen Anwendungen, da es entscheidend ist zu wissen, ob es sich um normale Anfrage oder um Cyberangriffe handelt. Log Analysis und Log Management beziehen sich auf die Sammlung, Bearbeitung, Speicherung und/oder Löschen, Weiterleitung und Überwachung von Loginformationen. In dieser Arbeit benutzen wir den Begriff "Log Analysis Tool", um diese Systeme zu referenzieren.

In diesem Projekt recherchieren und vergleichen wir existierende SIEM und Log Analysis Tools. Danach entscheiden wir uns für eine Open Source Lösung. Mit dem ausgewählten Tool wollen wir spezifische Logdateien analysieren und bewerten, damit wir demnächst potenzielle Angriffe erkennen können. Die Angriffserkennung soll automatisch mit der Eingabe von vordefinierten Regeln der Mitre ATT&CK® Matrix stattfinden.

Unser Ziel ist, eine umfangreiche Open Source Lösung zu finden bzw. gesltaten, die uns ermöglichen, Cyberangriffe schnell und einfach zu detektieren. Proprietary Lösungen gibt es viele auf dem Markt. Sie kosten meisten sehr viel und verlangen spezielle Kenntnis für die Verwaltung. Da solche Lösungen eher an großen Unternehmen eingeschränkt sind, beschäftigen wir uns mit dem Aufbau und Strukturierung eine eigene Lösung mithilfe von Open Source Tools.

Diese Arbeit wird in folgende Teile geteilt:

- Definition von SIEMs und Log Analysis Tools
- Beschreibung von existierenden Proprietary und Open Source Lösungen
- Entscheidung für die Implementation einer Open Source Lösungen
- Installation und Konfiguration von der ausgewählten Anwendung
- Definition von zwei spezifischen Cyberangriffen
- Festlegung von Regeln oder Use Cases für die automatische Erkennung von der vorherigen definierten Angriffen anhand der Mitre ATT&CK® Matrix
- Empfang, Bearbeitung und Eingabe in der ausgewählten Lösung der spezifischen Logdateien der Hochschule

1.1. Problemstellung

Während der Entwicklung dieser Arbeit wollen wir uns mit folgenden Fragen beschäftigen:

- Wie können wir ein Log Analysis Tool so konfigurieren, dass es vordefinierten Angriffe nach der Mitre ATT&CK® Matrix automatisch erkennen kann?
- Wie können wir eine allgemeine Uses Cases definieren, sodass wir sie später für verschiedene Angriffsmuster nach Mitre ATT&CK® Matrix leicht anpassen können?

1.2. Vorgehensweise

Um diese oben genannten Ziele zu erreichen, verwenden wir folgenden Methoden:

- Recherche in der Fachliteratur über SIEMs und Log Analysis Tools Lösungen
- Vergleich zwischen verschiedenen Open Source und Proprietary Lösungen
- Installation von dem ausgewählten Tool
- Importieren von Logdateien in der ausgewählten Lösung
- Definition der Use Cases für die Angriffe

Das folgende Diagramm stellt den Aufbau dieser Arbeit dar:

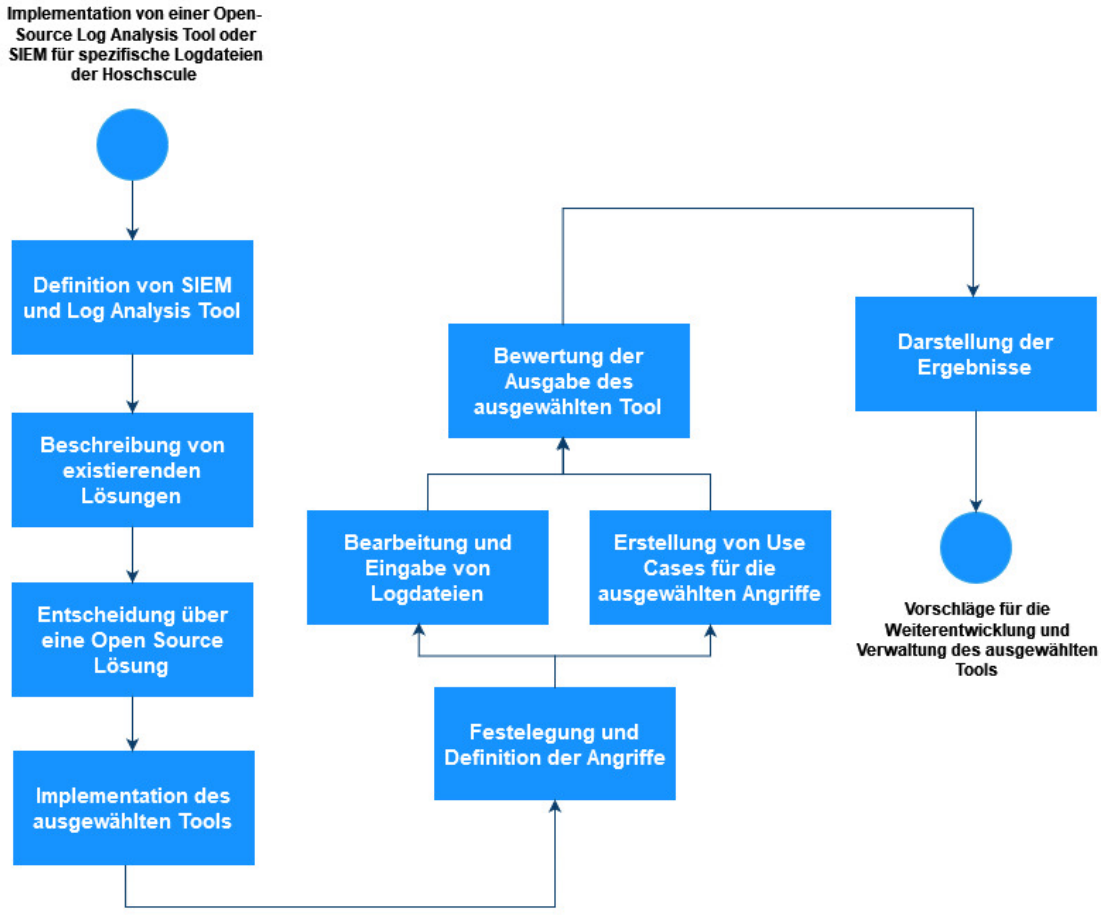

Abbildung 1: Aufbau dieser wissenschaftlichen Recherche SIEM
Quelle: Eigene Darstellung

2. Definition von SIEMs und Log Analysis Tools

SIEM ist das Ergebnis einer Kombination zwischen dem Security Event Management (SEM) und Security Information Management (SIM) (Dorigo, 2012). Das Erste bezieht sich auf die Identifizierung, Bewertung, Beobachtung und Bericht von Sicherheitsvorfällen mithilfe von verschiedenen Log Dateien (techopedia, 2015). Das Zweite ist eine Software, die bei der automatischen Sammlung von Loginformationen aus vielen Quellen, wie Firewall und Servers unterstützt (techopedia, 2022). Da die meisten SIEM-Lösungen kostenpflichtig sind, existieren auch viele Open Source Log Analysis Tools, die eine ähnliche Aufgabe erledigen, ohne die Kernelementen von SIEM zu besitzen.

Log Analysis Tools sind meistens Anwendungen die Logdateien empfangen, speichern, bearbeiten und nach spezifisichen eingegebenen Regeln bewerten. Diese Tools unterstützen Programmierer und Systemadministratoren bei der Überwachung des Zustands Systemen oder Software. Ein solches Tools kann Logdateien von verschiedenen Endpoints und mit verschiedenen Formattierungen bekommen und editieren, so dass es schließlich ein Bericht oder Graphik erzeugt (Łukasz Korzeniowski and Goczyła, 2022). Die Nutzung dieser Tools schränkt sich nicht in dem Sicherheitsbereich ein, sondern kann für das gesamte Rechenzentren nützlich sein.

In dem Universum des SOC mischen sich verschiedene Begriffe, die manchmal zur Verwirrung führen, weil sie ähnliche Bedeutung und Verantwortung haben. Intrusion Detection System (IDS), Intrusion Prevention System (IPS), Security Information and Event Management (SIEM) und Log Analysis Tools werden von *nonnative users* und sogar von Spezialisten oft verwechselt, da ihre Aufgaben mehr Zusammenhang als Unterschied haben. Um den Umfang dieser Arbeit wegen der zeitlichen Einschränkungen zu verringern, fassen wir kurz die Unterschiede zwischen denen zusammen und legen unsere Grenze auf den SIEMs Lösungen und auf Log Analysis Tools fest.

Intrusion Detection System (IDS) sind Software oder Hardware, die Cyberangriffe identifizieren und berichten. Sie haben eine passive Rolle, da sie die Cyberangriffe weder stoppen noch verhindern können. Intrusion Prevention System (IPS)allerdings haben eine aktive Haltung gegenüber Cyberangriffe - die können automatisch behandeln können, indem sie Blocking-Mechanismen einschalten, um den Angriff zu stoppen (Wendzel, 2018). Wie das Intrusion Detection System (IDS), kann das Intrusion Prevention System (IPS) auch Logdateien generieren, die von einer SIEM-Lösung gesammelt werden können. SIEMs können seinerseits die Logdateien von diesen und von anderen Endpoints bekommen und diese nach vordefinierten Regeln bewerten, um dem SOC-Team über Sicherheitsvorfälle zu informieren oder automatisch Maßnahmen zu ergreifen. Wie SIEMs bekommen Log Analysis Tools auch Logdateien, um Bericht oder Darstellung zu genieren. Ihre Nutzung ist aber nicht so spezifisch wie die von SIEMs.

Die folgende Abbildung stellt didaktisch eine allgemeine Struktur von SIEM-Lösungen dar:

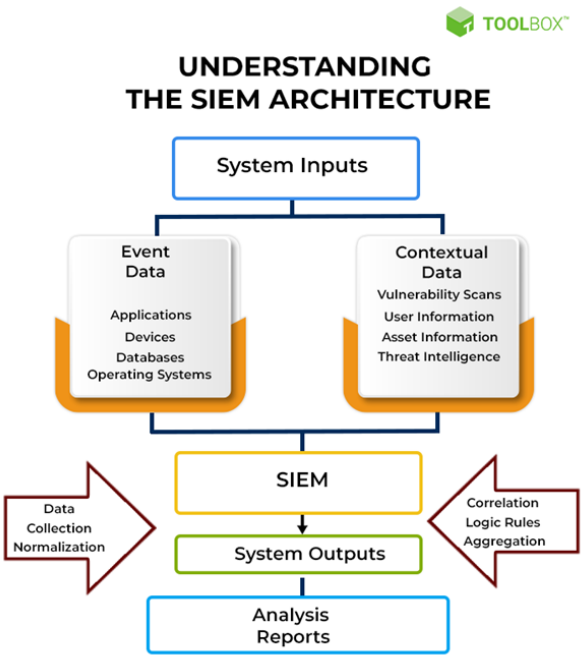

Abbildung 2: Allgemeine Struktur von SIEM
Quelle: (Mohanan, 2022)

Aus dem Bild können wir feststellen, dass SIEMs für die Zentralisierung von Sicherheitsdaten zuständig ist. Diese werden dann bearbeitet und in einem oder mehreren Berichten dargestellt, damit das SOC-Team schnellere und effektive Entscheidungen treffen können. Der Informationsfluss einer SIEM-Lösung können wieder in der folgenden Abbildung darstellen werden:

Abbildung 3: Allgemeine Informationsfluss von SIEM
Quelle: (Granadillo et al., 2021)

Die folgende Abbildung stellt eine allgemeine Architektur von Log Analysis Tools dar:

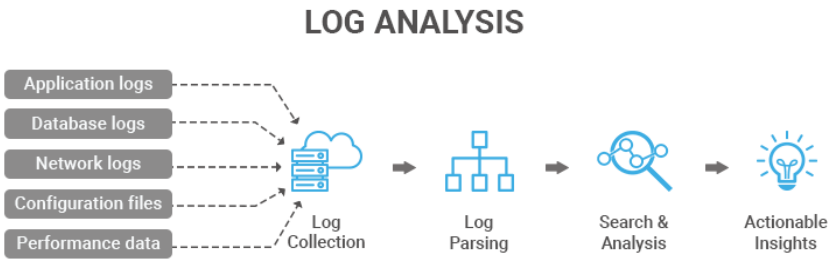

Abbildung 4: Allgemeine Struktur von Log Analysis Tools
Quelle: (Tek-Tools, 2020)

Den Informationsfluss eines Log Analyst Tools bildet folgendes Bild ab:

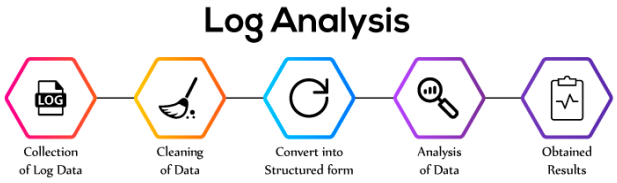

Abbildung 5: Allgemeine Informationsfluss von Log Analysis Tools
Quelle: (neptune, 2023)

Aus bisheriger Beschreibung stellen wir fest, dass SIEM viel mehr als eine Sammlung von Logdateien sind. Das Ziel dieser Software ist die automatische Analyse zu ermöglichen, indem Daten kombiniert und bewertet werden können. In vielen Bereichen, wie Finanzen (Payment Card Industry Data Security Standard (PCDI DSS)), Gesundheitswesen (Health Insurance Portability and Accountability Act (HIPAA)), sind SIEMss eine gesetzliche Verpflichtung (Jog, 2020). In Deutschland verpflichtet das zweite Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme Organisationen mit kritischen Infrastrukturen die Anwendungen solcher Lösungen, um Störungen der Confidentiality, Integrity and Availability (CIA) zu verhindern (BSI, 2021). Log Analysis Tools sind seinerseits allgemeine Tools zu der Speicherung, Anpassung, Bewertung und Darstellung von Logdateien, ohne dass sie sich auf der Sicherheitsebenen fokussieren.

