h_da
HOCHSCHULE DARMSTADT
UNIVERSITY OF APPLIED SCIENCES
fbi
FACHBEREICH INFORMATIK

Hochschule Darmstadt
- Fachbereich Informatik -

Analyse sicherheitsrelevanter Informationen in industriellen Netzwerken für den Einsatz eines SIEM Systems

Abschlussarbeit zur Erlangung des akademischen Grades
Master of Science (M.Sc.)

vorgelegt von

Niklas Breuer
727905

Referent:	Prof. Dr. Oliver Weissmann
Korreferent:	Jürgen Zorenc

Erklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die im Literaturverzeichnis angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder noch nicht veröffentlichten Quellen entnommen sind, sind als solche kenntlich gemacht. Die Zeichnungen oder Abbildungen in dieser Arbeit sind von mir selbst erstellt worden oder mit einem entsprechenden Quellennachweis versehen. Die Arbeit ist in gleicher oder ähnlicher Form noch bei keiner anderen Prüfungsbehörde eingereicht worden.

Darmstadt den 2. August 2018

Niklas Breuer

Abtract

Die vorliegende Masterarbeit beschreibt die Analyse des Modells eines industriellen Netzwerkes, um potentielle Lücken in der Informationsmenge zu finden, welche notwendig sind um eine Überwachung des Netzwerkes mit Hilfe eines SIEM Systems zu ermöglichen. Dabei wurden Modelle eines Unternehmensnetzwerkes und eines industriellen Netzwerkes auf ihre Informationsquellen analysiert und eine Vergleichsbasis erstellt. Die aus dem Vergleich resultierenden Unterschiede wurden anhand des Stuxnet-Angriffes auf ihre Wertigkeit für die Überwachung industrieller Netzwerke untersucht. Für eine bestimmte Lücke wurde ein Vorschlag für die Generierung der Informationen beschrieben.

This master thesis describes the analysis of the model of an industrial network to find potential gaps in the amount of information necessary to enable monitoring of the network with the help of a SIEM system. Models of a corporate network and an industrial network were analyzed for their sources of information and a comparison basis was created. The differences resulting from the comparison were examined by means of the Stuxnet attack on their value for monitoring industrial networks. For a given gap, a proposal for generating the information has been described.

Todo list

Inhaltsverzeichnis

Kapitel 1

Einleitung

1.1 Kontext

Ein grundlegendes Konzept für die Sicherung eines Netzwerkes ist das Wissen um den Zustand des Netzwerkes und seiner Elemente. In modernen Unternehmensnetzwerken mit tausenden Elementen, Schnittstellen und Abhängigkeiten ist es schwierig, jederzeit den Überblick zu behalten. Aus diesem Grund nutzen Unternehmen *Security Information and Event Management (SIEM)* Systeme, die für die Analyse protokollierter Ereignisse und der Erkennung von potentiellen Bedrohungsszenarien verwendet werden.

Die Industrie 4.0 ist ein Konzept, mit deren Hilfe die nächste Stufe der industriellen Revolution erreicht werden soll. Um dieses Ziel zu erreichen, sollen *Smart Factories* erstellt werden, in denen Anlagen miteinander sowie Produkte mit den entsprechenden Anlagen kommunizieren können. Für dieses Vorhaben besteht ein steigendes Interesse an vertikaler und horizontaler Integration, d.h. der stärkeren Vernetzung der einzelnen Komponenten / Anlagen miteinander sowie der Anbindung industrieller Produktionsnetzwerke an das WAN (Internet). Dabei ist jedoch zu berücksichtigen, dass industrielle Produktionskomponenten, die auch mit dem Begriff „cyber-physisches System“ bezeichnet werden, nicht für die Anbindung an externe Netzwerke entworfen wurden, sondern mit den Zielen Safety (Sicherheit der Mitarbeiter), Verlässlichkeit und Effizienz. Aus diesem Grund sind weder Komponenten noch die verwendeten Kommunikationstechnologien und -protokolle ausreichend gegen Bedrohungen aus dem Internet abgesichert. Dabei ist der Schutz dieser Systeme sogar besonders kritisch zu bewerten, da erfolgreiche digitale Angriffe sich aus der digitalen auf die physische Welt übertragen und materielle und finanzielle Schäden sowie eine Gefahr für die Beschäftigten der Anlagen darstellen können. Allerdings ist der Schutz dieser Systeme auf Grund mehrerer Faktoren problematisch.

Die Zertifizierungsnotwendigkeit für die sichere Ausführung und der teils sehr alten Technologien und -standards ist nicht ohne weiteres möglich, da neben den Anforderungen der IT-Sicherheit für kritische Systeme vor allem auch die Anforderungen der Anlagentechnik gewahrt bleiben müssen. Die verwendeten speicherprogrammierbaren Steuerungen (SPS) sind für die Kontrolle der einzelnen Fertigungsprozessschritte auf eine sichere Funktionalität, jedoch nicht auf Manipulationssicherheit durch externe Quellen ausgerichtet. Basierend auf den Anforderungen innerhalb der industriellen Fertigungsnetzwerke sind Logik und Technologie auf Verlässlichkeit (d.h. Determinismus / Fehlerfreiheit), Reaktionsschnelligkeit und Robustheit gegen äußere physikalische Einflüsse der Anlagenumgebung ausgelegt. Diese Anforderungen und die ursprüngliche Entwicklung der Automation von logischen Steckkombinationen hin zu programmierbaren Steuerungen ergeben knappe Ressourcen im Sinne der Kommunikationsbandbreite, Speicher und Rechenleistung. Programmerweiterungen können durch zusätzliche Programmlogik Fehler und unberechenbares Verhalten des physikalischen Equipments mit sich bringen. Fehlende Standards ergeben zusätzlich eine Netzwerklandschaft, in der viele verschiedene herstellerspezifische Kommunikationsprotokolle, Geräte, Gerätetypen und Kommunikationstechnologien vertreten sind. Zwar werden Versuche und Anstrengungen unternommen Standards zu schaffen, allerdings ist der aktuelle Stand noch nicht soweit.

Diese Faktoren erschweren die Anwendung von typischen Sicherheitselementen wie den erwähnten SIEM Systemen ungemein. Neben der Notwendigkeit der Anpassung an die Anforderungen der Fertigungssysteme behindern die knappen Ressourcen und Limitierungen der Erweiterbarkeit und Aktualisierung der Anlagen die Anwendung der etablierten Sicherheitskonzepte aus den Unternehmensnetzwerken.

1.2 Zielsetzung und Zweck

In dieser Ausarbeitung soll diese Problematik im spezifischen Fall der SIEM Systeme untersucht werden. SIEM Systeme benötigen Informationen der Netzwerkelemente (sowohl der Endgeräte als auch der Netzwerkgeräte und der Kommunikationsmechanismen- und Schnittstellen). Ist die Verfügbarkeit der Informationen limitiert, kann das SIEM System auch nur im Rahmen der verfügbaren Informationen ungewöhnliches Verhalten der Systeme und Bedrohungsszenarien erkennen und die Sicherheitsoperatoren bei ihrer Arbeit unterstützen. Für die Absicherung eines industriellen Netzwerkes im Kontext der Industrie 4.0 ist eine Analyse der verfügbaren Informationsquellen notwendig. Basierend auf der bereits erläuterten Problematik wird jedoch die Hypothese aufgestellt, dass die typischen Komponenten und Technologien eines industri-

ellen Netzwerkes kritische Informationslücken aufweisen, welche die Überwachung und Analyse kritischer Aspekte verhindern. Dies ist die Kernthese dieser Arbeit, die es zu untersuchen gilt. Für die Beantwortung dieser These werden die folgenden Forschungsfragen untersucht:

- Welche Informationen lassen sich jeweils aus Unternehmens- und Industrienetzwerken gewinnen?
- Welche Unterschiede bestehen bzgl. der verfügbaren Informationen im Vergleich zwischen den Unternehmenstypen?
- Sind die gefundenen Unterschiede als kritisch für die Überwachung industrieller Netzwerkes einzustufen?

Der Fokus für die Untersuchung der Unterschiede zwischen den Netzwerken soll auf das Fehlen von Informationskategorien in industriellen Netzwerken gesetzt werden. Die Elemente dieser konkreten potentiellen Untermenge der Unterschiede werden im Rahmen dieser Ausarbeitung als *Informationslücken* bezeichnet.

1.3 Vorgehen

Als Startpunkt für die Suche nach den potentiellen Informationslücken in industriellen Netzwerken wird eine Referenzmenge benötigt. Diese Referenzmenge ergibt sich für SIEM Systeme aus bereits etablierten Implementierungen in Unternehmensnetzwerken. Aus diesem Grund ist der erste Schritt die Analyse der Informationsmenge eines Unternehmensnetzwerkes. Zu diesem Zweck werden zunächst Grundkenntnisse über das SIEM Konzept und dessen Anwendung sowie über Elemente und Technologien in Unternehmensnetzwerken und industriellen Netzwerken zur Verfügung gestellt.

Die erste Herausforderung bei der folgenden Analyse ist die Eingrenzung der Informationsmenge. Durch eine sehr große Anzahl an unterschiedlichen Systemen, Protokollen, Schnittstellen und herstellerspezifischen Zusatzeigenschaften muss die Analyse auf ein handhabbares Modell beschränkt werden. Für die Analyse werden ein repräsentatives Modell definiert und die einzelnen Elemente genauer betrachtet. Ziel der Analyse ist es zum einen, die Informationsquellen, etwa in Form von Protokolldateien, zu identifizieren und diese in Kategorien auf Basis ihres Inhaltes einzuordnen. Dies ermöglicht die Eingrenzung und Fassung der Informationsmenge.

Der gleiche Ansatz wird für die Analyse des industriellen Netzwerkes gewählt. Im Unterschied zu der vorherigen Analyse wird jedoch der Versuch unternommen, die

Informationen den zuvor definierten Kategorien des Unternehmensnetzwerkes zuzuordnen. Basierend auf dieser Zuordnung entsteht eine notwendige Vergleichsbasis, auf der die beiden Netzwerkarten miteinander verglichen werden können.

Die Ergebnisse des Vergleiches müssen jedoch nicht zwangsläufig auf eine kritische Informationslücke schließen lassen. Durch die unterschiedlichen Zwecke und Fähigkeiten der Netzwerkelemente und Kommunikationstechnologien können Unterschiede in der Informationsmenge auch bspw. auf nicht unterstützte Funktionalitäten oder Dienste hinweisen. Um die Bedeutung von Unterschieden bzgl. der Notwendigkeit für die Überwachung zu ermitteln, müssen diese Unterschiede diesbzgl. bewertet werden. Die Bewertung wird anhand des populären Beispiels *Stuxnet* für einen Angriff auf eine industrielle Anlage durchgeführt. Die Analyse der Aktionen des Stuxnet Angriffes erlaubt Rückschlüsse darauf, welche Informationen eines industriellen Netzwerkes notwendig sind, um Angriffe wie diesen zu erkennen. Ein Abgleich der gefunden Unterschiede mit den Analysepunkten des Stuxnetangriffes ergibt die Bedeutung der gefundenen Unterschiede.

Schlussendlich wird auf Basis dieser Erkenntnisse versucht, einen Lösungsansatz für eine konkrete Problemstelle unter bestimmten Annahmen zu formulieren.

1.4 Struktur

Die Struktur der Arbeit orientiert sich direkt an der beschriebenen Vorgehensweise. Kapitel 2 liefert die notwendigen Grundkenntnisse, die für das Verständnis der Untersuchung als notwendig erachtet werden. In den darauf folgenden Kapiteln werden die Analyse der Modelle des Unternehmensnetzwerkes (Kapitel 3) und des industriellen Netzwerkes (Kapitel 4) beschrieben. Die Vergleichsmethodik und die Ergebnisse des Vergleichs finden sich in Kapitel 5. Kapitel 6 beschreibt die Bewertung der Vergleichsergebnisse. Kapitel 7 beschreibt einen Lösungsansatz für eine ausgewählte Informationslücke.

Kapitel 2

Grundlagen

In diesem Kapitel sollen die Grundlagen vermittelt werden, die für das Verständnis der nachfolgenden Kapitel notwendig sind. Das folgende Bild zeigt die Struktur des Kapitels:

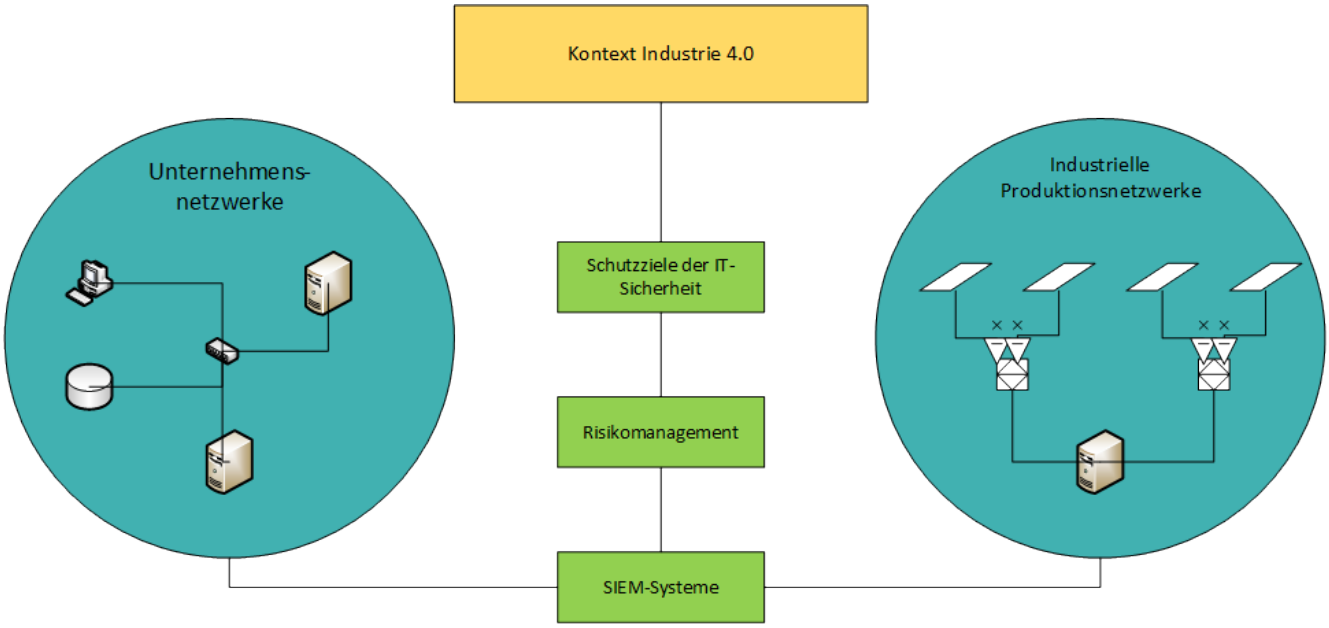

Abbildung 2.1: Aufbau des Grundlagenkapitels

Zunächst wird der Kontext der Industrie 4.0 beschrieben, um ein Rahmenverständnis für den aktuellen Stand und die Idee der Vernetzung von Fertigungsanlagen zu vermitteln. Darauf folgend werden die grundlegenden Schutzziele der IT-Sicherheit definiert, welche die Basis für eine IT-Sicherheitsanalayse darstellen. Da eine Analyse aller relevanten Daten und Datenquellen auf Grund der limitierten Ressourcen eines SIEM-Systems nicht möglich ist, ist eine Risiko Management Strategie von fundamentaler Bedeutung. Aus diesem Grund werden die Grundlagen für

Risiko Management in einer Sektion beschrieben. Aufbauend auf diesen Grundlagen erfolgt eine Beschreibung des SIEM-Systems- Konzepts, Komponenten und Funktionalitäten. Dies soll ein Grundverständnis für die zu betrachtenden Datenmengen schaffen, die ein SIEM-System in die Analyse und Korrelation der Daten benötigt. Schlussendlich sind Grundkenntnisse zu den Elementen sowohl von Unternehmensnetzwerken als auch industriellen Automatisierungsnetzwerken notwendig, weshalb eine grobe Beschreibung dieser Elemente hinzugefügt wurde.

2.1 Industrie 4.0 Konzept

Das Konzept der Industrie 4.0 stellt einen Ansatz zur Steigerung der Flexibilität und Effektivität der Wertschöpfungsketten. Dazu sollen verschiedene industrielle Anlagen innerhalb einer Wertschöpfungskette über das Internet (WAN) miteinander verbunden werden und sogenannte „Smart Factories“geschaffen werden, die sich besser auf die Wünsche der Kunden einstellen können. Wichtige Punkte dieses Konzeptes sind die horizontale und vertikale Integration. Zum aktuellen Zeitpunkt sind Unternehmens- und Industrienetzwerke voneinander getrennt und/oder durch eine Sicherheitsschicht voneinander getrennt, sodass keine Kommunikation zwischen Industrienetzwerk und Internet durchgeführt werden kann. Das Konzept der vertikalen Integration soll diesen Zustand verändern, sodass Produktionsverwaltung per Web Interface erfolgen kann. Gleichzeitig soll über das Konzept der horizontalen Integration eine stärkere Verknüpfung und Kommunikation der Elemente auf den verschiedenen Netzwerkebenen erreicht werden[31].

2.1.1 Stand der IT-Sicherheit in industriellen Produktionsnetzwerken

Industrielle Netzwerke wurden ursprünglich als isolierte Umgebungen entwickelt, sodass Sicherheitsmaßnahmen sich stärker auf den physischen Zugang zu den Anlagen als auf die Sicherung der IT Systeme konzentrierten [44].

Durch das Auftreten von Stuxnet auf das iranische Atomprogramm und andere Angriffe auf industrielle Anlagen ist die Sicherung der Kommunikation und des virtuellen Zugriffes auf die Komponenten verstärkt als wichtiger Punkt anerkannt worden [44]. Da in industriellen Netzwerken unterschiedliche Kommunikationstechniken existieren, die teilweise keinerlei Schutzmechanismen z.B. für die Verifizierung der Kommunikationspartner oder der Integritätsprüfung der ausgetauschten Informationen bieten, sind industrielle Netzwerke nach aktuellem Stand potentiell an-

fällig für Angriffe. Dementsprechend hat die Sicherung dieser Netzwerke eine hohe Priorität, da bei einer Verbindung mit dem Internet potentielle Angriffswege für Angreifer geschaffen werden, die bisher auf physischen Zugang oder das Einschleusen kompromittierter Speicher, wie z.B. USB Sticks, angewiesen waren [44].

Neben den Schwachstellen der Kommunikationstechnologien ist die Sicherung bei gleichzeitiger Garantie der Verfügbarkeit eine weitere Herausforderung. So ist es nach aktuellem Stand nicht möglich, regelmäßige Sicherheitsupdates für SPSen aufzuspielen, da dies das Anhalten der Produktionsprozesse und damit signifikante Verzögerungen und finanzielle Verluste zur Folge hat. Darüber hinaus müssen Neuerungen an der SPS Firmware und/oder geladenen Programmen jeweils zertifiziert und überprüft werden, um die Robustheit und Ausfallsicherheit zu garantieren. Dieser Zustand macht es schwierig gebräuchliche Sicherheitskonzepte aus Unternehmensnetzwerken in industriellen Netzwerken einzusetzen [44].

2.1.2 Forschungsgebiete der IT-Sicherheit

Um die IT-Sicherheit der industriellen Netzwerke zu erhöhen und eine Möglichkeit zu schaffen, die Konzepte der Industrie 4.0 sicher umsetzen zu können, wird an verschiedenen Themenfeldern geforscht, um Erkenntnisse zu gewinnen und Lösungen zu entwickeln. Quelle [44] zeigt eine mögliche Klassifizierung dieser Felder. Die Autoren teilen die Forschung in die folgenden Themenfelder auf:

- Sichere Kontrolle
- Erkennung von Eindringungsversuchen
- Simulation und Modellierung
- Kommunikations- und Infrastruktursicherheit

Sichere Kontrolle („Secure Control“)

Unter diesem Begriff werden Forschungen zusammengefasst, welche sich mit dem Schutz von Informationen (d.h. Daten in der Speicherung und bei der Übertragung) befassen. Es gilt die Verfügbarkeit der Daten abzusichern, z.B. gegen Angriffe, die eine (langfristige) Störung als Ziel haben (Denial of Service (DoS)) sowie die Integrität und Vertraulichkeit der Informationen zu sichern (zur Vermeidung von fehlerhaften Ausführungen und Täuschungen). Dabei spielt der Vertrauensgrad für die Korrektheit der übermittelten Daten („Veracity“) eine wichtige Rolle. Dabei sollen nicht nur Daten des IT-Bereichs, sondern auch Prozessdaten einbezogen werden. Kernthemen

sind etwa die sichere Identifizierung der Kommunikationsteilnehmer, Analyse von Entscheidungsmustern der Angreifer und damit verbundene Forschungen über die Robustheit bzw. die Anfälligkeit der Prozesses gegen Störungen.

Erkennung von Eindringungsversuchen

Wie in der klassischen IT-Sicherheit geht es bei diesem Gebiet darum herauszufinden, in welcher Form industrielle Kontrollsysteme anfällig sind gegenüber Angreifern und die darauf basierende Entwicklung von Algorithmen und Maßnahmen um diese Angriffe zu erkennen und vorbeugende Maßnahmen zu treffen. Bereits entdeckte Schwachstellen weisen u.a. unsichere Umsetzungen, aber auch Designfehler auf, die Angriffsmöglichkeiten eröffnen. Im Zuge dieses Themenfeldes haben sich Forscher auch u.a. mit dem Stuxnet-Programm beschäftigt, welches für die Manipulation des iranischen Atomprogramms verwendet wurde.

Simulation und Modellierung

Dieses Themenfeld beschäftigt sich hauptsächlich mit Fragen, die sich mit der Möglichkeit des Testens von IT-Sicherheitsmaßnahmen in industriellen Netzwerken auseinandersetzen. Die Grundproblematik teilt sich in zwei Elemente auf. Zum einen ist das Testen an realen Umgebungen sehr aufwendig, gefährlich und teuer, der Nachbau nicht öknomisch und aufwendig. Daher wird versucht mit sofware-basierten Frameworks zu arbeiten, um die Arbeit eines automatisierten Fertigungssystems zu simulieren. Ein sehr wichtiger Punkt ist dabei die akkurate Simulation von physikalischen Prozessen und die fehlende Kompetenz/Fachwissen der Sicherheitsforscher in diesem Bereich. Andere Bereiche der Modellierung und Simulation beinhalten das (komplexe) Verhalten solcher Umgebungen und die Analyse von Kommunikationsmustern (Datenverkehr) innerhalb dieser Systeme.

Kommunikations- und Infrastruktursicherheit

Die Grundlage dieses Forschungsbereichs ist u.a. das Streben der Industrie nach einem einheitlichen Kommunikationsmedium in Form des Industrial Ethernets. Kernthemen sind dabei etwa die horizontale und vertikale Integration der Netzwerkteilnehmer und das Verpacken von Prozessdaten in Ethernet-Pakete und die damit verbundenen Herausforderungen der klassischen IT-Sicherheit. Hinzu kommen die wachsende Verzahnung und Abhängigkeiten der Teilsysteme und die Analyse von Risiken und Anforderungen für einen sicheren Betrieb. Forschungen dieses Bereichs richten sich damit u.a. auf Angriffsszenarien auf allen Hierarchieebenen des Netzwer-

kes von der Unternehmensleitebene bis zur Feldebene, etwa Angriffsszenarien mit dem Ziel der Beschädigung oder Zerstörung des Fertigungssystems.

2.2 IT Security Schutzziele

In der IT-Sicherheit werden alle Ziele bezeichnet, die für die Sicherung von Systemen und Kommunikation zwischen Systemen sichergestellt werden müssen.

Mehr Kontext der Schutzziele einfügen

Dabei werden als Grundlage die folgenden Schutzziele betrachtet:

- Integrität: Bei der Kommunikation zwischen Systemen und der Speicherung bzw. dem Abruf von Daten auf einem System ist es wichtig, dass darauf vertraut werden kann, dass diese Daten nicht ohne Authorisierung verändert wurden. Daher befasst sich dieses Schutzziel mit der Korrektheit von Daten bzw. der korrekten Funktion eines Systems. Bei der Sicherstellung dieses Zieles geht es also darum, die Veränderung von übertragenen oder gesicherten Daten und Software sicherzustellen.