2.1. Existierende SIEMs Lösungen und Log Analysis Tools

Die existierenden SIEMs und Log Analysis Tools können in zwei Kategorien getrennt werden: *Proprietary* und *Open Source*. In folgenden Abschnitten präsentieren wir die proprietäre SIEM Splunk, um einen Maßstab für unsere Auswahl zu definieren, wenn es um Funktionalitäten geht. Wir analysieren folgende SIEMs und Log Analysis Tools:

- Prelude
- AlienVault Open Source Security Information Management (OSSIM)
- FortiSIEM
- ELK-Stack
- Grafana

2.1.1. Splunk

Splunk von dem Unternehmen Splunk Technology wurde 2003 in den USA veröffentlicht (Splunk, 2022b). Er gehört weltweit zu der meistverwendeten SIEM-Softwares und gilt als *State of the art* für andere ähnliche Lösungen (Kazarov et al., 2018). Zu ihren Kunden gehören große Konzerne wie Airbus, Coca-Cola, Intel und Deutsche Bahn.

Splunk bietet laut seiner Webseite folgenden Funktionalitäten an (Splunk, 2015):

- Skalierbare Datenplattform
- Risk-based Warnmeldung
- Bedrohungserkennung mithilfe von Machine Learning (ML)
- Automatische Aktualisierung von der Bedrohungs- und Schwachstelle-Database
- Unkomplizierte Installation und Anwendung

Die allgemeine Architektur und der Informationsfluss von Splunk unterscheiden sich nicht von den obigen dargestellten Struktur 2, Seite 5, und Informationsfluss3, Seite 6. Da es sich hier um eine proprietäre Lösung handelt, lässt sich Splunk mit vielen anderen Funktionalitäten verwalten und erweitern.

In Splunk funktioniert die Bedrohungserkennung mithilfe vonUses Cases. Laut der Dokumentation existieren sie in folgenden Szenarien: Überwachung, Untersuchung und Erkennung. Die Software ist sowohl mit glsmitre Matrix als auch mit Cyber Kill Chain (CKC®) für die Gestaltung ihrer Uses Cases integriert (Splunk, 2022a).

In einer spezifischen Arbeit wurden Angriffe auf einem System simuliert und schließlich mit Splunk analysiert, um Gefahren zu identifizieren und diese im Voraus zu sehen (Su et al., 2016). In anderer Arbeit beschrieben die Autoren, wie eine Splunk-Instanz installiert und konfiguriert wurden, um spezifische Brute-Force Angriffe zu erkennen (Selvaganesh et al., 2022).

2.1.2. Prelude

Das im Jahr 2002 in Frankreich von Yoann Vandoorselaere freigegebene Tool Prelude zählt zu einer europäischen Open Source SIEM Lösung. Laut dem Anbieter verfügt Prelude unter anderem folgende Funktionalitäten (Prelude SIEM, 2018):

- Informationszentralisierung
- Datenaggregation und -Zusammenhang mit vordefinierten und von dem Nutzer angepassten Regeln
- Einbruchserkennungsmechanismen
- Datennormalisierung

Die Anwendung besteht aus verschiedenen unabhängigen Modulen. Unter denen highlighten wir Warnmeldung, Archivierung, Analyse und Verwaltung. Das Erste gehört zu der zentralen Aufgabe dieser Lösung - es ist dafür zuständig, Daten zu empfangen, zu normalisieren, Zusammenhänge zu erschließen und Meldungen zu generieren. Das zweite Modul - Archivierung konzentriert sich auf die Speicherung und Verfügbarkeit der Daten. Zu dem Analyse-Modul gehören statistische Aufgabe und Darstellung in verschiedenen Formaten. Das letzte Modul dient dazu, die Anwendung zu steuern, Nutzer zu erstellen, dessen Rechte zu konfigurieren (European Comission, 2015).

Die folgende Abbildung zeigt die Integration verschiedener Module von Prelude und wie sie mit einander kommunizieren, um Analyse, Meldung und Speicherung zu generieren:

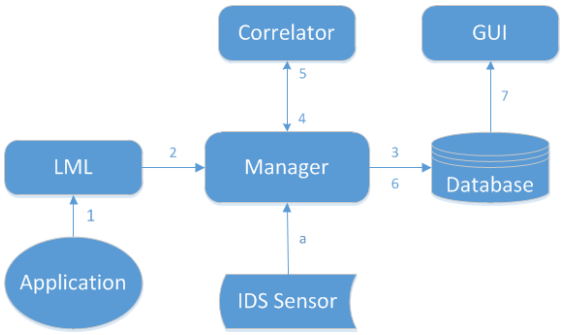

Abbildung 6: Integration zwischen den Modulen von Prelude
Quelle: (Prelude Team, 2007)

Aus der Abbildung und der Dokumentation können wir folgenden Informationsfluss erkennen - die Daten werden von Endanwendung generiert und zum Loganalyzer (Prelude Log Monitoring Lackey (LML)) geschickt, wo sie normalisiert und bewertet werden. Für solche Logs, wo es verdächtige Werte gibt, werden Warnmeldungen generiert. Diese Meldungen werden zum Manager Module weitergeleitet. Der Correlator oben sucht nach einem Zusammenhang zwischen anderen Daten. Das Ergebnis von Correlator wird wieder zum Manager geschickt und danach zu der Datenbank. Schließlich stehen die Berichte in dem User-Interface zur Verfügung (Prelude SIEM, 2020).

Die Architektur der Anwendung ermöglicht sowohl einen zentralisierten als auch einen dezentralisierten Aufbau. In der nächsten Abbildung sehen wir eine einfache Implementation von Prelude:

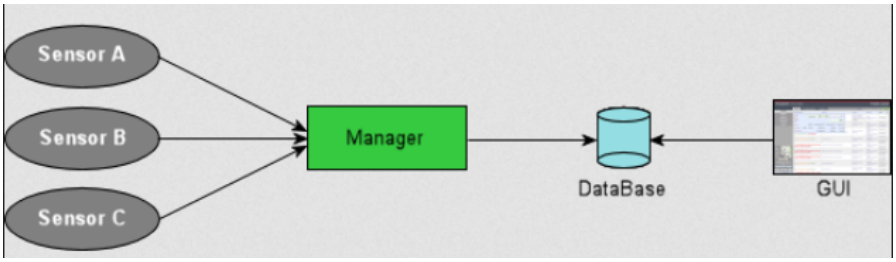

Abbildung 7: Einfache Architektur von Prelude
Quelle: (Prelude Team, 2007)

In einer dezentralisierten Umgebung werden Daten von verschiedenen und getrennten Quellen generiert und bearbeitet. schließlich können die Nutzer auf diesen Daten unter einem Graphical user interface (GUI) zugreifen.

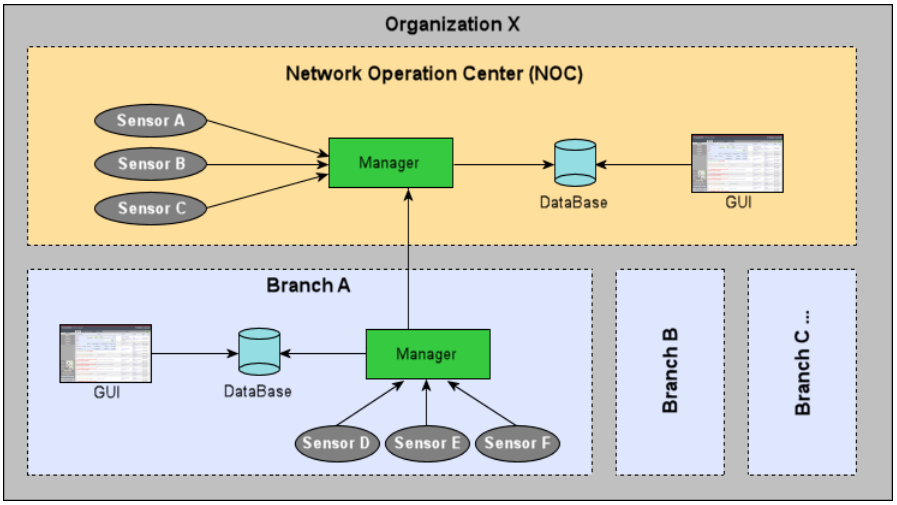

Abbildung 8: Erweiterte Architektur von Prelude mit der Nutzung von dezentralisierten Datenquellen und Bearbeitung
Quelle: (Prelude Team, 2007)

Die wissenschaftliche Literatur über Prelude ist sehr eingeschränkt. Wenige Publikationen fokussieren sich auf die Entwicklung, Implementation und unternehmerische Anwendung dieses Tools. Eine Studie von 2021 versuchte dieses und zwei andere Tools (AlienVault und Cyberoam iView) anhand technischer und nutzerfreundlicher Kriterien zu vergleichen. Unter diese Kriterien highlighten wir folgende(Radoglou-Grammatikis et al., 2021):

- **technische Kriterien**
  - *Real-time performance*,
  - *Range and flexibility of reporting*
  - *Alert correlation*
- **nutzerfreundliche Kriterien**
  - *Documentation comprehensiveness*
  - *Complexity of the installation process*
  - *Complexity of the system configuration*

In den technischen Kriterien lag Prelude auf dem dritten Platz und in den benutzerfreundlichen Kriterien bekam Prelude den ersten.

Auch in den nicht wissenschaftlichen Publikationen existiert eine begrenzte Anzahl von Texten über Preludes. Die existierenden Texte kommentieren ganz zusammenfassend die ausreichende Dokumentation und heben hervor, dass es eher eine in Europa konzentrierte Lösung ist.

2.1.3. AlienVault OSSIM

AlienVault OSSIM ist eine im Jahr 2007 entwickelte Open Source SIEM Lösung. Im Jahr 2018 wurden sie von der Firma AT&T Communication gekauft (CBNINSIGHTS, 2020). In der Beschreibung des Anbieters steht, dass er sie auch dabei unterstützt, Daten zu sammeln, zu normalisieren und zu bewerten. Er behauptet auch, dass sein Tool in der Lage ist, Schwachstellen und Angriffe zu erkennen, das Verhältnis zu beobachten und Datenzusammenhang zu erschließen (AT&T Cybersecurity, 2022).

AlienVault hat eine kostenpflichtige Version, die Alien Vault Unified Security Management (USM) heißt. In der Webseite von AT&T steht, dass es keine spezifische Dokumentation für die Open Source Version, AlienVault OSSIM, gibt, da viele Funktionalitäten von der anderen Version stammen (AT&T Cybersecurity, 2022).

Die folgende Abbildung zeigt das von dem Anbieter freigelegte Architekturdiagramm von der USM Version:

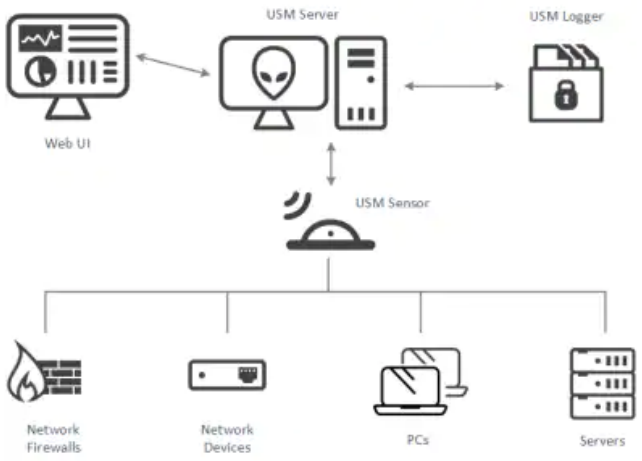

Abbildung 9: Architekturdiagramm von AlienVault USM
Quelle: (AT&T Cybersecurity, 2022)

Laut der Website Comparitech steht AlienVault in der 13ten Platz von den besten bewerteten SIEM-Lösungen. Die Seite beschreibt auch, dass zu dem Tool einen IDS, ein Verhaltensüberwachungssystem und einen Schwachstellen-Scanner integriert sind. Die Anwendung ist auch mit der Plattform Open Threat Exchange(OTX) verbunden - diese ermöglicht eine Teilung von Informationen über die Schwachstelle. Comparitech highlighted, dass die Anwendung wegen ihrer niedrigen Kosten besser für kleine oder mittelständige Unternehmen geeignet ist (comparitech, 2023).

Die Anwendung soll konsistenten Daten Zusammenhang anbieten und soll das Auftauchen von Falsch Positiv vermeiden. AlienVault kommt auch mit vordefinierten Use-Cases, die dabei unterstützen, gewöhnlichen Angriffsszenario zu erkennen. Die Installation, die Einstellung und die Integration mit anderen Tools ist auch benutzerfreundlich (Gómez et al., 2022). Aus einer anderen wissenschaftlichen Quelle fanden wir heraus, dass für viele Quellen eine manuelle Normalisierung der Logdatein notwendig ist (Nabil et al., 2017). Die Anwendung hat aber einen zuverlässigen Berichtsmechanismus.

Während unserer Recherche gab es wenig wissenschaftliche Literatur, die sich um AlienVault OSSIM kümmert. Die meisten Publikationen waren aus kommerziellen Quellen und diese konzentrierten sich auf eine kostenpflichtige SIEM-Lösung von AT&T..

2.1.4. FortiSIEM

FortiSIEM ist eine US-amerikanische SIEM-Lösung von der Firma Fortinet. Fortinet kaufte im Jahr 2016 das Unternehmen AccelOps und dessen SIEM-Lösung und benannte es zum FortSIEM (Fortinet, 2016).

Laut dem Anbieter hat FortiSIEM eine robuste Integration mit anderen Tools und lässt sich leicht und einwandfrei skalieren. Andere Versionen des Tools sind mit Machine Learning (ML) integriert, sodass die Anwendung auch Verhältnisanalyse durchführen kann (Fortinet, 2022). Das Tool bietet auch eine umfangreiche und ausführliche Dokumentation an. Die nächste Abbildung zeigt die skalierbare Architektur des Tools:

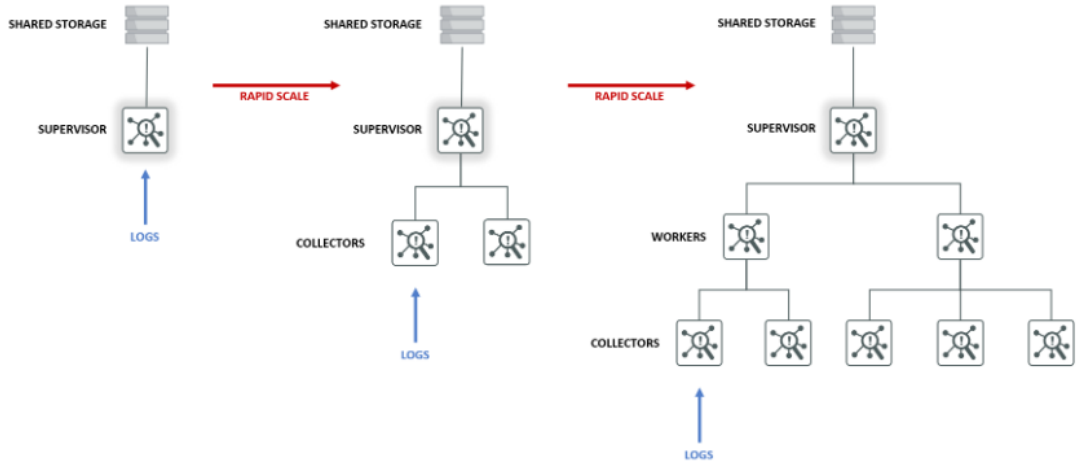

Abbildung 10: Skalierbare Architektur von FortiSIEM
Quelle: (Fortinet, 2020)

Auch zu dieser SIEM-Lösung ist die wissenschaftliche Produktion eingeschränkt. Eine von der gefundenen Publikation betont, dass FortiSIEM eine schnelle Erkennung von Angriffen anbietet und über Network Operations Center (NOC) Funktionalitäten verfügt (Ramírez Tomás, 2018). Wie andere SIEMs Lösungen, hat FortiSIEM folgende Funktionalitäten:

- Datensammlung und Normalisierung
- Daten Zusammenhang
- Generierung von Berichten
- Warnmeldungen
- Datenauswertung

2.1.5. ELK-Stack

ELK Stack stammt aus der Verbindung von drei Tools: Elasticsearch, Logstash und Kibana. Das Erste ist eine Such-und Analyse-Maschine. Das Zweite ist eine serverseitige Anwendung zur Datenverarbeitung und -Weiterleitung. Schließlich Kibana ist dafür zuständig, visuelle Darstellung in Grafik-Format auszugeben (packt, 2019). Von diesen drei Tools Logstasch ist der einzige Open Source (elastic, 2021). Obwohl die anderen zwei kostenlos verwendet werden können, gehören sie nicht zu der Open Source Kategorie (Open Source Initiative, 2007). Dieses Tool besitzt viele Eigenschaften einer SIEM-Lösung und wird von vielen SOC verwendet, ist aber, für viele Experten, kein SIEM für sich, da es über keine Warnmeldungssystem, Daten Zusammenhang und Vorfälleverwaltung verfügt (Miller, 2021). Diese und anderen Funktionalitäten lassen sich aber durch Plugins integrieren.

Das folgende Diagramm stellt die Architektur von ELK Stack mit ihren integrierten Elementen dar:

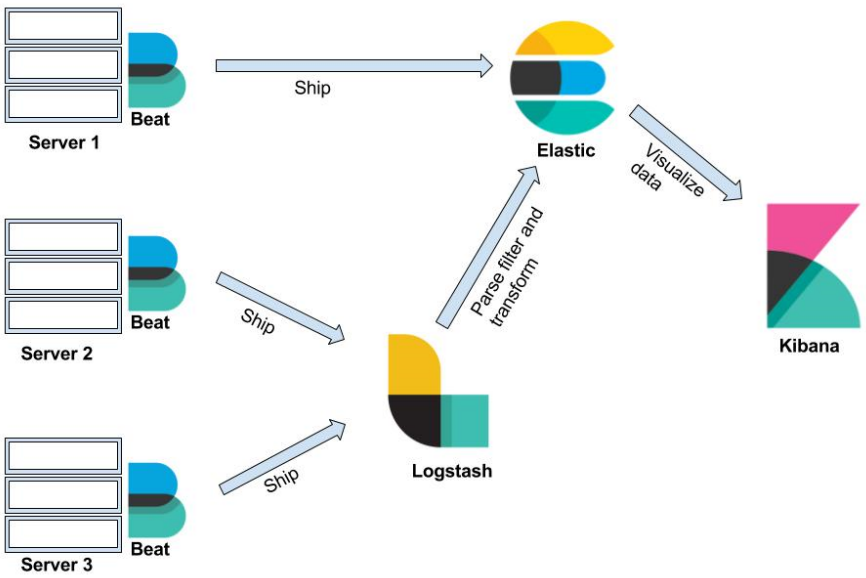

Abbildung 11: Integration zwischen Elasticsearch, Logstash und Kibana
Quelle: (packt, 2019)

Die Beats auf dem Bild sind an der Endpoints installiert und leiten Daten entweder zu Elasticsearch oder zu Logstash weiter, wo sie schließlich bearbeitet werden (Jain, 2018).

Ein Teil der wissenschaftlichen Literatur zeigt die Log Analyse-Funktionalitäten von ELK Stack und die Unterstützung bei Normalisierung und Indexierung von Daten für eine lesbare Ausgabe (Advani et al., 2020). Die starke Skalierbarkeit wurde auch bei einer Studie erwähnt, wo ELK Stack für Wi-Fi Logging eingesetzt wurde (Wang et al., 2019).

Die offizielle Dokumentation von ELK Stack betont, dass die Anwendung folgende Funktionalitäten besitzt (elastic, 2022):

- Datensuche, -Normalisierung, -Analyse und
- Speicherung
- visuelle Ausgabe

Folgendes Diagramm aus der offiziellen Dokumentation zeigt die Aufteilung der Funktionalitäten pro Element von ELK Stack:

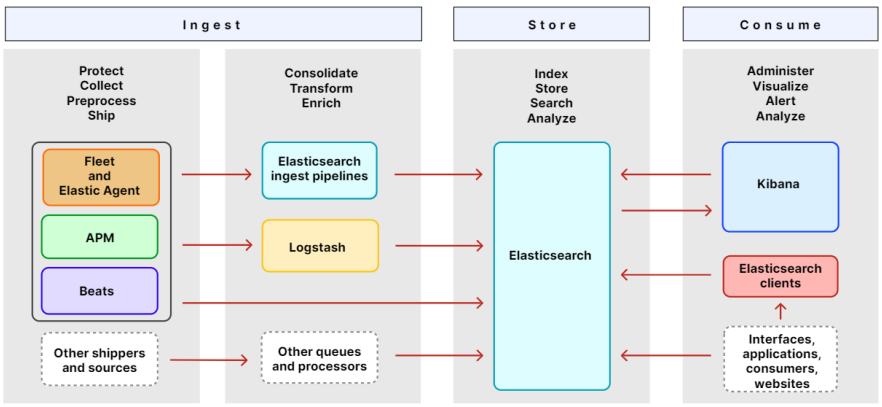

Abbildung 12: Aufteilung der Funktionalitäten zwischen den Komponenten
Quelle: (elastic, 2022)

Die wissenschaftliche Publikation über ELK-Stack ist vielfältiger als bei anderen recherchierten Tools. Es ist aber wichtig, zu betonen, dass die Mehrheit von denen sich eher mit dem Logging als mit den SIEM-Eigenschaften der Anwendung beschäftigt.

2.1.6. Grafana

Von allen recherchierten Lösungen ist Grafana die Einzige, die nicht als SIEM dargestellt ist. Grafana wird aber als Plattform für Visualisierung von Data beschrieben. Mit dem Tool ist es möglich eine Graphik zu erstellen und Meldungen zu definieren. Das Ziel der Anwendung ist, Information in einer einfachen und verständlichen Art und Weise zur Verfügung zu stehen (redhat, 2022).

Im Jahr 2014 wurde Grafana von der Firma Grafana Labs veröffentlicht. Das Tool basiert auf Kibana3,2.1.5. Ursprünglich sollte Grafana ein einfacheres Editingstool für Graphik sein und ermöglichen, Datenanfragen unkomplizierter zu machen. Die neuste Version, 9.4.3. wurde im März 2023 veröffentlicht und bietet viele Funktionalitäten an. Es ist auch möglich das Tool mithilfe von Plugins zu erweitern (Ödegaard, 2019)..

In der Webseite betont der Anbieter, dass Grafana die Zentralisierung und Zugang von Daten vereinfachen. Alle Art von Daten lassen sich analysieren und darstellen, von der Leistung von Anwendungen bis Verkaufsdaten und Krankheitsfällen. Die Anwendung soll auch den Zusammenhang von Daten ermöglichen, um wichtige Informationen herauszunehmen (Grafana Labs, 2016).

Grafana ist auch mit dem Logging Tool Loki und Promptail integriert. Promtail ist für Sammlungen der Logdateien und Weiterleitung an Loki zuständig. Promptail wird an jeden Endpoint installiert. In Loki werden diese Logdateien ohne Index für den schnellen Zugriff gespeichert. Diese Daten können dann in Grafana mithilfe der Abfragesprache LogQL aufgerufen werden. Schließlich können Warnmeldung mit spezfisichen Regel generiert werden, die in Loki eingeführt werden (Grafana Labs, 2018). Auf dem Folgenden Bild wird die Struktur von Grafana Loki dargestellt:

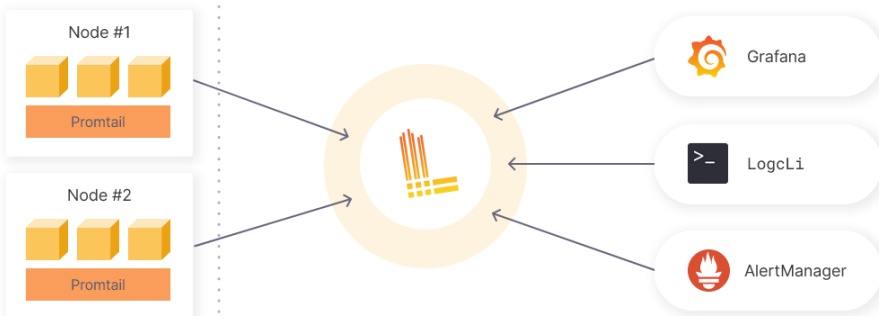

Abbildung 13: Integration von verschiedenen Log-Quellen mit Grafana Loki und Promptail
Quelle: (Grafana Labs, 2022a)

Das Tool hat auch eine umfangreiche Dokumentation, die ausführlich erklärt, wie sie sich installieren, bedienen und mit anderen Tools integrieren lässt.

Die wissenschaftlichen Literatur über Grafana konzentriert sich eher auf die Anwendung des Tools für die graphische Darstellung von Daten als für ihre Nutzung in dem Sicherheitsbereich. Einw Recherche, z.B., wollte das Ergebnis von der Überwachung von Cloud-Based Systemen, von Netzwerkaktivitäten und von Netzwerkverkehr mithilfe von

Grafana darstellen (Manases and Zinca, 2022). In dieser Hinsicht gibt es wenige wissenschaftliche Arbeit, wo die Implementation und Integration von Grafana mit anderen Tools für den Sicerheitsbereich die Hauptfigur ist.

2.2. Auswahlkriterien

Eine umfangreiche SIEM Software die viele automatische Lösung für die Erkennung und Bekämpfung von Cyberangriffe würde perfekt für jede Situation passen. Da solche Lösungen meistens (oder alle) Proprietary sind und nur für teure Preise angeboten werden, entschieden wir uns für die Anpassung an einem Open SourceTool, das zu unserem Kontext und Einschränkungen gehört.

Demnächst beschäftigen wir uns mit Grafana. Wir beschreiben, wie wir das Tool installieren, konfigurieren und mit verschiedenen Logdateien eingeben. Nachdem die Grundfunktionalitäten eingerichtet sind und einwandfrei funktionieren, generieren wir anhand der Mitre ATT&CK® Matrix Uses Cases für die zukünftigen ausgewählten Angriffe. Unser Ziel ist Grafana so einzustellen, dass es in der Lage ist, die Muster dieser Angriffe zu erkennen und darüber zu berichten.

3. Implementation

In diesem Kapitel beschäftigen wir uns mit der Implementation und mit dem Aufbau von Grafana, sodass wir Cyberangriff nach dem Mitre ATT&CK® Matrix erkennen können. Wir gestalten unser Arbeitslabor mit Container und virtuellen Maschine (VM), wie in dem folgenden Diagramm dargestellt:

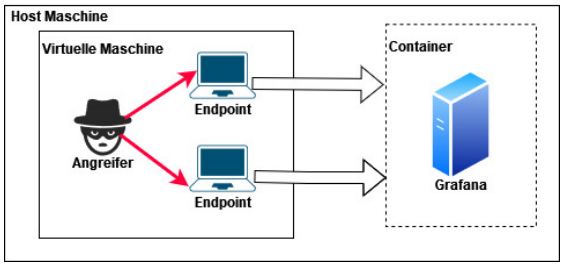

Abbildung 14: Aufbau unseres Arbeitslabors
Quelle: Eigene Quelle

Von unserem Aufbau wollen wir folgende Ziele erreichen: Aufnahmen und Anpassung der Logdateien für Grafana, Musterkerennung für die ausgewählten Cyberangriffe und schließlich Warnmeldung für die Endnutzer, damit sie geeignete Sicherheitsmaßnahmen ergreifen können.