- Vertraulichkeit: Innerhalb einer Infrastruktur oder auf einem System existieren verschiedene Arten von Daten, darunter auch sensible Daten, die von nichtauthorisierten Personen missbraucht werden können. Daher ist es notwendig, den Zugriff zu Daten zu limitieren und dadurch diese Daten vor unauthorisiertem Zugriff zu schützen. Aktivitäten dieser Art werden dem Schutzziel Vertraulichkeit zugeordnet

- Verfügbarkeit: Diesem Schutzziel werden alle Aktivitäten zugeordnet, die den Betrieb von Systemen und die Erhaltung von Kommunikationswegen sicherstellen. Darunter fallen z.B. Maßnahmen, die den Zugriff von Kunden und/oder Mitarbeitern auf einen Webservice sicherstellen, auch im Falle eines Angriffes, der es zum Ziel hat, die Erreichbarkeit des Services zu unterbinden.

Schutzziele: Beispiel für Schutzziele formulieren

Neben diesen Schutzzielen existieren auch noch weitere Schutzziele wie z.B. Authentizität, Nichtabstreitbarkeit, Zurechenbarkeit und Privatsphäre.

2.3 Risiko Management

Risiko Management bezeichnet einen Bereich bzw. eine Sammlung von Aktivitäten, die der Abschätzung, Planung und Vermeidung von Risiken für ein Unternehmen

bzw. eine Organisation dienen. Ein Risiko wird als Kombination aus der Eintrittswahrscheinlichkeit eines bestimmten Sicherheitsereignisses und der damit verbundenen Konsequenzen definiert [62]. Nach der ISO27001 definiert ein Sicherheitsereignis laut Quelle [42] eine Änderung eines Zustandes in der Informationsverarbeitung, welches mindestens theoretisch eine Auswirkung auf die Sicherheit haben kann. Die zentralen Begriffe für die Beschreibung eines Risikos sind „Schwachstelle“und „Bedrohung“. Eine Schwachstelle wird im Kontext der Informationstechnologie als eine potentiell ausnutzbare Schwäche bzw. Fehler in einem Asset bezeichnet. Der Begriff „Asset“beschreibt laut ISO27001 alles, was für das Unternehmen bzw. die Organisation wertvoll ist, im Bezug auf Informationssysteme schließt dies u.a. Daten, Systeme, Anwendungen und Dienste (Services) ein. Unter einer Bedrohung wird wiederum ein potentieller Auslöser für einen Sicherheitsvorfall verstanden, welcher Schaden am Unternehmen verursachen kann [42].

Daraus folgend wird das Risiko beschrieben durch die Wahrscheinlichkeit, dass eine Bedrohung konkrekt vorhanden ist und eine Schwachstelle ausnutzt.

Die Aktivitäten des Risikomanagements werden sowohl durch das NIST als auch durch den ISO27001 Standard in folgende Schritte zusammengefasst: Risikoabschätzung, Risikovermeidung, Risikoakzeptanz und Risikokommunikation.

Die Abschätzung potentieller Risiken ist der erste Schritt, der sich in verschiedene Unterpunkte gliedert. Dazu zählen zunächst die Identifikation potentieller Risiken und die Analyse der gefundenen Risiken. Von der Beschreibung eines Risikos, die aus der Analyse folgt, ist der Detailgrad sehr wichtig für die Präzision der Risiko Abschätzung [42]. So kann etwa die Risikoabschätzung für die potentielle Ausnutzung einer Schwäche an einem Systems innerhalb des Unternehmensnetzwerkes präziser geschätzt werden, wenn der konkrete Zugriffsweg des Angreifers präzise beschrieben wird. Einen weiteren Unterpunkt der Risikoabschätzung stellt die Risikoevaluation ein, die die Bewertung und Beurteilung eines Risikos beinhaltet. Dies beinhaltet neben der konkreten Einschätzung einer Eintrittswahrscheinlichkeit auch die Erarbeitung potentieller Gegenmaßnahmen [62].

Der zweite Schritt umfasst die Risikovermeidung. Diese schließt die Priorisierung bestimmter Risiken sowie die Umsetzung und Pflege der Maßnahmen zur Vermeidung der priorisierten Risiken ein.

Als dritter Schritt wird die Risikoakzeptanz genannt. In diesem Schritt werden die übrigen Risiken neu bewertet. An diesem Punkt können abhängig von der Entscheidung der Verantwortlichen weitere Gegenmaßnahmen eingeleitet oder aber auch Risiken als Restrisiko für das Unternehmen akzeptiert werden.

Die resultierende Risiko Management Strategie wird im letzten Schritt mit be-

troffenen Vertragspartnern kommuniziert [62].

2.4 Security Information and Event Management

2.4.1 SIEM Konzept und Zweck

Im Umgang mit Security Information and Event Management (SIEM) Systemen werden oft die Begriffe „Information“, „Ereignis“ und „Daten“ verwendet. Um eine Grundlage für die Verwendung dieser Begriffe in der vorliegenden Thesis zu geben, werden diese wie folgt definiert:

- Ereignis (Event): Unter einem Ereignis ist in diesem Zusammenhang das Auftreten von systemrelevanten Aktionen zu verstehen. Dabei kann zwischen Systemereignissen, z.B. das Laden eines Programmes, und Benutzerereignissen, z.B. Anschläge auf der Tastatur, unterschieden werden.

- Daten: Daten sind die Bausteine, aus denen Ereignisse zusammengesetzt werden. Daten sind z.B. der Name des geladenen Programms oder der Wert einer Benutzereingabe.

- Information: Eine Information ist in diesem Kontext die Interpretation verschiedener Ereignisse, die eine Aussage über den Zustand eines Systems ermöglicht.

Die IT Infrastruktur von Unternehmen umfasst eine große Menge an Elementen wie Server, Arbeitsstationen, Netzwerkgeräte, Sicherheitssysteme und mobile Endgeräte (z.B. Laptops und Mobilfunkgeräte). Um diese Infrastruktur zu schützen, reicht es nicht, diese hinter einem Schutzwall zu positionieren. Es ist auch wichtig, zu erfassen, welche Aktionen von wem wie und von wo innerhalb des Netzwerkes ausgeführt werden. Daher ist es wichtig, dass es Informationsquellen gibt aus denen diese Informationen ausgelesen und bewertet werden können. In Unternehmensnetzwerken wird dies typischerweise von den Elementen selbst durchgeführt. So befinden sich z.B. auf einem Server mit den Betriebssystem Windows verschiedene Protokolldateien, die Informationen darüber speichern welcher Anwender sich zu welchem Zeitpunkt angemeldet hat, wie viele Versuche für die Eingabe des Passwortes verwendet werden und welche Prozesse von diesem Anwender ausgeführt wurden. Die Analyse dieser Protokolldateien kann fachkundigen Administratoren Aufschluss darüber geben welche Aktionen von dem Benutzer oder dem System durchgeführt wurden. Eine Herausforderung bei dieser Analyse ist die Bewältigung der schieren

Menge an Informationen und das Filtern relevanter Ereignisse. Selbst eine geringe Anzahl an Systemen kann eine große Menge an Daten produzieren welche von einem Administratorenteam ohne Hilfe von Werkzeugen nicht zu bewältigen sind. Mit Hilfe technischer Werkzeuge können die Daten in Protokolldateien gefiltert werden. Die Suche nach sicherheitsrelevanten Informationen kann dabei automatisiert werden und Administratoren können über Anomalien im Verhalten der Systeme informiert werden.

Allerdings bietet diese Vorgehensweise auch Nachteile. So ist die Definition der Regeln, nach denen Werkzeuge Protokolldateien durchsuchen, eine komplexe Aufgabe. Diese Regeln müssen eng genug gefasst sein, um die Anzahl der Sicherheitsmeldungen verwaltbar zu halten, aber auch weit genug, um potentiell bedrohliche Situationen zu erkennen. Für die Bewertung einer bedrohlichen Situation, müssen Administratoren nicht nur in der Lage sein, die Meldungen des Werkzeuges zu verstehen, sondern diese auch im korrekten Kontext einordnen zu können. Diese Einordnung kann in komplexen Netzwerken sehr schwierig sein.

Um diese Einordnung zu vereinfachen und die Anzahl an Falschmeldungen („False Positives“) zu reduzieren, ist es also notwendig nicht nur ein Element, sondern verschiedene Elemente im Zusammenspiel zu betrachten und die Informationen dieser Netzwerkelemente zu verknüpfen, also einen Zusammenhang von Informationen von verschiedenen Elementen zu erfassen. Für die Unterstützung der Administratoren bei dieser Aufgabe wurden Security Information and Event Management (SIEM) Systeme entwickelt.

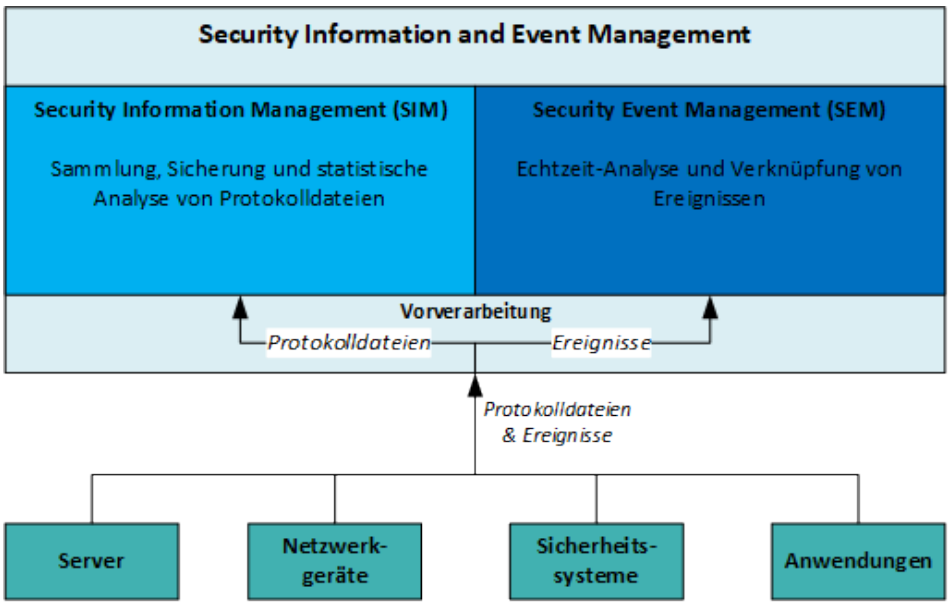

Abbildung 2.2: Schematische Darstellung des SIEM Konzeptes

Ein SIEM System ist also ein System, welches Informationen aus verschiedenen Quellen extrahiert, die einzelnen Informationsquellen analysiert und die gewonnenen Informationen in einen Zusammenhang bringt. Durch Aggregation von ähnlichen Daten und Korrelation dieser Daten lassen sich einzelne Alarmmeldung in einen Zusammenhang setzen und neue Informationen über den Wert und den Kontext der Meldungen gewinnen. Dadurch ist es möglich den Zustand eines Netzwerkes und seiner Elemente über ein zentrales System zu überwachen und kritische Situationen zu erkennen, die durch das Betrachten einzelner Elemente nicht erkennbar sind. Das SIEM System bildet damit eine zentrale Verwaltungsschicht oberhalb der Netzwerkelemente.

Um diese Funktion auszuführen, sind verschiedene Schritte notwendig. Die Aufgaben teilen sich dabei in zwei Bereiche auf: Security Information Management (SIM) und Security Event Management (SEM). SIM ist eine Unterkategorie des Log Management Feldes, das heißt, es beinhaltet Erfassung, Extraktion, Transfer und Sicherung von sicherheitsrelevanten Informationen in Protokolldateien. SEM umfasst Funktionen für die Analyse und Verknüpfung von Ereignissen in Echtzeit.

Der erste Schritt ist die Extraktion der Daten von den Netzwerkelementen. Dies kann entweder durch Anbindung einer geeigneten Schnittstelle durchgeführt werden (z.B. Meldungen von Sicherheitssystemen wie einer Firewall oder einem Intrusion Detection System (IDS)) oder durch Extraktion der Informationen über Kollektoren in Form von Software Agents. Der zweite Schritt betrifft die Übertragung der extrahierten Informationen. Diese muss sicherstellen, dass die Informationen vollständig und unverändert übertragen werden, da veränderte Informationen die Informationsgrundlage verändern, auf der das SIEM System seine Analyse durchführt.

Da nicht jedes System gleich ist und auch Protokolldateien und -formate sich deutlich unterscheiden können, müssen die vom SIEM empfangenen Informationen vorverarbeitet und in ein einheitliches Format umgewandelt werden. Dieser Schritt wird als „Normalisierung“ bezeichnet. Die normalisierten Daten können dann zentral gespeichert und für statistische Langzeitanalysen und statistische Verwertungen gespeichert werden. Dabei ist es wichtig, darauf zu achten, dass die abgelegten Daten nicht verändert werden können und dass Änderungen nachvollziehbar sind. Da die Datenmenge abhängig von der Größe und Komplexität sehr groß sein kann und die Ressourcen des SIEM Systems für die Analyse und Bewertung begrenzt sind, werden ähnliche Daten in einem Zwischenschritt zusammengefasst, sodass die zu analysierende Menge an Daten deutlich reduziert wird. Man spricht dabei von der Aggregation der Daten. Die Aggregation kann z.B. dadurch erfolgen, dass Ereignisse desselben Ursprungs und desselben Inhalts als eine Meldung zusammengefasst

und mit einem Zähler versehen werden, der die Anzahl der Meldungen widerspiegelt. Im nächsten Schritt werden die aggregierten Daten dann durch ein Regelwerk analysiert. Dieses Regelwerk wird manchmal auch als „Rule Correlation Engine“ bezeichnet. Dabei werden verschiedene Meldungen mit Regeln abgeglichen. Wird eine Regel als erfüllt angesehen, wird eine Alarmmeldung an verantwortliche Administratoren ausgegeben, sodass weitere Maßnahmen ergriffen werden können. Die Bewertungm ob eine Regel erfüllt wurde, hängt von der verwendeten Korrelationstechnik ab. So müssen z.B. eine Reihe von Bedingungen erfüllt werden, damit eine Regel als erfüllt angesehen wird. So könnte z.B. eine Regel beinhalten, dass ein Alarm ausgegeben wird, wenn mehrere nicht-erfolgreiche Anmeldungsversuche von einem inaktiven Account von einer nicht-registrierten IP-Adresse protokolliert wurden. Die Bildung solcher Regelwerke kann sehr komplex sein und durch verschiedene Techniken gebildet werden.

In der Praxis werden SIEM Systeme u.a. in Security Operations Center (SOC) eingesetzt. Das SIEM hilft in diesem Kontext nicht nur den Administratoren für die Verwaltung, sondern ersetzt auch die Verwaltung und Analyse über verschiedenen UIs unterschiedlicher Sicherheitsprodukte (z.B. von Firewalls, IDSs, Anti-Virus Software). Hier zeigen sich neben den Vorteilen von SIEM Systemen auch aktuelle Grenzen. So kann ein SIEM nur auf der Informationsbasis operieren, die in der jeweiligen Umgebung zur Verfügung steht. So kann sich das Fehlen von Informationen zum Kontext der Meldungen auf die Auswertung auswirken, da fehlende Informationen zu ungenauen Analysen oder False Positives bzw. False Negatives (kritische Meldungen, die als harmlos klassifiziert werden) führen können. Eine andere Limitierung sind Abfragezeiten von gespeicherten Langzeitdaten für die Echtzeitanalyse durch die entweder die Analyse oder die Menge der Daten begrenzt wird. Auf der technischen Seite ergeben sich zudem Herausforderungen entlang der Funktionskette eines SIEM Systems bei der Datenextraktion aus proprietären Formaten, der effektiven und sicheren Speicherung der Daten sowie der Analyse und Erstellung von Korrelationsregeln in komplexen Systemen.

2.4.2 SIM Log Management

Security Information Management (SIM) bezeichnet die Verwaltung von Protokolldateien im SIEM Kontext und bildet damit eine Untermenge des Log Management. Sicherheitsrelevant sind in diesem Falle alle Protokolldateien, die Auskunft über den Zustand eines Systems oder einer Kommunikation zwischen Netzwerkelementen geben können. Dazu zählen sowohl Protokolldateien von Betriebssystemen und Programmen als auch von Netzwerkswitches und anderen Elementen, die den Daten-

verkehr zwischen Teilnehmern im Netzwerk aufzeichnen. Eine Protokolldatei besteht aus Ereignissen, die Daten zu Aktionen (u.a. Zeitpunkt, ausführender Benutzer, ausführender Prozess (ID), ...) enthalten. Diese Daten können interpretiert werden und liefern Informationen über den Zustand des Systems zum bestimmten Zeitpunkt.

Log Management umfasst die Sammlung, Übertragung, Normalisierung, Zentralisierung und Aufbewahrung der Menge an Protokolldateien. In großen Unternehmen kann diese Menge mehrere hundert Gigabyte umfassen. Das Verwalten der Protokolldateien wird aus mehreren Gründen in Unternehmen als außerordentlich wichtig angesehen. Zum einen beinhalten Protokolldateien Informationen über den Zustand der Umgebung und können damit Rückschlüsse auf potentiell schädliche Aktionen zulassen, zum anderen können durch diese Informationen auch die Einhaltung von geltenden Richtlinien und Anforderungen gegenüber Kunden und Institutionen belegt werden. Daraus folgt, dass sichergestellt werden muss, dass die Informationen nicht nur korrekt aus den verschiedenen Quellen extrahiert werden können, sondern auch die Umwandlung und ggf. Veränderung der Daten dokumentiert werden muss, um die Integrität der Informationen zu gewährleisten. Dies spielt u.a. für die Sicherheitsanalyse eine wichtige Rolle, da nur mit den richtigen Informationen die korrekten Rückschlüsse auf die Sicherheit der Umgebung geschlossen werden können. So lassen sich u.a. mit forensische Analysen rückwirkend Aktionsketten, die zu einer Richtlinienverletzung oder eines nicht-authorisierten Eindringens in das System geführt haben, rekonstruieren. Darüber lassen sich aus den Informationen Grundlagen für den Normalzustand eines Systems ablesen und entsprechende Regeln formulieren.

Log Management: Unterschied zwischen Log Management Systemen und SIEM Systemen herausstellen/formulieren

Protokolldateien können aus fast jedem System gewonnen werden. Dazu zählen u.a.

- Anti-Malware und Anti-Virus Systeme
- Intrusion Detection Systems / Intrusion Prevention Systems
- Remote Access Software
- Web Proxieserver
- Vulnerability Management Software
- Network Access Control (NAC) Server
- Firewalls

- Router

Ein interessantes Thema in diesem Zusammenhang ist die Frage, welche Protokolldateien und Event-Daten sicherheitsrelevant sind. Moderne und komplexe IT-Umgebungen produzieren mehrere hundert Gigabyte an Event-Daten auf einer täglichen Basis. Daher ist die Definition sicherheitskritischer Elemente (Systeme, Kommunikationspfade, Applikationen, ...) eine wichtige Aufgabe. Dazu können unter anderem diese Elemente gehören:

- Protokolle der Security Controls (z.B. Firewalls, Intrusion Detection SystemIntrusion Prevention System, Data Loss Protection)
- Protokolle der Netzwerk Infrastruktur (z.B. Domain Name Service (DNS) Server, Dynamic Host Configuration Protocol (DHCP) Server, VPN Logs)
- Informationen über die Infrastruktur aus anderen Quellen (z.B. Informationen zu Systembestand und Netzwerksegment eines Elements)
- Informationen über das Unternehmen

Log Management: Log Management Architektur einfügen?

Log Management: Beispiel für Log Management Architektur einfügen

Eine große Herausforderung des Log Management ist das Schaffen einer Balance zwischen limitierten, verfügbaren Ressourcen für das Verwalten der Protokolldateien und der Menge an zu verarbeitenden Protokolldateien pro Zeiteinheit. Dabei spielen nicht nur die potentiell große Menge an Dateien in einer komplexen Umgebung eine Rolle, sondern auch der Umgang mit Inkonsistenzen zwischen vergleichbaren Protokollen (etwa in Bezug auf den Zeitstempel), der Umwandlung aus verschiedenen Formaten und das Wachstum an Daten mit dem Hinzufügen von weiteren Systemen in die Umgebung.

2.4.3 Kollektoren

Die Extraktion oder auch Sammlung der Daten von den Elementen des Netzwerkes bildet die Grundlage auf der das SIEM System (und ein Log Management System) funktioniert. Die Daten werden von vielen verschiedenen Geräten extrahiert, die eine gewisse gemeinsame Grundmenge an Daten bieten, darüber hinaus aber auch weiteren Kontext abhängig von Applikation, Betriebssystem oder Gerät selbst. Daher muss eine Schnittstelle geschaffen werden, die diese proprietären Datenformate mit Hilfe eines Parsers auszulesen und zu einem einheitlichen Format umwandeln [14].

Dies ist die Aufgabe der Kollektoren. Eine weitere Aufgabe des Kollektors ist zudem die eines Filters für relevante Ereignisse.

Ein Kollektor ist demnach ein Service, oder auch Software Agent, der diese Aufgabe übernimmt. Dabei sind verschiedene Elemente zu betrachten, wie die verfügbaren Daten, Ressourcen der Datenquelle, Kommunikation zwischen Datenquelle, Kollektor und SIEM System sowie der Ansatz für einen bestimmten Gerätetyp. So stellt etwa das Protokoll, das für die Kommunikation mit der Datenquelle genutzt wird, eine Art Sprache dar, in der die Kommunikationspartner miteinander kommunizieren. Diese Kommunikationsform muss vorher vereinbart worden sein und dabei kann auf eine große Variation an Kommunikationsprotokollen zurückgegriffen werden. Der Kollektor für den entsprechenden Geräte-, Betriebssystem- oder Applikationstyp muss natürlich dieses Protokoll unterstützen. Alternativ dazu können Daten auch in weit verbreiteten Formaten wie etwa im „syslog“ Format abgelegt und durch den Kollektor oder eine Schnittstelle übermittelt werden.

Bei dem Ansatz der Datensammlung kann bzgl. SIEM Kollektoren zwischen dem Agent-basierten und dem Agent-losen Ansatz unterschieden werden. Der Agent-basierte Ansatz wird durch die Installation der Kollektoren auf den jeweiligen Netzwerkelementen aufgebaut. Dadurch ergibt sich eine dezentrale Verarbeitung, die zum einen Ressourcen des SIEM Systems spart, zum anderen aber auch einen erhöhten Verwaltungsaufwand für Kollektoren auf den verschiedenen Elementen des Netzwerkes bedeutet. Der Agent-lose Ansatz hingegen nutzt Kollektoren als Schnittstelle, die vom zentralen SIEM System aus mit den Komponenten über Schnittstellen kommuniziert. Hier wird die Verwaltung der Kollektoren erleichtert, aber es muss auch die Netzwerklast berücksichtigt werden, die durch die Kommunikation zwischen den SIEM Kollektoren und den Datenquellen erzeugt wird. Zudem muss über die entsprechenden Schnittstellen sichergestellt werden, dass Daten nicht auf der Datenquelle selbst oder während der Übertragung verändert werden.

Orientiert an den Aussagen aus der Industrie wird mehr und mehr zum dezentralen Ansatz tendiert, da die Agenten zusätzliche Funktionalitäten bieten können sowie Herausforderungen im Bereich der Regelung von Remote-Administrationsrechten, Security Compliance, Ressourcenmanagement und der Speicherung von hoch privilegierten Zugriffsdaten zu verschiedenen Systemen auf einer zentralen Instanz. Allerdings ist der Einsatz von Agenten auf Elemente mit einem entsprechenden Betriebssystem beschränkt. Hardware-Elemente mit proprietären Betriebssystemen wie etwa Netzwerk-Geräte unterstützen den Agenten-basierten Ansatz nicht und müssen daher selbst die Informationen in einem für das SIEM verständlichen Format zu der SIEM Instanz schicken oder eine Schnittstelle für den administrativen

Remote-Zugriff bieten.

Herausforderungen auf Basis von Protokolldateien

Unabhängig von der Wahl des Ansatzes bestehen für die Analyse der Protokolle einige Herausforderungen, die auch oder im Speziellen Kollektoren betreffen. Da Protokolldateien pro System oder gar pro Anwendung in unterschiedlichen Formaten und mit unterschiedlichen Strukturen angelegt werden, muss ein Kollektor in der Lage sein, sowohl das Format, die Struktur als auch gängige Konstrukte zu erkennen und auszulesen. So muss ein Kollektor pro System, auf die systemspezifischen Schnittstellen und/oder Protokolldateien zugreifen können. Die Protokolldateien auf dem System können abhängig von der Systemart und/oder dem Betriebssystem bspw. als Textdatei, XML-Datei oder in einem binären Format abgelegt werden. Die Einträge innerhalb der Protokolldateien können einzeilig oder mehrzeilig gespeichert werden. Ein Kollektor muss in der Lage sein, diese Informationen zu erkennen und in den Lese- und Verarbeitungsprozess einfließen zu lassen [39].