Der gezielte Ablauf ist in dem folgenden Diagramm dargestellt:

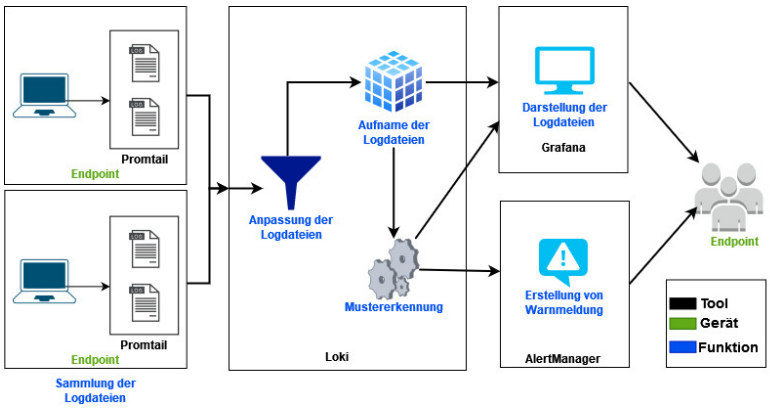

Abbildung 15: Erwarteter Ablauf der Sammlung der Logdateien bis zur Warnmeldung
Quelle: Eigene Quelle und (Grafana Labs, 2018)

3.1. Angriffserkennung anhand der Mitre ATT&CK Matrix®

Es gibt verschiedene Methoden und Frameworks zur Vermeidung, Erkennung und Unterbrechung von Cyberangriffe. Open Web Application Security Project®(OWASP®), Cyber Kill Chain (CKC®) und die Mitre ATT&CK® Matrix sind einige Beispiele, die von SOC-Teams verwendet werden, um die Sicherheit von Systemen und/oder Netzwerken zu gewährleisten. Da sich die Richtlinien und Schwerpunkte dieser Frameworks unterscheiden können und deshalb einen anderen Aufbau unserer Struktur erfordern könnten, haben wir uns für die Anwendung der Mitre ATT&CK® Matrix zur Erkennung von Cyberangriffe entschieden, insbesondere weil dieser Framework auch in Splunk integriert ist.

Die Mitre ATT&CK® Matrix hat folgende Hauptnutzung (MITRE ATT&CK, 2018b):

- Erkennung und Analyse von Angriffstechnik
- strukturierte Datensammlung über Bedrohungen
- Emulieren von Cyberangriffe für die Anwendung an Angriffsübungen
- Systemhärtung und Verbesserung der Verteidigungsmaßnahmen

Die Matrix bietet eine umfangreiche Verwendung für Unternehmen und für SOC-Team an, um ihre wertvollen Ressource schützen und ihre Fachkenntnisse über Cybersicherheit zu erweitern (Hazel, 2021). Hier konzentrieren wir uns auf die Entwicklung und auf die Implementierung einer Methode für die automatische Erkennung und Analyse von AngriffsTechnik in Grafana.

Die Mitre ATT&CK® Framework besteht aus 14 Taktik. Zu jedem Taktik gehören Technik, die ihrerseits in SubTechniks aufgeteilt sind. Jede SubTechnik wird mit Beispielen, Härtungsmaßnahmen und Erkennungsregeln dargestellt.

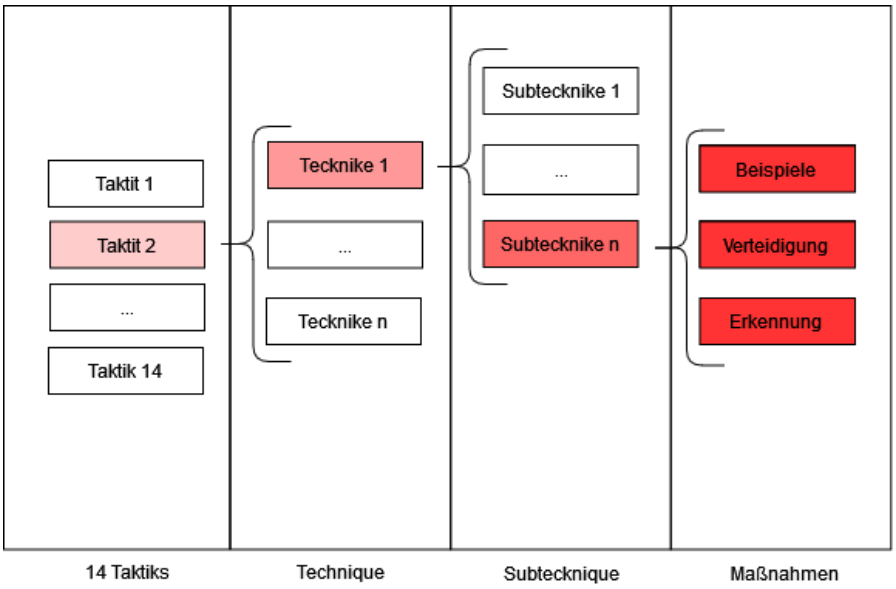

Abbildung 16: Struktur der Mitre Matrix
Quelle: Eigene Quelle und (MITRE ATT&CK, 2018b)

Die 14 Taktiks sind folgende:

- Informationssammlung für zukünftige Angriffe
- Entwicklung von Ressource von Angreifer
- Erster Zugang zum Opfersysteme
- Ausführung von bösartigen Coden
- Beharrlichkeit von System
- Privilegienausweitung
- Vermeidung von Verteidigungssysteme
- **Zugang zu Anmeldedaten**
- Umgebungserkennung
- Seitliche Bewegung zu anderem Systemen innerhalb des Angriffsziels
- interne Informationssammlung
- Steuerung und Kontrolle (C2 - Command and Control im Original)
- Datenextrahierung
- Auswirkung auf die Integrität

3.2. Auswahl des Angriffes

In dieser Arbeit beschäftigen wir uns mit dem Taktik "Zugang zu Anmeldedaten" und deren Technik Brute-Force Angriffe. Diese Technik ist in vier SubTechnik aufteilt:

- Erraten von Anmeldedaten
- Entschlüsselung von Hashwerte
- *Password Stuffing*
- *Password Spraying*

Da unser Ziel hier ist Grafana, zu benutzen, um Angriffe zu erkennen, entschieden wir uns für einen einfachen reproduzierbaren Angriff, die weniger Ressource verlangt. In diesem Fall, lässt sich Brute-Force Angriffe einwandfrei mit zwei VMs darstellen. Für diesen Angriff benutzen wir die SubTechnik "Erraten von Anmeldedaten und *Password Stuffing*", da sie ähnliche Erkennungmethode haben. Hier schließen wir auch die anderen Maßnahmen aus.

Die nächste Abbildung zeigt den Umfang unseres Implementationsversuchs:

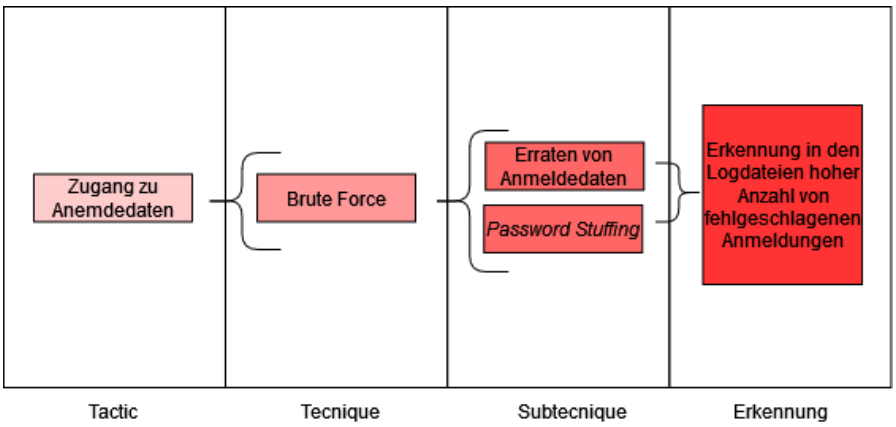

Abbildung 17: Analysestruktur für diese Arbeit Cyberangriffe
Quelle: Eigene Quelle und (MITRE ATT&CK, 2020)

3.3. Installation und Erstellung von Logdateien

In diesem Abschnitt fokussieren wir uns auf folgenden Punkten:

1. Einrichtung der VMs für Opfersystem und Angreifen
2. Angriffsssimulation für die für die Generierung von Logdateien
3. Installation und Einrichtung von Grafana Loki und Promtail mit Container
4. Weiterleitung der Logdateien zum Grafana

Die Installation und Anwendung können entweder mit dem Graphical user interface (GUI) oder mit der Kommandozeilen durchgeführt werden. In dieser Arbeit benutzen wir die Kommandozeile.

3.3.1. Einrichtung der VMs für Opfersystem und Angreifen

Die beiden VMs sind eine vorgebaute "Kali virtuellen Maschine (VM)" und "Ubuntu Server 22.04.2" in ihren standardmäßigen Einstellungen. Beiden Maschinen wurden nach der jeweiligen Dokumentation installiert (Kali, 2019) und (Ubuntu, 2023a).

Für das Opfersystemen entschieden wir uns für die Passwörte "qwertz" und "password". Laut einer Umfrage gehört dieses Passwort zu den zehn meisten verwendeten Passwort in Deutschland (silicon.de, 2022).

Für die Durchführung von Password Spraying erstellen wir folgende Benutzer und Passwörterkombinationen:

Opfersystem 1	Opfersystem 2
admin:123456	bob:hallo
user1:passwort	master:alice
user2:abc123	hans:daniel
user3:qwertyuiop	bruno:super123

3.3.2. Generierung von Logdateien mit der Angrifsssimulation

Für den Angriff verwenden wir folgenden Tools:

- Secure Shell Protocol (SSH)
- Hydra

In diesem Szenario schickt Hydra gleichzeitig mehrere Authentifizierungsversuche zum Opfersystem, um eine SSH-Verbindung mit dem Opfersystem herzustellen. Das Tool verwendet ein sogenanntes Wörterbuch mit verschiedenen Einträgen, die als Passwörter dienen. Für unseren Test benutzen wir die bekannte Rockyou-Datei.

Die nächste Abbildung zeigt, wie Password Stuffing abläuft:

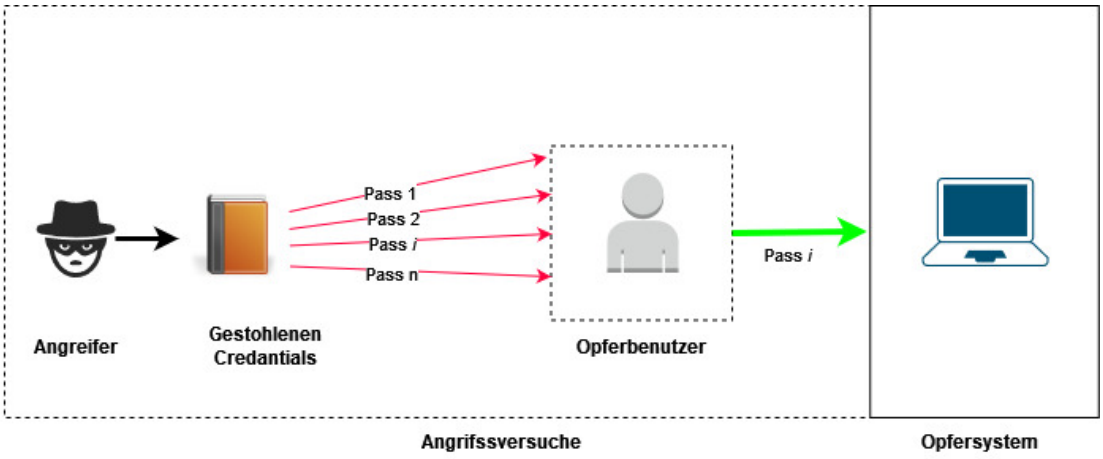

Abbildung 18: *Password Stuffing*
Quelle: Eigene Quelle und (Ba et al., 2021)

Password Stuffing wurde mit folgendem Kommando durchgeführt (Kali, 2022a):

```
hydra -l [Benutzername] -P rockyou.txt [Opfersystem] ssh -V -t 4

# Erklärung
-l: Spefizikation der Benutzername, den wir Angreifen
-P: Auswahl der Datei mit bekannten Passwörter
ssh: Auswahl der Anwendung, die wir angreifen wollen
-V: Ausführliche Ausgabe über Versuche, Fehler und Erfolg
-t 4: Anzahl von gleichzeitigen Verbindungen
```

Das folgende Bild zeigt ein Teil der Ausgabe von Hydra bei der Ausführung von Password Stuffing gegen das Opfersystem1:

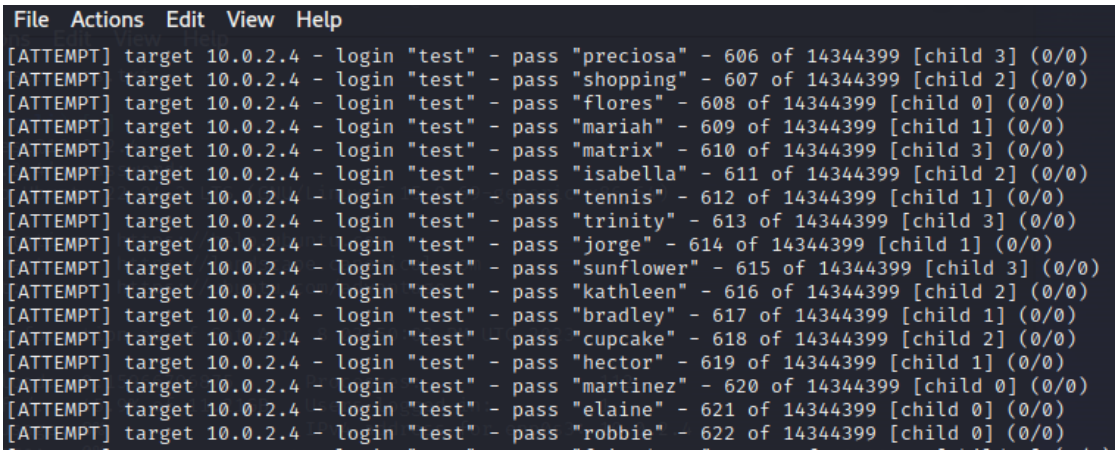

Abbildung 19: *Password Stuffing* gegen Opfersystem1
Quelle: Eigene Quelle und (Ba et al., 2021)