Neben diesen formalen Herausforderungen besteht zudem die Möglichkeit der Korruption von Protokolldateien sowie das Vorkommen von Inkonsistenzen. Ein Kollektor muss also in der Lage sein, den Lese- und Verarbeitungsprozess trotz fehlerhafter Einträge fortzuführen. Die Erkennung von Inkonsistenzen kann eine weitere Herausforderung im Bezug auf wiedersprüchliche Einträge, sofern dies Teil der Filterungsaufgabe des Kollektors ist. In diesem Bezug spielt natürlich auch die Performanz des Kollektors bei der Verarbeitung eine Rolle. So ist z.B. die Frage zu stellen, inwiefern bestimmte Protokolldateien einen sogenannten „Performance Overhead“ erzeugen basierend auf der Formatierung und der Menge des Inhalts [39].

2.4.4 Event Verarbeitung

Durch die hohe Anzahl an Daten und Protokollquellen ist es notwendig, relevante Einträge herauszufiltern und weiter zu verarbeiten, sodass die gewonnenen Informationen für die Überwachung und Analyse der Organisation verwendet werden können.

Die Ereignisdaten, die aus den Protokollen gewonnen werden, müssen daher zusammengefasst und auf Grund der Diversität der Protokollquellen vereinheitlicht werden. Aus diesem Grund können Protokolleinträge von den Kollektoren ausgelesen, und sofern möglich, die Daten in neue Protokolldateien mit einheitlichen Feldern transferiert werden. Im Kontext der SIEM Systeme gibt es verschiedene Begriffe und

Ebenen für die Zusammenführung unterschiedlicher Datensätze zu einem einheitlichen Datensatz, im Bezug auf die Vereinheitlichung von Protokolleinträgen wird in diesem Fall der Begriff „Normalisierung“ verwendet. Dabei werden Protokolleinträge (Ereignisse) in verschiedene Kategorien unterteilt und die Daten in entsprechend strukturierten Datensätzen gespeichert [52].

Diese kann, abhängig von der Menge an unterschiedlichen Protokolldateien und -quellen, auf Grund der notwendigen Vorarbeit für die Erarbeitung eines geeigneten Datenbank Schemas komplex oder sehr schwierig sein [72].

Die zweite Aktivität im Bezug auf die Verarbeitung von Protokolleinträgen ist die Aggregation. Bei der Aggregation werden ähnliche Ereignisse, meist basierend auf bestimmten Feldern wie etwa Quell-IP, Ziel-IP und Event-ID, zu einem Ereignis zusammengefasst. Diese Methodik hilft dabei, die Analyse der eingehenden Daten zu vereinfachen und kann damit die benötigte Menge an technischen und zeitlichen Ressourcen reduzieren. Auch kann sie ggf. bei der Vorverarbeitung helfen, die Menge der zu übertragenden Informationen, und damit die Netzwerklast, deutlich zu reduzieren. Zu guter Letzt können durch die Reduzierung der Datenmenge diese Daten formatiert schneller und mit geringerem Speicherverbrauch gespeichert und abgerufen werden. Die Verwendung von Aggregationsmethoden kann jedoch auch dazu führen, dass ggf. wichtige Daten nicht zur Verfügung stehen. Daher ist es in der Praxis wichtig, die Aggregationsmethodik bzw. -regeln abhängig von bestimmten Faktoren wie etwa der Häufigkeit oder Bedeutung im Bezug auf den Sicherheitskontext [53].

Diese Aufgaben können durch das zentrale System oder verteilte Agenten durchgeführt werden.

2.4.5 Security Event Management

Security Event Management befasst sich mit der Echtzeit-Analyse von normalisierten Events, der Korrelation dieser Events sowie der Benachrichtigung von Sicherheitsadministratoren und der Darstellung relevanter Informationen. Man kann in diesem Zusammenhang auch von kontext-bewusster Überwachung sprechen.

Bei der Echtzeit-Analyse werden die extrahierten, normalisierten Events auf ihre Bedeutung untersucht. Dabei werden die Events einzeln für sich betrachtet und basierend auf den Daten(feldern) Informationen gewonnen. Diese Informationen werden bei der Korrelation der Events verwendet, um einen Zusammenhang herzustellen. Dabei werden u.a. zusätzliche Informationen hinzugezogen, z.B. die Quelle der Informationen und Informationen zu der entsprechenden Quelle. Dies ist notwendig, um die Herstellung falscher Zusammenhänge zu minimieren. Die Herstellung

der Korrelation basiert auf den verwendeten Techniken. So können Regelwerke auf verschiedene Art und Weisen hergestellt werden. Die Spannweite der Korrelationsregeln reicht dabei von simplen Wertevergleichen über Regeln mit verschiedenen Bedingungen hin zu der Erzeugung komplexer Szenarien durch Machine Learning und Big Data Ansätze. Wird eine Korrelationsregel erfüllt, wird ein Alarm ausgegeben. Diese Alarme können abhängig von ihrer Priorität und Häufigkeit wiederum durch Aggregationsregeln zusammengefasst werden.

Die durch die Korrelationsregeln ausgelösten Alarme helfen den Analysten in einer Organisation, kritische Situationen zu erkennen. Sie dienen dabei als Indikator für eine Kompromitierung (IoC, Indicator of Compromise). In größeren Organisationen sind die Analysten Teil eines „Security Operation Centers“. In diesem Zentrum arbeiten Analysten mit verschiedenen Qualifikationen. So werden Indikatoren etwa zunächst von einem Analyst der Stufe 1 auf einen potentiell fälschlich ausgelösten Alarm (False Positive) untersucht und bei Feststellung eines Problems an Analysten der Stufe 2 weitergegeben. Diese Struktur dient der effektiven Bewältigung der ausgelösten Alarme [11].

2.4.6 Features & Gemeinsamkeiten von SIEM-System Anbietern

Für die allgemeine Betrachtung von SIEM Systemen kann es nützlich sein, die Gemeinsamkeiten und Unterschiede gebräuchlicher SIEM Anbieter zu kennen. Nach den Quellen [65, 46, 16] gehören zu dieser (nicht vollständigen) Liste AlienVault, HP Enterprise, IBM, LogRythm, RSA, Solar Winds und Splunk.

Die Produkte dieser Anbieter bieten jeweils ein Minimum an grundlegenden Funktionen eines SIEM Systems zusammen. Dazu zählt die Möglichkeit der Sammlung von Protokolldateien von vielen typischen Quellen, die Archivierung und Analyse der erhobenen Daten sowie die Korrelation und Echtzeit-Überwachung der Quellen (d.h. Server, Netzwerkgeräte, Sicherheitslösungen, Anwendungen, etc.). Zusätzlich setzen die Produkte sogenannte „Threat Intelligence Feeds“ ein, um (Kontext-)Informationen zu neuen Schwachstellen und/oder Bedrohungen zu erhalten.

Die Unterschiede ergeben sich sowohl in der Architektur als auch in der Spezialisierung auf die Größe einer Organisation. Bzgl. der Architektur ergeben sich prinzipiell drei unterschiedliche Modelle: Cloud-basierte Lösungen, hardwarebasierte Lösungen und anwendungsbasierte Lösungen. Cloud-basierte Lösungen bieten SIEM Funktionalitäten ausgelagert als einen Service an, der die Protokollquellen einer Organisation sammelt und analysiert. Hardware- und anwendungsbasierte Lösungen

werden innerhalb der Infrastruktur der Organisation installiert und von dieser Organisation konfiguriert und gepflegt. Die Größe der Organisation kann dabei von sogenannten „SMBs“ (Small to Middle Sized Businesses) bis zu sehr großen und komplexen Infrastrukturen reichen.

Die Spezialisierungen der einzelnen Produkte können nach verschiedenen Kriterien festgelegt werden. Nach Studie der Quellen ergeben sich die folgenden Kategorien für Spezialisierungspunkte:

- Möglichkeiten der Erschließung von Protokollquellen: Dies kann entweder durch eine größere Menge an unterstützten Protokollquellen erfolgen oder durch eine flexiblere Anpassungsmöglichkeit (mit höherem administrativen Aufwand)(s. Splunk)
- Möglichkeiten der Korrelation unter Einbeziehung der Protokolldaten sowie weiterer externer Quellen für die Integrierung von Kontextdaten wie etwa Threat Intelligence Feeds
- Zusätzliche Fähigkeiten, wie etwa „Deep Packet Inspect“ oder „User Behavior Analytics“
- Möglichkeiten der Visualisierung des Status von Datenströmen und Netzwerken
- Schwierigkeitsgrad der Installation und Einrichtung

Bei dieser Kategorisierung sei angemerkt, dass diese Liste keinesfalls vollständig ist oder sein kann. Nach den Einschätzungen der oben genannten Quellen muss für jede Organisation individuell geprüft werden, welche Fokuspunkte notwendig sind. Diese Kategorien geben lediglich grundlegende Elemente an. Quelle [66] etwa kategorisiert nach sieben grundlegenden Fragen: Grundlegende Unterstützung von Protokollquellen und -format, Möglichkeiten der zusätzlichen Protokollierung von zusätzlichen, relevanten Daten, die Nutzung und Qualität von Threat Intelligence Feeds, Möglichkeiten für forensische Analysen, zusätzliche Möglichkeiten für die Analyse und Examinierung von Daten, Qualität von automatischen Reaktionsmaßnahmen und Unterstützung für die Erfüllung von Industriestandards.

2.5 Unternehmensnetzwerke

Für den Vergleich zwischen Unternehmensnetzwerk und industriellem Netzwerk ist es notwendig, beide Netzwerke zu kennen. Dieser Abschnitt stellt eineFundament zur

Verfügung, auf dessen Basis die Analyse des Modells eines Unternehmensnetzwerkes gestartet werden kann. Das Ziel ist es, einen guten Überblick über die allgemeinen Elementtypen in solchen Netzwerkinfrastrukturen zu geben und ihre Kerncharakteristiken zu beschreiben und ihre Anordnung über Architekturtypen in den Kontext zu setzen. Neben den Netzwerkelementen als Quelle der Daten, müssen zudem die Kommunikation bzw. die verwendeten Kommunikationsschnittstellen in Form der Netzwerkprotokolle berücksichtigt werden, da die versendeten Pakete ebenfalls wichtige Daten enthalten, um das Verhalten der Systeme zu verstehen. Zuletzt soll noch eine grobe Übersicht über potentielle Angriffsszenarien bzw. -techniken gegeben werden. Die Kenntnis dieser Szenarien ermöglicht es später in der Analyse, einen Bezug zu der Bedeutung der verfügbaren Daten des Netzwerkes herzustellen.