Und gegen Opfersystem2:

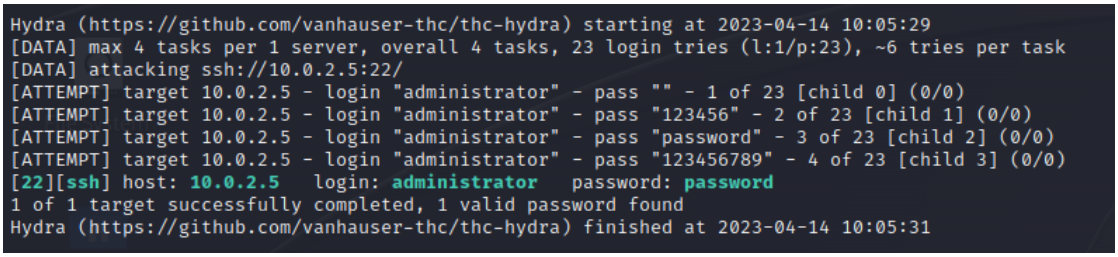

Abbildung 20: *Password Stuffing* gegen Opfersystem2
Quelle: Eigene Quelle und (Ba et al., 2021)

Unser nächster Angriff, Password Spraying, sieht wie folgende aus:

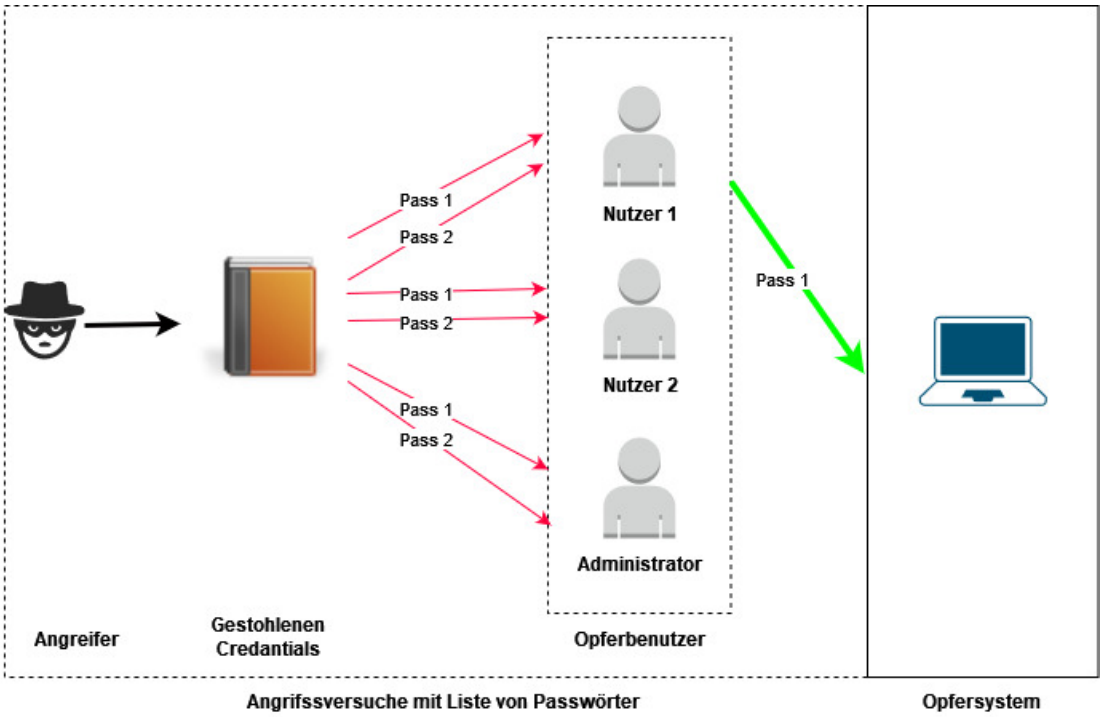

Abbildung 21: *Password Spraying*
Quelle: Eigene Quelle und (Swathi, 2022)

Für diesen Angriff benutzen wir folgendes Kommando:

```
hydra -L username2.txt -P passwoerter.txt [Opfersystem2] ssh -V -t 4

# Erklärung
-L: Auswahl der Datei mit gefunden Benutzername
```

In diesem Fall gehen wir davon aus, dass der Angreifer einige oder alle Benutzernamen schon kennen. Da es bei diesem Angriffe weniger Anmeldeversuche pro Nutzer stattfindet, benutzen wir eine selbsterstellte Datei mit weniger Passwörter als bei der Rockyou-Datei. Unsere Datei beinhalt die beliebigste Passwörter in Deutschland (silicon.de, 2022).

Die folgenden Screenshots zeigen die Ausführung von Password Spraying:

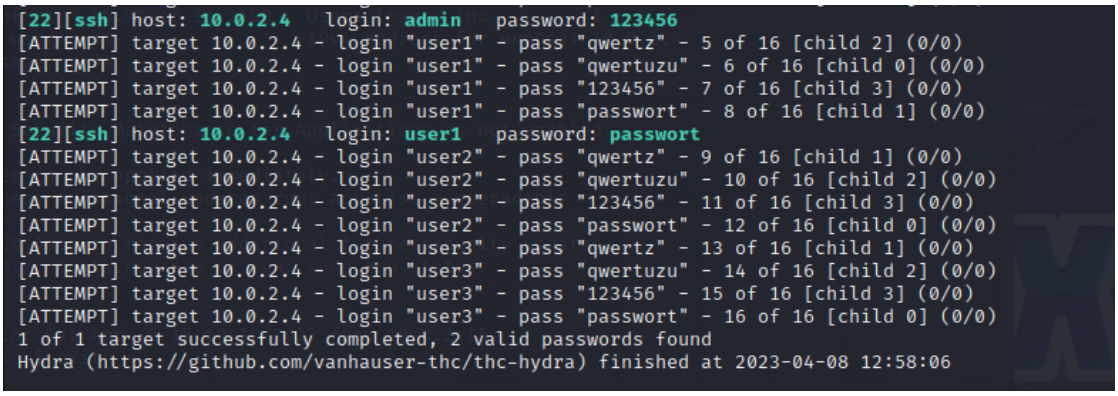

Abbildung 22: Ausführung *Password Spraying* in Kali Linux gegen Opfersystem1
Quelle: Eigene Quelle

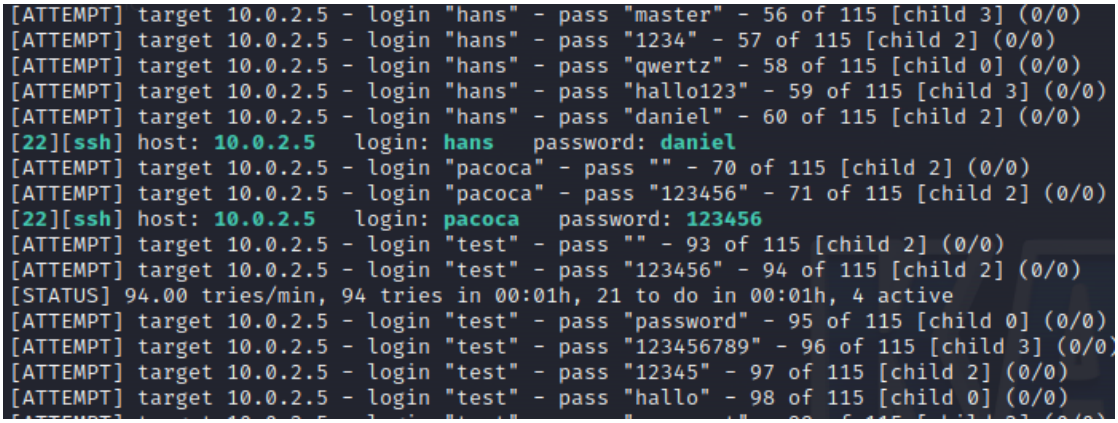

Abbildung 23: Ausführung *Password Spraying* in Kali Linux gegen Opfersystem2
Quelle: Eigene Quelle

3.3.3. Installation und Einrichtung von Grafana Loki und Promtail mit Container

Die offizielle Dokumentation von Grafana war nicht immer über die Ausführung eindeutig, deshalb benutzten wir auch fremde Quellen, um die Einstellungen an unsere Umgebung anzupassen (Polinowski, 2019). Unter befindet es sich die von Grafana zur Verfügung gestellte Konfigurationsdateien und Installationsverfahren (Grafana Labs, 2020a).

```
wget https://raw.githubusercontent.com/grafana/loki/v2.8.0/cmd/loki
/loki-local-config.yaml -O loki-config.yaml (die Datei wurde angepasst)

wget https://raw.githubusercontent.com/grafana/loki/v2.8.0/clients/
cmd/promtail/promtail-docker-config.yaml -O promtail-config.yaml
(die Datei wurde angepasst)

docker-compose -f docker-compose.yaml up
```

Im Anhang befinden sich die originalle (Siehe A) und die angepassten Dateien (Siehe B).

Die obigen Kommandos haben folgende Bedeutungen:

1. Herunterladen der Konfigurationsdatei von Loki
2. Herunterladen der Konfigurationsdatei von Promptail
3. Ausführung von den Containers, indem beiden Konfigurationsdateien in einer eingepackt und angepasst wurden und schliesslich von der Container-Anwendung gelesen werden

Für spezifische Versionen oder andere weitere Einstellungen bietet die Dokumentation umfangreiche Möglichkeiten an (Grafana Labs, 2020a).

Für diesen ersten Test, wurden die Logdatei des Opfersystems manuell zu dem Container übertragen.

Nach der Ausführung des Kommandos ist die Anwendung schon benutzbar, wie in dem folgenden Screenshot:

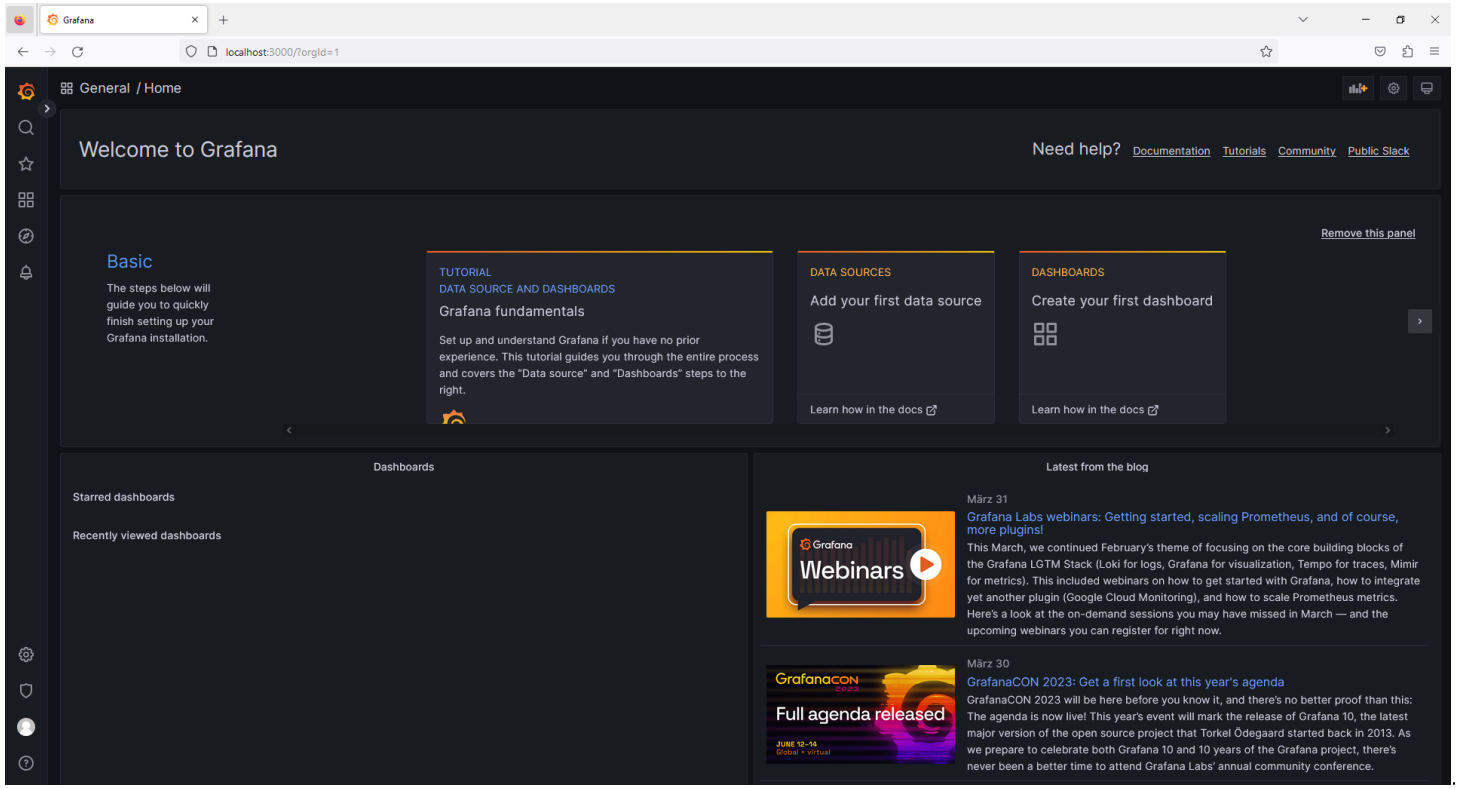

Abbildung 24: Screenshot der Willkommensseite von Grafana Loki
Quelle: Eigene Quelle und (Grafana Labs, 2022a)

3.3.4. Weiterleitung der Logdateien zum Grafana

Grafana Loki bietet mehrere Möglichkeiten an, Logdateien empfangen. In unserer Arbeit benutzen wir **Promtail**, der in einem Container läuft. Dieser Instanz schickt die von uns ausgewählten Logdateien Grafana und bearbeitet alle diese Dateien innerhalb eines sogenanntes "job". Hätten wir verschiedene Art von Logdateien, würde jeder Typ einem "job" zugewiesen (Grafana Labs, 2021b). Jeder "job" hat ihre eigene Regeln, um nach dem gewünschten Information zu suchen.