2.5.1 Geräte

Ein Unternehmensnetzwerk kann aus vielen verschiedenen Elementen bestehen. Dies beinhaltetet (verteilte) Anwendungen und Dienste, die auf der Basis von verschiedenenen, teils spezialisierten, Betriebssystemen installiert werden. Im Bezug auf physische Geräte können die Elemente jedoch in drei Basiskategorien unterteilt werden:

- Endgeräte (PCs, Laptop, Handy, ...)
- Server
- Netzwerkgeräte (Switches, Router, Firewalls, Sicherheitselemente (z.B. Intrusion Detection Systems))

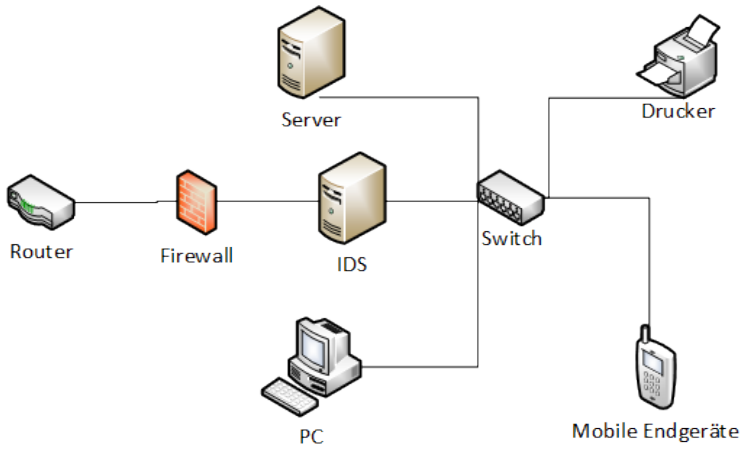

Abbildung 2.3: Elemente eines Unternehmensnetzwerkes

Um Unternehmensprozesse durchzuführen werden verschiedene Server-Elemente benötigt. Ein Server ist ein zentrales Computerelement, welches mit mehreren anderen Elemente verbunden werden kann und entsprechenden Zugriff von diesen Elementen erlaubt. Zudem verfügen Server oft über leistungsfähigere Hardware und Speicherressourcen als ein Endgerät. Server werden benutzt, um zentrale Applikationen bereitzustellen und zu verwalten. Dazu zählen neben Applikationen und (zugehörigen) Datenbanken auch Benutzerverwaltungssysteme und Sicherungscluster (Cluster := eine Menge an Servern, die durch eine zusätzliche Virtualisierungsschicht zu einem Element verknüpft werden). Zu den üblichen Serverarten zählen Webserver, Datenbankserver, Anwendungsserver, Proxyserver und Dateiablagesysteme, auf die über eine Netzwerkschnittstelle zugegriffen werden kann.

Ein Webserver stellt über das Netzwerkprotokoll HTTP (Hypertext Transport Protocol) bzw. HTTPS erreichbare Dienste, wie etwa Webseiten oder Webdienste, zur Verfügung. Zu den typischerweise verwendeten Webservern zählen Microsofts IIS sowie Apache Webserver (meistens auf Basis eines Linux-basierten Servers wie etwa Ubuntu, CentOS oder RedHat). Ein Datenbankserver stellt den Zugriff auf eine strukturierte Menge von Daten in Form einer Datenbank über das Netzwerk zur Verfügung. Das verwendete Netzwerkprotokoll ist abhängig von dem installierten Datenbanktyp. Typische Datenbanken sind etwa Microsoft SQL Server, Oracle Database oder MySQL. Ein Anwendungsserver ist meist ein dedizierter Server für eine spezielle Anwendung, die einen Dienst für das interne Netzwerk bereitstellt. Beispiele sind etwa VPN-Server, Email-Server oder Sicherheitszugänge zu speziell geschützten Resourcen des Netzwerkes. Dateiablagen (FTP Server) nutzen die File Transfer Protocol (FTP) Schnittstelle, um den Zugang zu auf dem Server gespeicherten Daten (z.B. Dokumente verschiedenen Typs) über das Netzwerk verfügbar zu machen. Proxyserver schließlich dienen als Zwischenpunkt zu einem anderen Netzwerk. Proxyserver werden z.B. benutzt, um sonst vom WAN/Internet abgeschotteten Servern oder Endgeräten den kontrollieren Zugang zum Internet zu ermöglichen.

Netzwerkgeräte dienen der Verbindung und/oder Überwachung der Kommunikation zwischen den einzelnen Netzwerkelementen. Diese lassen sich typischerweise in Router und Switche unterteilen. Router dienen der Verknüpfung mehrerer lokaler Netzwerke. Typischerweise existieren in Unternehmensnetzwerken eine Vielzahl an unterschiedlichen lokalen Netzwerken (auch Netzwerksegmente genannt), die es zu verbinden gilt. Daher kann im Rahmen dieser Netzwerken von zwei Routertypen gesprochen werden: „Edge-Router“ und „Core-Router“. Der Typ Edge-Router

bezeichnet Router, die an den Grenzen des Unternehmensnetzwerkes zum WAN/Internet platziert sind, d.h. sie stellen die Verbindung zum Internet her. Core-Router wiederum dienen dazu, die verschiedenen Netzwerksegmente innerhalb des Unternehmensnetzwerk miteinander zu verbinden. Diese Geräte existieren, je nach Hierarchiestufe, in unterschiedlichen Fertigungen und sind angepasst für die jeweilige Aufgabe und Position im Netzwerk.

Die Netzwerksegmente werden aus Sicherheitsgründen oft bestimmten Zonen zugewiesen (dazu später mehr im Abschnitt „Architekturtypen“). Die Verbindungen, die durch Core-Router hergestellt werden, werden typischerweise durch Firewalls abgesichert. Firewalls stellen eine Art vorkonfigurierten Filter dar, der abhängig von den konfigurierten Regeln und Parametern nur Datenverkehr mit bestimmten Netzwerkprotokollen aus und zu bestimmten Teilen des Netzwerkes zulassen. Zusätzlich zu Firewalls werden außerdem Sicherheitselemente benötigt, um den Datenverkehr zwischen den Netzwerkelementen zu überwachen. Diese Elemente werden Intrusion Detection Systems (IDS) bzw. Intrusion Prevention Systems (IPS) genannt. Ein IDS kann an einer bestimmten Stelle im Netzwerk platziert werden und ein Netzwerkpaket bzw. eine Sequenz von Netzwerkpaketen mitlesen und auf verdächtige Inhalte untersuchen. Für die Untersuchungsmethodik wird typischerweise eine Kombination aus der Suche nach bekannten Mustern im Datenverkehr als auch nach anomalen Datenpaketen oder -sequenzen verwendet. Ein IPS ist prinzipiell ein Gegenstück zu einer Firewall. Die Aufgabe des IPS ist es, den Datenverkehr zu analysieren und nach vorkonfigurierten Regeln Datenpakete von spezifizierten Netzwerkprotokollen aus spezifizierten Teilen des Netzwerkes zu blockieren. Es existieren noch weitere potentielle Sicherheitselemente (wie etwa ein hardware-basierendes SIEM System), die jedoch in ihrer Grundfunktion auf Basis eines Server fungieren und daher hier nicht näher erläutert werden.

Endgeräte werden von Mitarbeitern und Gästen des Unternehmens verwendet als Zugangswerkzeug zum Unternehmensnetzwerk. Basierend auf den Richtlinien und Sicherheitsvorgaben sind diese Geräte deutlich eingeschränkt. Die Benutzer auf den Endgeräten werden üblicherweise in eine Domäne eingebunden, um eine zentrale Benutzerverwaltung zu ermöglichen. Endgeräte können sowohl stationär als Arbeitsstation als auch mobil in Form von Laptops, Tablets oder Mobiltelefonen existieren.

Sicherheitsanforderungen (Zweck und grobe Beschreibung) formulieren

2.5.2 Architekturen & Anforderungen

Unter „Unternehmensarchitekturen“ bzw. dem Design dieser Architekturen versteht man eine Sammlung an Architekturen aus verschiedenen Perspektiven. Diese definieren die verschiedenen Sichten und Schichten, die ein Unternehmen definieren. Im speziellen wird der Begriff Unternehmensarchitektur definiert als ein zusammenhängendes Ganzes von Privilegien, Methoden und Modellen, die genutzt werden im Design und der Umsetzung von organisatorischen Unternehmensstrukturen, Unternehmensprozessen, Informationssystemen und -Infrastrukturen [45]. Im Kontext der Betrachtung von Informationsquellen für SIEM Systemen wird hier nur die Infrastruktur der technischen, sicherheitsbezogenen Architektur betrachtet. Das Ziel ist, eine grobe Übersicht zu vermitteln, um einen Kontext für die Relationen der Netzwerkelemente zu vermitteln.

Während die Architekturdetails, oder genauer definiert die Infrastrukturelemente, abhängig von anderen Modellen und Perspektiven gestaltet werden, lassen sich folgende grundlegende Kommunikations-Bereiche definieren: Extranet (Zugang zum globalen Netzwerk), demilitarisierte Zone (Zugangsbereich zum Unternehmen) und Intranet (internes Netzwerk des Unternehmens), ggf. mit weiteren Unterteilungen [70].

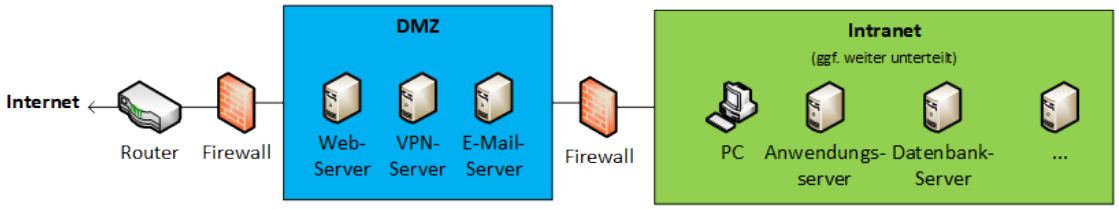

Abbildung 2.4: Schematisches Beispiel der Netzwerksegmentierung

Zunächst besteht eine Verbindung zum Internet (Extranet). Diese wird durch gesicherte Gateways ermöglicht. Damit allerdings keine direkte Verbindung von diesen Gateways in das interne Netzwerk gewährt wird, wird eine zusätzliche Netzwerkzone zwischen das Gateway und das interne Netzwerk geschaltet. Diese Zone wird demilitarisierte Zone (DMZ) genannt. Innerhalb der DMZ werden Server angebunden, die aus dem Internet erreichbar sein sollen. Dies umfasst u.a. Webserver, die einen oder mehrere Webservices beherbergen oder Sprungserver (z.B. für eine VPN-Verbindung). Diese Server werden im Bezug auf ihre Kommunikationsfähigkeit limitiert, sodass keine Netzwerkverbindung von diesen Servern in das interne Netzwerk hergestellt werden kann. Die Herstellung einer Verbindungnur ist nur in umgekehrter Richtung möglich. Das interne Netzwerk wiederum kann in viele verschiedene

weitere Netzwerkzonen und Domänen unterteilt sein. Diese Segmentierung dient der Strukturierung des Netzwerkes und eröffnet zusätzlich die Möglichkeit, weitere Sicherheitsschichten in die Umgebung einfließen zu lassen. Das interne Netzwerk beherbergt Endgeräte von Mitarbeitern, Server mit Firmendaten, Datenbankserver und Datensicherungscluster. Alle Elemente in einer solchen Umgebung sowie die Kommunikation zwischen diesen Elementen können von verschiedenen Sicherheitssystemen überwacht werden.

2.5.3 Kommunikation

In Unternehmensnetzwerken kommunizieren die Elemente auf Basis der Ethernet-Technologie und dem Internet Protocol (IP). Der zugehörige TCP/IP Stack bestimmt die Grundlagen der Kommunikation durch die Zuweisung einer eindeutigen IP Adresse. Die Zuweisung dieser IP Adresse kann fest zugeordnet werden (z.B. für Server), manuell oder automatisch über das Dynamic Host Configuration Protocol (DHCP) erfolgen, welches einem neuen Netzwerkelemente automatisch eine freie IP Adresse aus einem Adressenpool zuweist. Die meisten Elemente in einem Netzwerk werden einer sogenannten Domäne zugeordnet. Eine Domäne ist eine Zusammenstellung verschiedener Netzwerkelemente, die eine zentrale Benutzerverwaltung für die zugehörigen Elemente erlaubt. Zu einem der wichtigsten Netzwerkdienste zählt der „Domain Name Service“ (DNS). Dieser Netzwerkdienst erlaubt es den Namen einer Domäne einer oder mehreren IP-Adressen zuzuordnen und übernimmt die Beantwortung von Anfragen zur Namensauflösung. Die Namensauflösung wandelt einen Domänennamen in eine zugehörige IP-Adresse um, die von dem anfragenden Gerät genutzt werden kann, um eine Verbindung herzustellen. Dies eröffnet menschlichen Benutzern die Möglichkeit, eine bestimmte Domäne (z.B. „google.de“) über ihren Namen anzufragen anstatt über das Wissen um die korrekte IP-Adresse (die sich ggf. ändern kann) verfügen zu müssen. Die Ausführung der Anfrage und der Empfang der Antwort werden von einem lokal installierten „DNS-Resolver“ übernommen. Der DNS ist ein hierarchischer Verzeichnisdienst, das bedeutet, dass die Administration und Namensauflösung über ein hierarchisches Netz bestehend aus Namensservern durchgeführt wird. Der Namensraum des Internets wird dabei in einem hierarchischen Baum zunächst in sogenannte „Top-Level Domains“ (z.B. „de“) unterteilt, die wiederum in weitere Subdomänen unterteilt werden. Der von rechts nach links zusammengesetzt Pfad ergibt den Namen der Domäne. Dieser hierarchische Aufbau ermöglicht den strukturierten Aufbau der Namen und eine umgekehrt hierarchische Abfragemöglichkeit für die Namensauflösung. Dieses Konzept kann nicht nur im Internet, sondern auch lokal innerhalb eines Unternehmensnetzwerkes verwendet

werden [67]. Basierend auf der IP-Adresse kann eine verbindungsorientierte Kommunikation per Transmission Control Protocol (TCP) oder eine verbindungslose Kommunikation per User Datagram Protocol (UDP) erfolgen. Ist es eine Priorität, dass eine Nachricht vollständig und in richtiger Reihenfolge übertragen wird, wird z.B. TCP verwendet. Auf weiteren zusätzlichen Schichten können weitere Funktionalität wie etwa eine kryptografische Verschlüsselung erfolgen. Die verschiedenen Schichten sind u.a. im OSI-7-Schichtenmodel festgehalten. Das OSI-Modell bietet eine herstellunabhängige Grundlage für das Design von Kommunikationsprotokollen und Computernetzwerken. Das OSI-Modell unterteilt die Kommunikation in sieben Schichten:

- Schicht 7: Anmeldung - Funktionen und Anwendung
- Schicht 6: Darstellung - Umwandlung von Daten in ein systemunabhängiges Format
- Schicht 5: Kommunikation - Steuerung des Datenaustausches (Sessions)
- Schicht 4: Transport - Zuordnung von Datenpaketen zu einer Anwendung
- Schicht 3: Vermittlung - Routing
- Schicht 2: Sicherung - Segmentierung der Pakete
- Schicht 1: Bitübertragung - Umwandlung für die physikalische Übertragung

Jede Schicht fügt den vorherigen Schichten neue Elemente hinzu, um die Kommunikation um zusätzliche Funktionalitäten und Eigenschaften zu erweitern [43].

2.5.4 Bekannte Angriffsvektoren

Für das Risiko Management wie auch für die Implementierung von Verteidigungsmaßnahmen ist es notwendig, die gebräuchlichen Angriffsvektoren zu kennen, die ein potentieller Angreifer nutzen kann, um unberechtigterweise auf Komponenten in einem Unternehmensnetzwerk zuzugreifen.

Ein Angriffsvektor ist eine Technik, durch die ein Angreifer ein Netzwerk oder Netzwerkelemente angreifen oder ausnutzen kann. Angriffsvektoren helfen dabei, Schwachstellen der angegriffenen Komponente, inklusive dem menschlichen Benutzern, auszunutzen. Eine Schwachstelle kann ein programmierter Fehler in einer Anwendung sein, ein Fehler in der Prozesskette einer Anwendung, Schnittstelle oder eines Netzwerkprotokolles oder eine Lücke in der Absicherung der Komponente [71].

Da es viele verschiedene Komponenten in einem Unternehmensnetzwerk gibt in vielen verschiedenen Versionen und mit vielen verschiedenen Abhängigkeiten, Kommunikationsprozederen und Anwendungsarten, ist es nicht möglich, sich hundertprozentig vor Angriffen zu schützen. Jedoch ist es möglich, durch Kenntnis von bekannten Angriffsvektoren Schwachstellen zu schließen und somit das Risiko eines erfolgreichen Angriffes zu verringern und diesen zu erschweren.

Für Unternehmensnetzwerke lassen sich Angriffsvektoren grob in die folgenden Kategorien unterteilen [12]

- „Malware“
- „Denial-of-Service“
- „Angriffsvektoren auf Webinhalte “

Malware

Malware, auch Schadsoftware, ist der Überbegriff für Programme, die auf einem Computersystem mit bösartiger Absicht ausgeführt werden. Eine Malware kann etwa dazu benutzt werden, um über den infizierten Host auf Informationen und andere Komponenten des Netzwerkes zuzugreifen, die Resourcen des Hosts in Form von Rechenleistung, Netzwerkbandbreite, o.ä. zu nutzen oder den Host zu kontrollieren. Eine populäre Methode, um Malware auf einem Ziel auszuführen, ist die Versendung als Anhang von gefälschten E-Mail, etwa in der Form von Spam-Emails oder Phishing-Emails (personalisierte E-Mails für den entsprechenden Benutzer) oder auch Spear-Phishing (speziell angepasste E-Mails nach Hintergrundrecherchen der Zielperson). Der Angreifer versucht in diesem Falle, den Benutzer dazu zu bringen etwa einen Link zu einer gefälschten Webseite anzuklicken, über die der Schadcode im Hintergrund auf den Computer des Ziels heruntergeladen wird oder einen modifizierten Anhang zu öffnen, durch den der Schadcode auf dem Computer installiert wird. Gängige Malwaretypen sind trojanische Pferde, Computerviren, Computerwürmer, „Rootkits“, logische Bomben und Spionageprogramme[12].

Die Unterscheidung dieser Typen ist wichtig, da die Funktionalität und Operation der verschiedenen Typen Auskunft über potentielle Indikatoren geben kann. Aus diesem Grund werden die verschiedenen Typen im Folgenden kurz beschrieben. Ein Computervirus ist ein Programm, welches im Programmcode eines modifzierten Programmes oder einer modifizierten Datei versteckt ist. Wie der biologische Virus benötigt dieser Malware-Typ einen Host, also anderes Programm oder einen

Prozess, um sich weiter zu verbreiten. Ein Computerwurm hingegen ist eine unabhängige Anwendung und kann sich unabhängig von seinem Host der Ressourcen des Hosts bedienen, um sich weiterzuverbreiten. Sowohl Computerviren als auch Computerwürmer werden genutzt, um weitere Computer im Einflussbereich des infizierten Hosts zu infizieren und den Einfluss des Angreifers auszuweiten. Ein Rootkit wird oft in diesem Zusammenhang verwendet, um seine eigene Ausführung und die Ausführung anderer Malware zu verbergen, etwa durch das Verbergen der Prozess im Windows Task Manager oder durch Blockierung des Benutzerzugriffes auf sein Quellenverzeichnis. Dafür wird das Rootkit oft als Folgeschritt der Infizierung eines Hosts von einer Quelle heruntergeladen und mit administrativen Privilegien installiert, um Zugriff auf die Betriebssystemkomponente zu erlangen.

Eine weitere Malware-Art ist das sogenannte „trojanische Pferd“. Diese Art von Malware ermöglicht die Ausführung von Kommandos auf dem infizierten Host ohne das Wissen des Benutzers und können unabhängig von den Operationen des Hosts ausgeführt werden. Trojanische Pferde ermöglichen u.a. die Öffnung von Hintertüren, die dem Angreifer den Zugriff auf das kompromitierte System ermöglichen. Diese Technik wird oft für die Erstellung oder Erweiterung von Botnetzen verwendet. Botnetze bestehen aus einer Sammlung kompromitierter Systeme, die von einem „Command& Control-Server“ gesteuert werden können. Dieser Server (Botmaster) ermöglicht die Versendung von Befehlen an die kompromitierten Systeme (Bots) zur Ausführung dieser Befehle auf den Systemen. Logische Bomben sind eine Malware-Art, die auf ein System heruntergeladen, deren Ausführung jedoch verzögert wird, bis eine vordefinierte Bedingung bzw. eine Sammlung von Bedingungen erfüllt werden. Dies kann von einem bestimmten Zeitpunkt (Datum) bis zu komplexen Abhängigkeiten von Versionen installierter Software, Patches oder anderer Systemvariablen reichen. Unter „Spyware“ versteht man eine Malware, die Informationen des kompromitierten Systems an den Angreifer übermittelt. Dies können z.B. Tastatureingaben (Key Logger), Screenshots (Screen grabber) oder auch Videostreams vom aktuellen Bildschirm sein[12].

Denial-of-Service

Neben dem Ziel, Informationen zu stehlen oder Systeme zu kontrollieren, kann ein Angreifer auch das Ziel haben, die Erreichbarkeit (s. Schutzziele) eines Webservers oder eines Webdienstes zu unterbinden, etwa um finanziellen Schaden oder rufschädigende Auswirkungen zu erzielen. Dieser Angriffsvektor wird als „Denial-of-Service “ bezeichnet. Die Unterbindung der Erreichbarkeit wird durch das Aufbrauchen einer

notwendigen Resource erreicht, etwa Bandbreite, Anzahl der maximalen Verbindungen zum Server oder Rechenleistung. Um dieses Ziel zu erreichen, wird üblicherweise eine hohe Anzahl an Systemen benötigt, die gemeinsam den Angriff ausführen. Aus diesem Grund werden z.B. die beschriebenen Botnetze verwendet, um parallel eine große Anzahl an Anfragen an einen Server zu senden und um die Bandbreite auszulasten. Durch die verteilte Natur dieses Angriffes werden diese Angriffe als „Distributed Denial-of-Service“ Angriffe bezeichnet[12].

Angriffsvektoren auf Webinhalte

Bei diesem Angriffsvektor werden Schwachstellen von Webinhalten ausgenutzt, die den Zugang zu Informationen oder das Ausführen von Schadcodes auf dem unterliegenden Webserver ermöglichen. Dieser Vektor lässt sich weiter unterteilen in „Active Content“ Angriffe und „Content Injection“ Angriffe. Als Active Content werden Webinhalte bezeichnet, die Funktionalitäten zu Webseiten hinzufügen. Dazu zählen etwa JavaScript, PHP, Adobe Flask, Perl, HTML5 und Java. Oft bringen diese Inhalte vollständige Programmierumgebungen mit sich, die eigene Schwachstellen und ausnutzbare Fehler enthalten können. So kann etwa durch das Ausnutzen einer Schwachstelle JavaScript Code in eine Webseite integriert werden. Dieser Schadcode wird ausgeführt, wenn ein anderer Benutzer diese Webseite aufruft um bspw. Malware auf den Computer des Benutzers herunterzuladen. Der Vektor Content Injection wird meistens durch fehlende Überprüfungsroutinen einer Webanwendung geöffnet. Durch das Einfügen von spezifischen Eingaben, die von den gewollten Eingabewerten abweichen, wird etwa das Abrufen von Benutzerdaten einer Datenbank über ein Login-Feld oder die Möglichkeit des Stehlens von Cookies der Benutzersitzung über „Cross-Site Scripting“(XSS) ermöglicht, welches die Möglichkeit der Impersonifizierung des Benutzers gegenüber der Webanwendung eröffnet[12].

2.6 Infrastruktur industrielle Produktionsnetzwerke

Kontext der industriellen Infrastruktur einfügen

2.6.1 Architektur & Anforderungen

Die Architektur eines industriellen Netzwerkes ist darauf ausgelegt, einen Fertigungsprozess zu kontrollieren. Dementsprechend existiert eine hierarchische Struktur, wel-

che die Kontrolle des Prozesses von der Aufgabenannahme bis hin zur Kontrolle der einzelnen Schritte durch SPSen, Aktoren und Sensoren ermöglicht. Zu diesem Zweck wird auf die Disziplin der Prozesskontrolle zurückgegriffen. Die Prozesskontrolle ist eine Ingenieursdisziplin, die sich auf Architekturen, Mechanismen und Algorithmen für die Aufrechterhaltung der Funktionalität eines Prozesses beschäftigt. Allgemein spricht man bei Elementen innerhalb der Prozesskontrolle von industriellen Kontrollsystemen. Diese Kontrollsysteme können in verschiedenen Architekturen abgebildet werden.

Die Automatisierungspyramide

Um das Zusammenspiel und die Anordnung / Strukturierung der Komponenten in einem industriellen Netzwerk zu verstehen, ist es sinnvoll, die grundlegenden Konzepte hinter einer solchen Architektur zu beleuchten. Die Grundlage bildet die industrielle Automatisierung, durch die sich ein industrieller Fertigungsprozess abbilden lässt. Ein automatisierter Fertigungsprozess ist ein Prozess, der ohne das Eingreifen von Menschen ablaufen kann und bei dem ein Material oder ein Produkt mit Energie und Informationen umgewandelt wird [50]. Die industrielle Automatisierung durchzieht verschiedene Hierarchieebenen, die unterschiedliche Aufgaben erfüllen. Für die Darstellung dieser Ebenen kann auf die Automatisierungspyramide (Abbildung 2.5) verwiesen werden.