In einen produktive Umgebung wäre die Installation von **Grafana Agents** in jedenm Endpoint eine andere Lösung, um Grafana Loki mit Logdateien befüllen. In diesem Fall würde jeder Endpoint mithilfe von Promtail die Dateien weiterleiten (Grafana Labs, 2022b). Wie in unserer Lösung, müsste der Nutzer für jeden Art von Logdateien einen spezifischen "job" konfigurieren.

Inhalt von Logdateien lassen sich auch mit **Application Programming Interface (API)** zum Grafana Loki schicken. In dieser Situation schickt der Endpoint einen HTTP POST Anfrage zum Endpoint von Grafana Loki mit den Inhalt der Logdateien (Grafana Labs, 2020b):

```
# Endpoint
POST [Addresse_von_Grafana_Loki_Instanz]/loki/api/v1/push

# Inhalt
{
  "streams": [
    {
      "stream": {
        "label": "value"
      },
      "values": [
          [ "Zeit in Unixformat", "<Inhalt der Logdateie>" ],
      ]
    }
  ]
}
```

Grafana Loki ist auch mit dem Open Source Tool OpenTelemetry integriert, um Logdateien zu empfangen (Grafana Labs, 2022c). Im Allgemein wird OpenTelemetry dazu verwenden, Daten zu schicken, zu bearbeiten und zu empfangen. Laut dem Anbieter ist OpenTelemetry mit verschiedenen anderen integriert, um die Datenübertragung zu ermöglichen. Das Tool hat *Agents* und *Collectors*. Der erste wird an jedem Endpoint installiert, um die Daten zu schicken und der zweite empfängt die Daten, um diese dann weiterzuleiten (Grafana Labs, 2022c). Die Integration mit Grafana Loki findet mit der Nutzung von API statt. Der *collector* läuft in derselben Umgebung wie Grafana Loki, damit er die Logdateien schicken kann, die *Agents* läufen in jeden Endpoint und kommunizieren sich mit dem *collector*. Die folgende Abbildung soll den beschriebenen Vorgang besser darstellen:

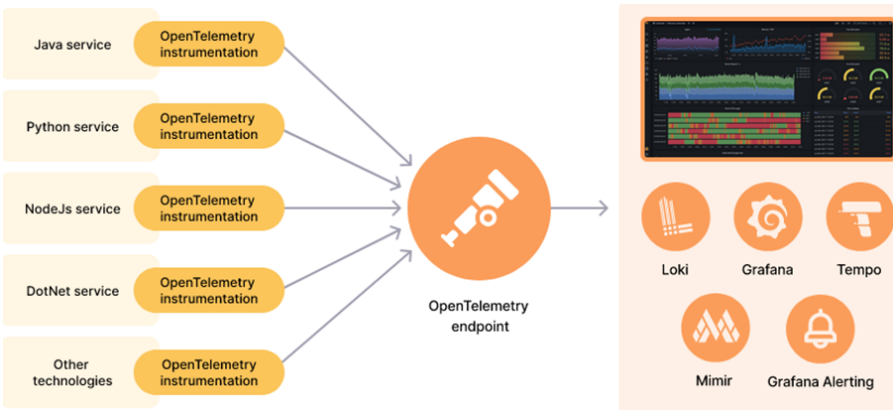

Abbildung 25: Datenfluss zwischen OpenTelemetry und Grafana Loki
Quelle: (Grafana Labs, 2021d)

An der linken Seite haben wir die verschiedenen Endpoints, in dem einen *agent* läuft. In der Mitte haben wir den *Collector*, der die Logdateien schließlich zu Grafana Loki und/oder zu anderen Tools weiterleitet.

3.4. Aufbau der Erkennungsregel für den ausgewählten Angriff

Der Brute-Force Angriffe lässt sich durch die Anzahl des fehlgeschlagen Anmeldungsversuchs erkennen (Selvaganesh et al., 2022). Wir bearbeiten eine Situation, in der es keine Gegenmaßnahmen, wie Kontosperre nach n beliebigen Versuchen oder MFA, implementiert sind. Das folgende Aktivitätsdiagramm stellt einen allgemeinen Ablauf eines Anmeldungsverfahrens dar:

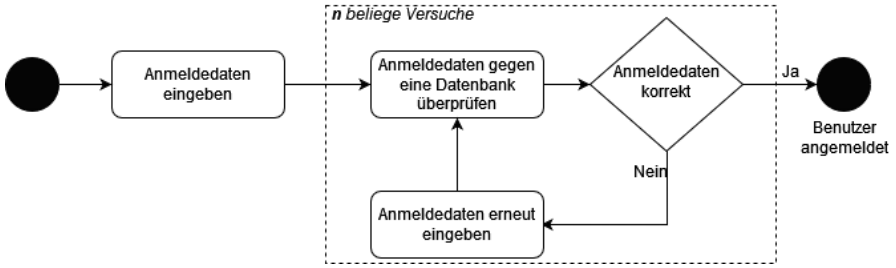

Abbildung 26: Allgemeiner Ablauf eines Anmeldungsverfahrens
Quelle: Eigene Quelle und (Selvaganesh et al., 2022)

Eine Erkennungsregel hätte folgende Logik:

```
# Gefundenen Wert in den Logdateien
# Av = Anzahl fehlgeschlagenen Anmeldungsversuche
# Ia = Intervallzeit zwischen fehlgeschlagenen Anmeldungsversuche

# Festgestellte Werte für legitime und bösartige Verbindungen
# Ga = Grenze zwischen legitimen und bösartigen Anmeldungsversuchen
# Nt = Intervallzeit zwischer legitime Anmeldungsversuche

wenn (Av >= Ga) und (Ia < Nt)
   Warnmeldung(Brutefoce)
sonst
   weiterBeobachten()
```

Grafana Loki bietet einen Einstellungsmuster für die Eingabe und Darstellung von SSH Logdateien an. In dieser Einstellung befinden sich schon Graphik und Regelsätze für eine umfangreiche Analyse dieser Daten (VoidQuark, 2022). Die extrahierte Logdateien werden mit folgenden Elementen gelesen und bearbeitet:

Element	Beschreibung
json	Lesbare Dateiformat, deren Daten nach dem Regel *Schlüssel:Wert* gespeichert sind
Muster	Lesen und Extraktion der Information der Logdateien
Regex	Mustererkennung aus der Logdatei
Logfmt	Extraktion von Schlüssel:Wert Paar der Logdateien

Tabelle 1: Aufbau der Regelsätze in Grafana Loki für SSH Logdateien
Quelle: Eigene Quelle, (VoidQuark, 2022) und (Setter, 2015)

Jedes Angriffszenario hat spezifische Regeln, die mit LogQL aufgebaut sind. In Promtail wird jeder Endpoint "Instance" genannt. Eine oder mehrere "Instance" werden einem "Job" zugewiesen. Diese Struktur kommt aus dem Tool Prometheus. Die "Instances" in einem "Job" werden nach dem gleichen Regeln bearbeitet. Die Abfrage für unseren Angriff sieht so aus (VoidQuark, 2022):

```
(1) - "sum by (username) (count_over_time({$label_name=~\"$label_value\",
      job=~\"$job\", instance=~\"$instance\"}
(2) - |=\"sshd[\"
(3) - |=\": Failed\" !~\"invalid user\"
(4) - | pattern ‘<_> for <username> from <_> port‘
(5) - | __error__=\"\" [$__interval]))",

(1) - Aufsummierung der Benutzername, den dieser Regel entsprechen und
      Filtern nach dem Job und Instance
(2) - Suche nach Zeilen mit dem Wort "sshd"
(3) - Suche nach Zeilen mit dem Wort "Failed" und ohne den Ausdruck
      "invalid user"
(4) - Extrahierung des Benutzernames und der Port der Zeilen
(5) - Suche nach andere Fehlermeldung, falls vorhanden
```

Nachdem die SSH-Logdateien gelesen und bearbeiten wurden, bekommen wir von Grafana Loki folgende Zasammenfasung der Ergebnissen:

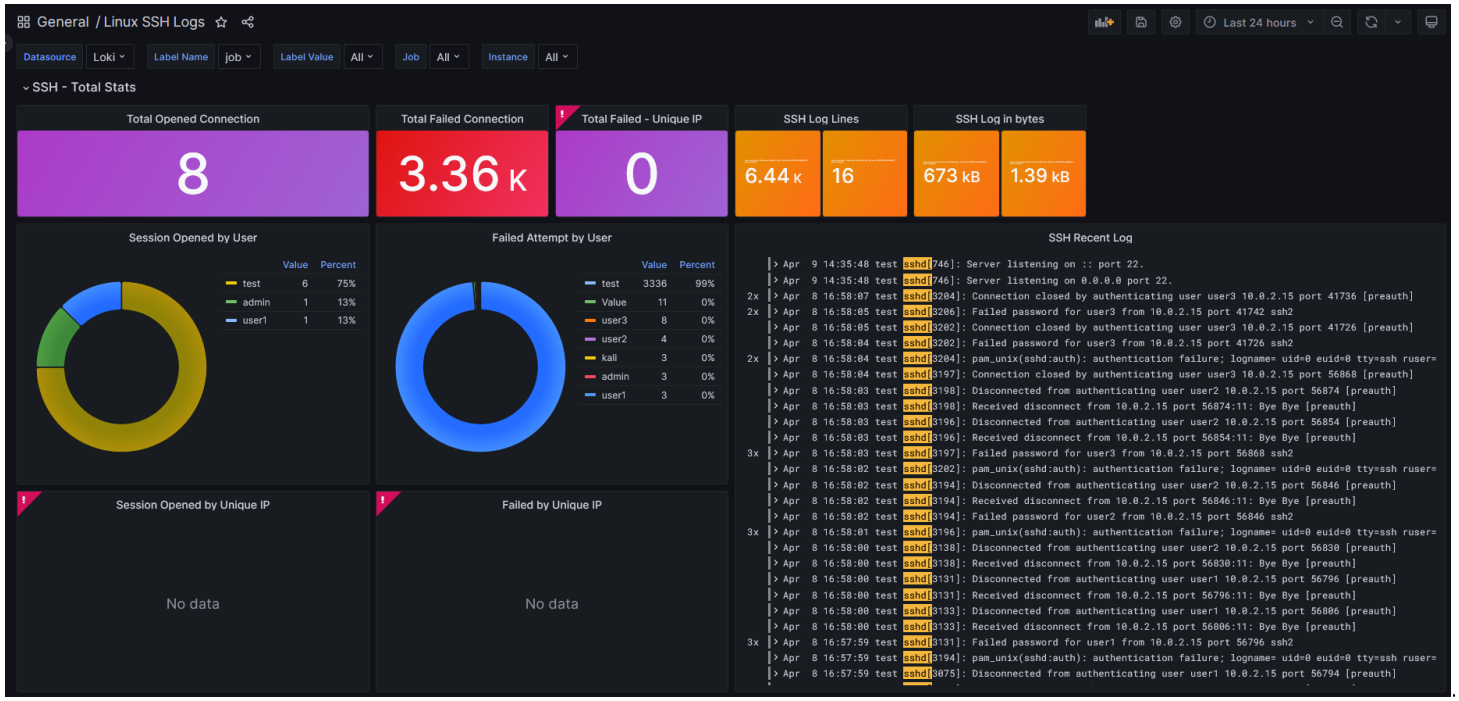

Abbildung 27: Bearbeitung der SSH Logdateien von Grafana Loki
Quelle: Eigene Quelle and (VoidQuark, 2022)

Das nächste Bild gibt ausführliche Informationen der Logdateien:

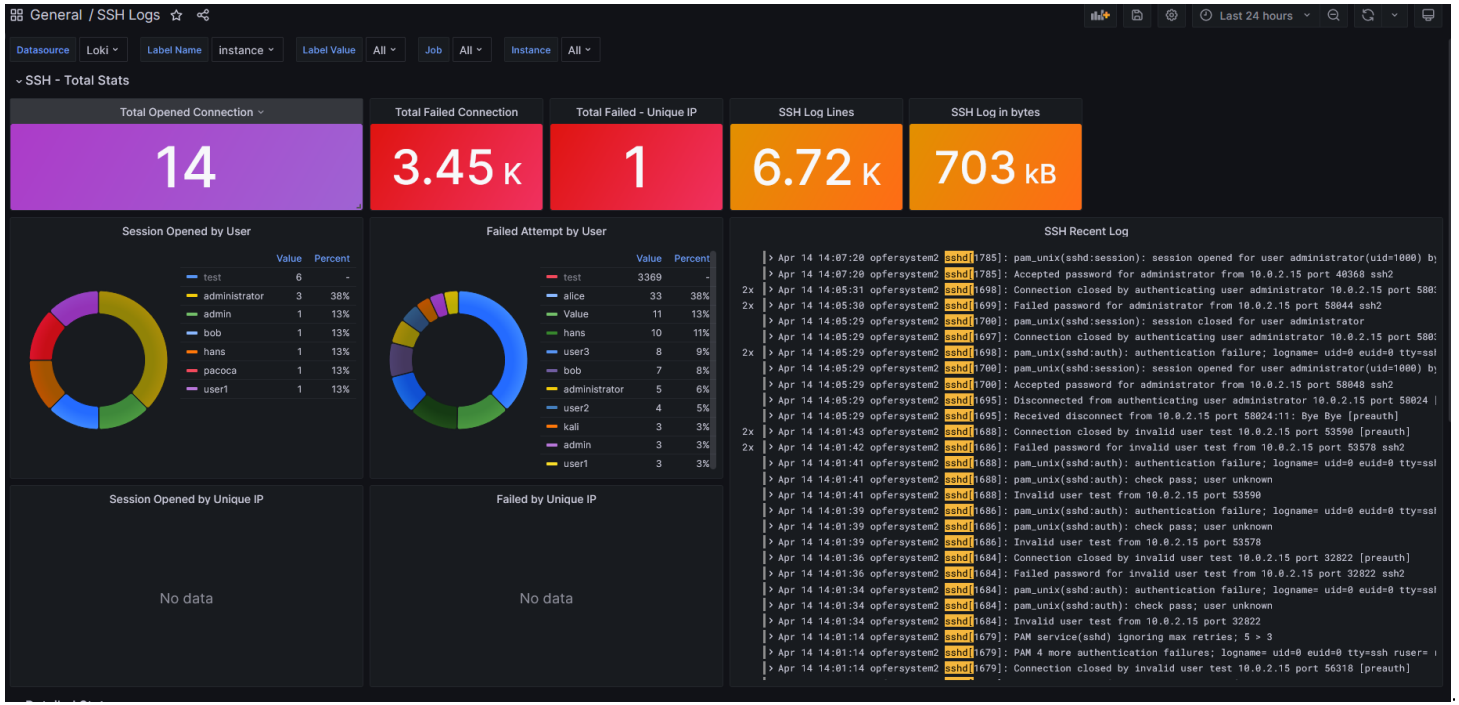

Abbildung 28: Ausführliche Darstellung der SSH Logdateien von Grafana Loki
Quelle: Eigene Quelle and (VoidQuark, 2022)

3.5. Einrichtung des Warnmeldungskomponent

In den vorherigen Teilen dieser Arbeit setzten wir uns damit auseinander, Grafana so einzustellen, damit wir schließlich eine SIEM ähnliche Lösung bekommen könnnten. Von unserer ursprünglichen Vorschlag erreichten wir schon Folgendes:

1. Sammlung der Logdateien aus den Endpoints mit Promtail
2. Anpassung der Logdateien mit Loki für die nachträgliche visuelle Darstellung
3. Nutzung von Regelsätzen in Loki für die Analysierung der SSH Logdateien
4. Graphische Darstellung der Logdateien in Grafana mit den in Loki verwendeten Regelsätzen

Unser letztes Ziel ist es, Warnmeldungen für potenzielle Angriffe mit den Ergebnissen von Loki zu generieren. Grafana lässt sich mit internen und externen Tools integrieren, um Warnmeldungen zu generieren (Grafana Labs, 2021a). Für diese Arbeit benutzen wir das integrierten Tool AlertManager. AlertManager benötigt keine externe Installation und kann nach vordefinierten Regel Warnmeldungen zu beliebigen Endpoints schicken. Diese Regel wird so aufgebaut:

```
(1)  groups:
      - name: example
(2)     rules:
(2.1)    - alert: HighRequestLatency
(2.2)      expr: job:request_latency_seconds:mean5m{job="myjob"} > 0.5
(2.3)      for: 10m
           labels:
             severity: page
           annotations:
             summary: High request latency


(1)   Warnmeldungen können in beliebigen Gruppen kategorisiert werden. Die können so defi
(2)   Ab diesem Punkt definieren wir die Regelsätzen für die Erkennung der Warnmeldung
(2.1) Titel für den Alert, z.B. Pontezieller Angriff gegen ssh
(2.2) Logql Regelsätze für die Erkennung der Warnmeldung
Titel der Alert, z.B.
(2.3) frei definierbare Metaden über die Warnmeldungen
```

AlertManager

mit dem Tool AlertManager integrieren. Mit diesem Tool

Aus der Gra

Über Alarmsignal zu schreiben

4. Bewertung der Daten in Grafana

Zuammenfassung von

- Zielen
- Ergenissen
- Herausforderungen

4.1. Zukünftige Entwicklungen

- Dynamische Regel
- Maschine Learning
- Grafana Nutzung einschränken

Literaturverzeichnis

Advani, S., Mridul, M., Vij, P. S. R., Agarwal, M., and A., L. P. (2020). Iot data analytics pipeline using elastic stack and kafka. *International Journal of Computer Sciences and Engineering*, 8:144–148.
https://www.ijarcce.com/upload/2016/april-16/IJARCCE%2013.pdf. Zugriff am 07.03.2023.

at (2022). Abfragesprache.
https://www.alexanderthamm.com/de/data-science-glossar/abfragesprache/. Zugriff am 08.04.2023.

AT&T Cybersecurity (2022). Alienvault ossim.
https://cybersecurity.att.com/products/ossim. Zugriff am 05.03.2023.

Ba, M. H. N., Bennett, J., Gallagher, M., and Bhunia, S. (2021). A case study of credential stuffing attack: Canva data breach. In *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 735–740.
https://doi.org/10.1109/CSCI54926.2021.00187. Zugriff am 26.03.2023.

BSI (2021). Zweites Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme (IT-Sicherheitsgesetz 2.0).
https://www.bsi.bund.de/DE/Das-BSI/Auftrag/Gesetze-und-Verordnungen/IT-SiG/2-0/it_sig-2-0_node.html. Zugriff am 04.03.2023.

CBNINSIGHTS (2020). Alienvault.
https://www.cbinsights.com/company/alienvault. Zugriff am 05.03.2023.

Centers for Disease Control and Prevention (2016). Health Insurance Portability and Accountability Act of 1996 (HIPAA).
https://www.pcicomplianceguide.org/faq/. Zugriff am 04.03.2023.

Chai, W. and Ferguson, K. (2021). What is HTTP?
https://www.techtarget.com/whatis/definition/HTTP-Hypertext-Transfer-Protocolk. Zugriff am 17.04.2023.

comparitech (2023). The Best SIEM Tools for 2023 Vendors & Solutions Ranked.
https://www.comparitech.com/net-admin/siem-tools/. Zugriff am 05.03.2023.

Dorigo, S. (2012). Security Information and Event Management. Master's thesis, Radboud University Nijmegen.
https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiNu-XkhsD9AhV4FzQIHdMkBWYQFnoECCYQAQ&url=https%3A%2F%2Fwww.ru.nl%2Fpublish%2Fpages%2F769526%2Fthesissanderdorigo.pdf&usg=AOvVaw3oPn4KBFwgJwexoXZ1Be40. Zugriff am 03.03.2023.

Douglis, F. and Nieh, J. (2019). Microservices and containers. *IEEE Internet Computing*, 23(6):5–6.
https://doi.org/10.1109/MIC.2019.2955784. Zugriff am 236.03.2023.

elastic (2021). *FAQ on 2021 License Change*.

https://www.elastic.co/pricing/faq/licensing. Zugriff am 26.03.2023.

elastic (2022). *Elastic Docs*.
https://www.elastic.co/guide/en/welcome-to-elastic/current/new.html.
Zugriff am 5.02.2023.

European Comission (2015). Siem design and development.
https://cordis.europa.eu/project/id/644425. Zugriff am 05.03.2023.

Fortinet (2016). Fortinet Announces Acquisition of AccelOps .
https://www.fortinet.com/corporate/about-us/newsroom/press-releases/2016/fortinet-announces-acquisition-of-accelops. Zugriff am 06.03.2023.

Fortinet (2020). FortiSIEM Reference Architecture.
https://www.fortinet.com/content/dam/maindam/PUBLIC/02_MARKETING/02_Collateral/DeploymentGuide/dg-fortisiem-reference-architecture.pdf. Zugriff am 06.03.2023.

Fortinet (2022). FortiSIEM Solutions.
https://www.fortinet.com/products/siem/fortisiem. Zugriff am 06.03.2023.

Fu, F. (2018). Chapter six - design and analysis of complex structures. In *Design and Analysis of Tall and Complex Structures*, pages 177–211. Butterworth-Heinemann.
https://www.sciencedirect.com/science/article/pii/B978008101018100006X.
Zugriff am 06.03.2023.

Grafana Labs (2016). Dashboard anything. Observe everything.
https://grafana.com/grafana/. Zugriff am 12.03.2023.

Grafana Labs (2018). Grafana Loki.
https://grafana.com/oss/loki/. Zugriff am 08.04.2023.

Grafana Labs (2020a). Getting started.
https://grafana.com/docs/loki/latest/getting-started/. Zugriff am 09.04.2023.

Grafana Labs (2020b). Grafana Loki HTTP API.
https://grafana.com/docs/loki/latest/api/. Zugriff am 17.04.2023.

Grafana Labs (2021a). Alertmanager.
https://grafana.com/docs/grafana/latest/alerting/manage-notifications/alertmanager/. Zugriff am 21.04.2023.

Grafana Labs (2021b). Collect logs with Grafana Agent.
https://grafana.com/docs/grafana-cloud/data-configuration/logs/collect-logs-with-agent/. Zugriff am 17.04.2023.

Grafana Labs (2021c). LogQL: Log query language.
https://grafana.com/docs/loki/latest/logql/. Zugriff am 14.04.2023.

Grafana Labs (2021d). What is opentelemetry?
https://grafana.com/oss/opentelemetry/. Zugriff am 17.04.2023.

Grafana Labs (2022a). Dashboard anything. Observe everything.

https://grafana.com/logs/. Zugriff am 12.03.2023.

Grafana Labs (2022b). Grafana Agent.
https://grafana.com/docs/agent/latest/. Zugriff am 17.04.2023.

Grafana Labs (2022c). How to send logs to grafana loki with the opentelemetry collector using fluent forward and filelog receivers.
https://grafana.com/blog/2022/06/23/how-to-send-logs-to-grafana-loki-with-the-opentelemetry-collector-using-fluent-forward-and-filelog-receivers/. Zugriff am 17.04.2023.

Granadillo, G., González-Zarzosa, S., and Diaz, R. (2021). Security information and event management (siem): Analysis, trends, and usage in critical infrastructures. *Sensors*, 21:4759.
file:///C:/Users/bruno/Downloads/Security_Information_and_Event_Management_SIEM_Ana.pdf. Zugriff am 21.02.2023.

Gómez, E. C. F., Almeida, O. X. B., and Gamboa, L. M. A. (2022). Analysis of centralized computer security systems through the alienvault ossim tool. *Ecuadorian Science Journal*, 6(1):23–31.
https://journals.gdeon.org/index.php/esj/article/view/181. Zugriff am 03.03.2023.

Hazel, T. (2021). How To Use the MITRE ATT&CK Framework.
https://www.chaossearch.io/blog/how-to-use-mitre-attck-framework. Zugriff am 26.03.2023.

IBM (2020). What is an api (application programming interface)?
https://www.ibm.com/topics/api. Zugriff am 17.04.2023.

Ibrokhimov, S., Hui, K. L., Al-Absi, A. A., hoon jae lee, and Sain, M. (2019). Multi-Factor Authentication in Cyber Physical System: A State of Art Survey. In *2019 21st International Conference on Advanced Communication Technology (ICACT)*, pages 279–284. https://doi.org/10.23919/ICACT.2019.8701960, Zugriff am 26.03.2023.

IT-Service.Network (2020). Was ist ein plug-in?
https://www.hhs.gov/hipaa/for-professionals/privacy/index.html. Zugriff am 04.03.2023.

Jain, U. (2018). *Lateral Movement Detection Using ELK Stack*. PhD thesis, University of Houston.
https://uh-ir.tdl.org/handle/10657/3109. Zugriff am 07.03.2023.

Janiesch, C., Zschech, P., and Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31(3):685–695.
https://doi.org/10.1007/s12525-021-00475-2. Zugriff am 13.03.2023.

Jog, Y. (2020). Security Information and Event Management (SIEM).
https://www.linkedin.com/pulse/security-information-event-management-siem-yatin-jog. Zugriff am 04.03.2023.

Kali (2019). Kali inside virtualbox (guest vm).

https://www.kali.org/docs/virtualization/install-virtualbox-guest-vm/. Zugriff am 02.04.2023.

Kali (2022a). Hydra.
https://www.kali.org/tools/hydra/. Zugriff am 02.04.2023.

Kali (2022b). What is kali linux & kalis features.
https://www.kali.org/docs/introduction/. Zugriff am 02.04.2023.

Kazarov, A., Avolio, G., Chitan, A., and Mineev, M. (2018). Experience with splunk for archiving and visualisation of operational data in atlas tdaq system. *Journal of Physics: Conference Series*, 1085:032052.
http://dx.doi.org/10.1088/1742-6596/1085/3/032052. Zugriff am 04.03.2023.

Manases, L. and Zinca, D. (2022). Automation of network traffic monitoring using docker images of snort3, grafana and a custom api. In *2022 21st RoEduNet Conference: Networking in Education and Research (RoEduNet)*, pages 1–4.
https://doi.org/10.1109/RoEduNet57163.2022.9921063. Zugriff am 13.03.2023.

Martin, L. (2018). The cyber kill chain.
https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html. Zugriff am 12.03.2023.

Microsoft Security (2022). Endpoints defined.
https://www.microsoft.com/en-us/security/business/security-101/what-is-an-endpoint. Zugriff am 12.03.2023.

Mikalauskas, E. (2023). Rockyou2021: largest password compilation of all time leaked online with 8.4 billion entries.
https://cybernews.com/security/rockyou2021-alltime-largest-password-compilation-leaked/. Zugriff am 02.04.2023.

Miller, J. (2021). is elastic stack (elk) the best siem option?
https://www.bitlyft.com/resources/is-elk-the-best-siem-option#:~:text=The%20ELK%20stack%20is%20a,system%20from%20a%20system%20provider. Zugriff am 07.03.2023.

MITRE ATT&CK (2018a). Frequently Asked Questions.
https://attack.mitre.org/resources/faq/. Zugriff am 12.03.2023.

MITRE ATT&CK (2018b). Getting Started.
https://attack.mitre.org/resources/getting-started/. Zugriff am 26.03.2023.

MITRE ATT&CK (2020). Brute Force.
https://attack.mitre.org/techniques/T1110/. Zugriff am 26.03.2023.

Mohammed, S. A., Mohammed, A. R., Côté, D., and Shirmohammadi, S. (2021). A machine-learning-based action recommender for network operation centers. *IEEE Transactions on Network and Service Management*, 18(3):2702–2713.
https://doi.org/10.1109/TNSM.2021.3095463. Zugriff am 20.02.2023.

Mohanan, R. (2022). What is security information and event management (siem)? definition, architecture, operational process, and best practices.

https://www.spiceworks.com/it-security/vulnerability-management/articles/what-is-siem/. Zugriff am 26.02.2023.

Nabil, M., Soukainat, S., Lakbabi, A., and Ghizlane, O. (2017). Siem selection criteria for an efficient contextual security. In *2017 International Symposium on Networks, Computers and Communications (ISNCC)*, pages 1–6. https://doi.org/10.1109/ISNCC.2017.8072035. Zugriff am 26.02.2023.

neptune (2023). A Machine Learning Approach to Log Analytics: How to Analyze Logs? .

https://www.nexcess.net/blog/open-source-vs-proprietary/. Zugriff am 12.03.2023.

Nexcess (2022). Open source vs. proprietary: Which is better? https://www.nexcess.net/blog/open-source-vs-proprietary/. Zugriff am 26.02.2023.

NIST (2020a). About nist. https://www.nist.gov/about-nist. Zugriff am 19.02.2023.

NIST (2020b). Glossary. https://csrc.nist.gov/glossary/. Zugriff am 19.02.2023.

Open Source Initiative (2007). The Open Source Definition (Annotated). https://opensource.org/definition/. Zugriff am 17.02.2023.

packt (2019). What is elk stack? https://subscription.packtpub.com/book/big-data-and-business-intelligence/9781788831031/1/ch01lvl1sec10/what-is-elk-stack. Zugriff am 07.03.2023.

Polinowski, M. (2019). What is elk stack? https://mpolinowski.github.io/docs/DevOps/Provisioning/2021-04-07--loki-prometheus-grafana/2021-04-07/. Zugriff am 09.04.2023.

Prelude SIEM (2018). Prelude SIEM: Smart Security. https://www.prelude-siem.com/en/prelude-siem-en/. Zugriff am 05.03.2023.

Prelude SIEM (2020). *Prelude Documentation: version 5.2.* https://www.prelude-siem.org/docs/5.2/en/. Zugriff am 06.03.2023.

Prelude Team (2007). *Manual User.* https://www.prelude-siem.org/projects/prelude/wiki/. Zugriff am 06.03.2023.

Prometheus (2016). Documentation. https://prometheus.io/docs/introduction/overview/. Zugriff am 14.04.2023.

Radoglou-Grammatikis, P., Sarigiannidis, P., Iturbe, E., Rios, E., Martinez, S., Sarigiannidis, A., Eftathopoulos, G., Spyridis, Y., Sesis, A., Vakakis, N., Tzovaras, D., Kafetzakis, E., Giannoulakis, I., Tzifas, M., Giannakoulias, A., Angelopoulos, M., and Ramos, F. (2021). Spear siem: A security information and event management system for the smart grid. *Computer Networks*, 193:108008. https://doi.org/10.1016/j.comnet.2021.108008. Zugriff am 03.03.2023.

Ramírez Tomás, I. (2018). *Implementación de un sistema de gestión de eventos de seguridad en una empresa de tamaño medio.* PhD thesis, Universitat Politècnica de València.
https://riunet.upv.es/bitstream/handle/10251/109765/Ram%c3%adrez%20-%20Implementaci%c3%b3n%20de%20un%20sistema%20de%20gesti%c3%b3n%20de%20eventos%20de%20seguridad%20en%20una%20empresa%20de%20tama%c3%b1....pdf?sequence=1&isAllowed=y. Zugriff am 06.03.2023.

redhat (2022). What is grafana?
https://www.redhat.com/en/topics/data-services/what-is-grafana. Zugriff am 13.03.2023.

Roser, M., Ritchie, H., and Ortiz-Ospina, E. (2015). Internet. *Our World in Data.*
https://ourworldindata.org/internet. Zugriff am 17.02.2023.

Savic, D., da Silva, A. R., Vlajic, S., Lazarevic, S., Stanojevic, V., Antovic, I., and Milic, M. (2012). Use case specification at different levels of abstraction. In *2012 Eighth International Conference on the Quality of Information and Communications Technology*, pages 187–192.
https://doi.org/10.1109/QUATIC.2012.64. Zugriff am 12.03.2023.

Selvaganesh, M., Karthi, P., Kumar, V. A. N., and Moorthy, S. R. P. (2022). Efficient brute-force handling methodology using indexed-cluster architecture of splunk. In *2022 International Conference on Electronics and Renewable Systems (ICEARS)*, pages 697–701.
https://doi.org/10.1109/ICEARS53579.2022.9752323. Zugriff am 12.03.2023.

Setter, M. (2015). Logfmt: A Log Format That's Easy To Read and Write.
https://www.cloudbees.com/blog/logfmt-a-log-format-thats-easy-to-read-and-write. Zugriff am 10.04.2023.

silicon.de (2022). Das beliebteste deutsche Passwort 2022 lautet: 123456.
https://www.silicon.de/41703603/das-beliebteste-deutsche-passwort-2022-lautet-123456. Zugriff am 02.04.2023.

Sowmya, G. V., Jamuna, D., and Reddy, M. V. K. (2012). Blocking of Brute Force Attack. *International journal of engineering research and technology*, 1.

Splunk (2015). Splunk Enterprise Security.
https://www.splunk.com/en_us/products/enterprise-security.html. Zugriff am 12.03.2023.

Splunk (2022a). Use Cases.
https://docs.splunk.com/Documentation/ES/7.1.0/Usecases/Overview. Zugriff am 12.03.2023.

Splunk (2022b). What Is Security Information and Event Management (SIEM)?
https://www.splunk.com/en_us/data-insider/what-is-siem.html. Zugriff am 12.03.2023.

Su, T.-J., Wang, S.-M., Chen, Y.-F., and Liu, C.-L. (2016). Attack detection of distribu-

ted denial of service based on splunk. In *2016 International Conference on Advanced Materials for Science and Engineering (ICAMSE)*, pages 397–400. https://doi.org/10.1109/ICAMSE.2016.7840355. Zugriff am 12.03.2023.

Swathi, K. (2022). Brute Force Attack on Real World Passwords. *International Journal of Research Publication and Reviews*, 3(11):552–558. https://www.ijrpr.com/archive.php?volume=3&issue=11. Zugriff am 26.02.2023.

Tanembaum, A. S. (2009). *Moderne Betriebssysteme*. Pearson, München.

techopedia (2015). Security Event Management. https://www.techopedia.com/definition/25763/security-event-management. Zugriff am 03.03.2023.

techopedia (2022). Security Information Management (SIM). https://www.techopedia.com/definition/25763/security-event-management. Zugriff am 03.03.2023.

Tek-Tools (2020). Log Analysis – How to Use a Log Analyzer Tool? https://www.tek-tools.com/apm/choosing-log-analyzer-tool. Zugriff am 12.03.2023.

tutorialspoint (2009). HTTP - Methods. https://www.tutorialspoint.com/http/http_methods.htm. Zugriff am 17.04.2023.

Ubuntu (2023a). Get Ubuntu Server. https://ubuntu.com/download/server. Zugriff am 31.03.2023.

Ubuntu (2023b). Ubuntu. https://ubuntu.com/. Zugriff am 31.03.2023.

U.S. Department of Health & Human Services (2016). The HIPAA Privacy Rule. https://www.hhs.gov/hipaa/for-professionals/privacy/index.html. Zugriff am 04.03.2023.

Vielberth, M. (2021). *Encyclopedia of Cryptography, Security and Privacy*, chapter Security Operations Center (SOC), pages 1–3. Springer Berlin Heidelberg. http://dx.doi.org/10.1007/978-3-642-27739-9_1680-1. Zugriff am 04.03.2023.

VoidQuark (2022). Parsing SSH Logs with Grafana Loki. https://voidquark.com/parsing-ssh-logs-with-grafana-loki/. Zugriff am 10.04.2023.

Wang, Y.-T., Yang, C.-T., Kristiani, E., and Chan, Y.-W. (2019). The implementation of wi-fi log analysis system with elk stack. In *Frontier Computing*, pages 246–255, Singapore. Springer Singapore. https://link.springer.com/chapter/10.1007/978-981-13-3648-5_28. Zugriff am 07.03.2023.

Wendzel, S. (2018). *IT-Sicherheit für TCP/IP- und IoT-Netzwerke*. Springer Vieweg, Wiesbaden.

Ödegaard, T. (2019). The (Mostly) Complete History of Grafana UX.

https://grafana.com/blog/2019/09/03/the-mostly-complete-history-of-grafana-ux/. Zugriff am 13.03.2023.

Łukasz Korzeniowski and Goczyła, K. (2022). Landscape of automated log analysis: A systematic literature review and mapping study. *IEEE Access*, 10:21892–21913. https://doi.org/10.1109/ACCESS.2022.3152549. Zugriff am 12.03.2023.

A. Originalle Einstellungsdateien

Unten befindet sich die originalle Konfigurationsdateien (Grafana Labs, 2020a):

- **Grafana Loki** für die Speicherung und Bearbeitung der Logdateien

```
auth_enabled: false

server:
  http_listen_port: 3100
  grpc_listen_port: 9096

common:
  instance_addr: 127.0.0.1
  path_prefix: /tmp/loki
  storage:
    filesystem:
      chunks_directory: /tmp/loki/chunks
      rules_directory: /tmp/loki/rules
  replication_factor: 1
  ring:
    kvstore:
      store: inmemory

query_range:
  results_cache:
    cache:
      embedded_cache:
        enabled: true
        max_size_mb: 100

schema_config:
  configs:
    - from: 2020-10-24
      store: boltdb-shipper
      object_store: filesystem
      schema: v11
      index:
        prefix: index_
        period: 24h

ruler:
  alertmanager_url: http://localhost:9093

# By default, Loki will send anonymous, but uniquely-identifiable
usage and configuration
# analytics to Grafana Labs. These statistics are sent to
# https://stats.grafana.org/

# Statistics help us better understand how Loki is used, and they
# show us performance levels for most users. This helps us
#  prioritize features and documentation.

# For more information on what's sent, look at
# https://github.com/grafana/loki/blob/main/pkg/usagestats/stats.go
# Refer to the buildReport method to see what goes into a report.

# If you would like to disable reporting, uncomment the following
# lines analytics:
#  reporting_enabled: false
```

- **Promtail** für die Sammlung der Logdateien

```
  server:
  http_listen_port: 9080
  grpc_listen_port: 0

positions:
  filename: /tmp/positions.yaml

clients:
  - url: http://loki:3100/loki/api/v1/push

scrape_configs:
- job_name: system
  static_configs:
  - targets:
      - localhost
    labels:
      job: varlogs
      __path__: /var/log/*log
```

B. Angepasste Einstellungsdateien von Grafana

Unten befindet sich die angepasste Konfigurationsdateien (Polinowski, 2019):

- **Grafana Loki**

```
auth_enabled: false

server:
  http_listen_port: 3100
  grpc_listen_port: 9096

ingester:
  wal:
    enabled: true
    dir: /tmp/wal
  lifecycler:
    address: 127.0.0.1
    ring:
      kvstore:
        store: inmemory
      replication_factor: 1
    final_sleep: 0s
  # Any chunk not receiving new logs in this time will be flushed
  chunk_idle_period: 1h
  # All chunks will be flushed when they hit this age, default is 1h
  max_chunk_age: 1h
  # Loki will attempt to build chunks up to 1.5MB, flushing first if
  # chunk_idle_period or max_chunk_age is reached first
  chunk_target_size: 1048576
  # Must be greater than index read cache TTL if using an index cache
  #(Default index read cache TTL is 5m)
  chunk_retain_period: 30s
  # Chunk transfers disabled
  max_transfer_retries: 0

schema_config:
  configs:
    - from: 2020-10-24
      store: boltdb-shipper
      object_store: filesystem
      schema: v11
      index:
        prefix: index_
        period: 24h

storage_config:
  boltdb_shipper:
    active_index_directory: /tmp/loki/boltdb-shipper-active
    cache_location: /tmp/loki/boltdb-shipper-cache
    # Can be increased for faster performance over longer query
    # periods, uses more disk space
    cache_ttl: 24h
    shared_store: filesystem
  filesystem:
    directory: /tmp/loki/chunks

compactor:
  working_directory: /tmp/loki/boltdb-shipper-compactor
  shared_store: filesystem

limits_config:
  reject_old_samples: true
```

```
  reject_old_samples_max_age: 168h

chunk_store_config:
  max_look_back_period: 0s

table_manager:
  retention_deletes_enabled: false
  retention_period: 0s

ruler:
  storage:
    type: local
    local:
      directory: /tmp/loki/rules
  rule_path: /loki/rules-temp
  alertmanager_url: http://localhost:9093
  ring:
    kvstore:
      store: inmemory
  enable_api: true
```

- **Promtail**

```
---
server:
  http_listen_port: 9080
  grpc_listen_port: 0

positions:
  filename: /tmp/positions.yaml

clients:
  - url: http://loki:3100/loki/api/v1/push
    tenant_id: tenant1

scrape_configs:
- job_name: Opfersystem
  static_configs:
  - targets:
      - loki
    labels:
      instance: OpferSystem
      env: Variable
      job: varlogs
      __path__: /opt/*.log
```

- **Docker Compose Datei**

```
version: "3"

networks:
  loki:

services:
  loki:
    image: grafana/loki:2.3.0
    volumes:
      - <lokales_Verzeichnis>/loki-config.yaml:/etc/loki
        /loki-config.yaml
    ports:
      - "3100:3100"
    command: -config.file=/etc/loki/local-config.yaml
    networks:
      - loki

  promtail:
    image: grafana/promtail:2.3.0
    volumes:
      - <lokales_Verzeichnis>/promtail-config.yaml
      - <lokales_Verzeichnis>/ssh1.log:/opt/ssh1.log
      - <lokales_Verzeichnis>/ssh2.log:/opt/ssh2.log
    command: -config.file=/etc/promtail/promtail-config.yaml
    networks:
      - loki

  grafana:
    image: grafana/grafana:latest
    ports:
      - "3000:3000"
    networks:
      - loki
```