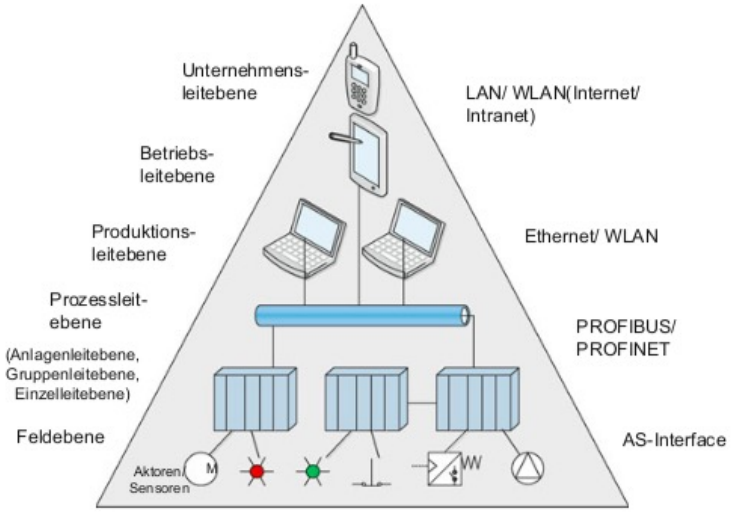

Abbildung 2.5: Automatisierungspyramide [50]

Es wird zwischen den folgenden Hierarchieebenen unterschieden:

- Unternehmensleitebene

- Betriebsleitebene
- Produktionsleitebene
- Prozessleitebene (mit weiteren Unterteilungen)
- Feldebene

In der Unternhemensleitebene werden Prozesse durchgeführt, die die Ausrichtung des Unternehmens bestimmen. Dazu werden u.a. sogenannte Enterprise Resource Planning (ERP) Systeme (z.B. von SAP) für die Verwaltung der zur Verfügung stehenden Resourcen sowie für die Verwaltung der kaufmännischen Prozesse eingesetzt. Die Betriebsleitebene hat die Aufgabe, die eingehenden Aufträge zu verwalten und fristgerecht durchzuführen inklusive der Produktionsplanung und der Verwaltung und Überwachung der täglichen Betriebsprozesse [50]. In dieser Ebene werden MES-Systeme für die Produktionssteuerung und Kontrolle der Fertigung in Echtzeit eingesetzt [69]. Die kurzfristigere Planung des Einsatzes bestimmter Maschinen oder Anlagen ist Teil der darunterliegenden Produktionsleitebene. In dieser Ebene werden sogenannte Supervisory Control and Data Acquisition (SCADA) Systeme eingesetzt.

SCADA-Systeme, siehe Handbuch Quelle

Die Prozessleitebene strukturiert die verschiedenen Fertigungszellen des Produktionsprozesses. Abhängig von der Größe der Anlage kann sie in mehrere Unterebenen unterteilt werden, die wiederum hierarchisch die Anlage, Gruppen von Fertigungszellen und einzelne Fertigungszellen (Zellebene) abbilden. Innerhalb dieser Fertigungszellen wird ein bestimmter Bearbeitungsprozess durchgeführt, der durch verschiedene Bearbeitungsmaschinen durchgeführt wird. Die Bearbeitungsmaschinen werden durch speicherprogrammierbare Steuerungen (SPS) kontrolliert, die ihrerseits Daten aus der Feldebene und übergeordneten Systemen erhalten. Zu der Feldebene werden Aktoren, Sensoren und Anzeigegeräte gezählt, d.h. diese Ebene umfasst die Erfassung und Weiterleitung von (Mess-)Daten sowie die Ausführung von veränderten Stelldaten, die als Reaktion auf die ausgewerteten Informationen der Messdaten veranlasst werden[50].

Prozessleittechnik und Architekturanforderungen

Auf der Grundlage der oben beschriebenen Ebenen, ihre Funktion und ihre Aufgaben wird im folgenden der Begriff der Prozessleittechnik und die damit in Verbindung stehenden Anforderungen an die Komponenten der Architektur beleuchtet. Die Prozessleittechnik ist eine Unterkategorie der Produktionsleittechnik. Sie betrachtet

Aspekte der Mess-, Steuer- und Regelungstechnik sowie die systematische Ordnung von Produktionsprozessen, die Informations- und Kommunikationstechnik sowie die Konzeption für das Erstellung und Betreiben von Leitsystemen. Damit dient die Prozessleittechnik als Schnittstelle zwischen der technischen Umsetzung des Prozesses und der Beobachtung/Bedienung des Prozesses durch den Menschen. Sie bildet das Zusammenspiel zwischen Prozessleitebene und Feldebene ab. Die Grundfunktion der Prozessleittechnik ist die Erfassung, Übertragung, Verarbeitung und Ausgabe von Prozessdaten für die Steuerung und Überwachung des Produktionsprozesses. Dafür müssen abhängig von den Informationen, die aus den verarbeiteten Daten gewonnen werden, Entscheidungen innerhalb eines bestimmten Zeitraumes getroffen werden, um Fehlfunktionen zu vermeiden. Dies wirkt sich allerdings auf die Qualität der getroffenen Entscheidung aus. Aus diesem Grund wird die Entscheidungsfindung hierarchisch auf verschiedene Teilsysteme (Unterebenen) aufgeteilt, die eine geringe Menge an Daten verarbeiten müssen. Durch diese hierarchische Struktur wird die Informationsdichte zu den oberen Ebenen hin höher, die Anforderungen bzgl. Reaktionsschnelligkeit und Zuverlässigkeit/Verfügbarkeit können jedoch im Vergleich zu den unteren Ebenen mit weniger Priorität betrachtet werden. Neben der Verfügbarkeit und Zuverlässigkeit ist auch die Robustheit gegenüber Fehlern eine Anforderung. Auch diese Anforderung ist durch die autarken Teilsysteme in der grundlegenden Architektur eines Fertigungssystems verankert. Im Bezug auf Soft- und Hardwareanforderungen sind die Anforderungen ebenfalls abhängig von der Hierarchieebene. In der Feld- und Prozessleitebene ist die Anzahl der Teilnehmer an der Kommunikation ist geringer und die zu übertragenden Datenmengen vergleichsweise klein, dafür ist es wichtig, dass Anweisungen sehr schnell (in Echtzeit) gesendet und verarbeitet werden können. Dazu werden zusätzlich standardisierte Softwarebausteine verwendet, um die Zuverlässigkeit des Programmablaufs zu maximieren. In den oberen Ebenen, wie etwa der Betriebsleitebene, werden andere Anforderungen wie in einem Unternehmensnetzwerk priorisiert[50].

2.6.2 Geräte

In industriellen Netzwerken werden verschiedene Komponenten und Technologien auf den verschiedenen Ebenen der Prozesskontrollstruktur eingesetzt. Zu den Hauptkomponenten zählen:

- Speicherprogrammierbare Steuerung (SPS)
- Human-Machine Interface (HMI)

- OPC Server
- Sensor & Aktuator
- Weitere Server und Arbeitsstationen der Büroebenen

Speicherprogrammierbare Steuerungen

Eine speicherprogrammierbare Steuerung (SPS) ist ein Gerät, welches Maschinen und Anlagen steuern oder regeln kann. Für die Steuerung als auch für die Regelung einer Maschine wird ein vordefiniertes Programm verwendet, welches auf der vorhanden Firmware ausgeführt wird. Die Firmware beinhaltet die verwendbaren Operanden und Programmbausteine, die für die Programmierung genutzt werden können. Für die Programmierung selbst werden von vielen SPSen standardisierte Bausteine der Norm IEC 61131-3 unterstützt [36]. Das auf die SPS geladenen Programm verarbeitet Informationen, welche von den Eingangsmodulen empfangen werden. An die Eingangsmodule sind üblicherweise ein bestimmter Sensortyp angeschlossen, welcher Informationen über einen bestimmten Aspekt des Prozessschrittes liefert. Basierend auf der Implementierung des Programms, dies kann z.B. eine logische Sequenz, eine Zeitschaltung oder eine arithmetische Operation sein, wird durch diese Informationen ggf. über ein Ausgangsmodul ein Aktor (z.B. ein Motor) aktiviert, der ein Element des Prozessschrittes kontrolliert und Änderungen anstößt oder durchführt. Ein Programm kann mehrere Operanden kombinieren und in Zusammenhang setzen, inklusive Input, Output, Arguments, Counter, Timer und Function Blocks. Eine modulare SPS besteht aus mehreren Basiskomponenten, zu der noch weitere Module optional hinzugefügt werden können. Das Basisset umfasst mindestens einen sogenannten Baugruppenträger, d.h. einen Kasten in dem die Module platziert werden können, eine Stromversorgung, eine Baugruppe, welche die CPU und den Speicher enthält sowie digitale und analoge Ein- und Ausgänge für spezifische Elemente der Feldebene. Dazu können noch weitere Module, wie etwa ein Kommunikationsprozessor (Communication Processor (CP)), weitere Kommunikationsmodule sowie Zähler-, Positionier- oder Regelungsmodule, hinzugefügt werden [36].

Sensoren und Aktoren

Auf der Feldebene werden neben speicherprogrammierbaren Steuerungen Sensoren und Aktoren eingesetzt. Ein Sensor ist ein Element, welches Änderungen einer physikalischen Gegebenheit in seiner Umgebung registriert und in ein elektronisches

Signal umwandelt. Sensoren sind ein wichtiges Element, da Daten aus dem Produktionsprozess mit hoher Zuverlässigkeit und zeitnah an die kontrollierende SPS bzw. die Prozesskontrolle übermittelt werden müssen [40]. Sensoren sind üblicherweise Teil des Schaltkreises eines Sensorgerätes. Abhängig von der Art des Sensors reagiert das Sensorelement auf eine bestimmte physikalische Veränderung. Sensoren unterscheiden sich daher in der Art der wahrgenommenen Veränderung, zu denen in der Automatisierungsindustrie u.a. die folgenden zählen:

- Änderung des Widerstands
- Änderung der Kapazität
- Änderung der Induktivität
- Elektromagnetische Induktion
- Thermoelektrischer Effekt
- Sonar
- Optische Veränderung
- Hall

Durch die technologische Entwicklung stehen heutzutage auch sogenannte Mikrosensoren zur Verfügung. Mikrosensoren, oder auch „Smart Sensors“, integrieren signalverarbeitende Schaltkreise, Analog-Digital-Wandler, programmierbare Speicher und Mikroprozessoren in ein Sensorgerät. Zudem ist es auch möglich, über die Integration einer Antenne eine kabellose Verbindung zu ermöglichen und Sensoren in kabellosen Sensornetzwerken (Wireless Sensor Networks (WSN)) zu strukturieren[40].

Aktoren ermöglichen es, per SPS auf die prozessnahen Maschinen des Produktionsprozesses einzuwirken und den Prozess zu kontrollieren. Aktoren verwenden u.a. Federmechanismen, hydraulische und/oder pneumatische Geräte, Magnetismus oder Termalenergie. Analog zu den Sensoren ist es möglich, Aktoren mit zusätzlichen Elemente auszustatten um u.a. die Kommunikationsmöglichkeiten zu erweitern[40].

Human Machine Interface (HMI)

Ein Human Machine Interface, oder auch Mensch Maschinen Schnittstelle (MMS), ist die Bedienungsschnittstelle zwischen Mensch und Maschine. Sie dient daher als Element für die Steuerung und Beobachtung der Prozesse durch einen Menschen. In der industriellen Fertigung werden HMIs in Form von Panelen, Standalone-PCs

oder auch Thin Clients (d.h. mit der Prozesskontrolle vernetzte Monitore) eingesetzt [37]. Die Anwendung einer bestimmten Hardwareform für die Benutzerschnittstelle wird dabei sowohl durch den speziellen Zweck als auch durch die Einsatzumgebung bestimmt. Die verschiedenen Typen unterscheiden sich dabei u.a. durch die Kommunikationsmodule, Größe und Art des Displays, Performance und angebotenen Zusatzmodulen[68]. HMIs sind oft auf einen speziellen Zweck spezialisiert, der durch den Einsatz und die Konfiguration spezialisierter Softwarekomponenten umgesetzt wird[37].

Server und industrielle PCs

Die bereits im Unterkapitel „Infrastruktur Office“ definierten Server-Komponenten finden auch verschiedene Einsatzmöglichkeiten in industriellen Produktionsnetzwerken. Abhängig von der betrachteten Hierarchieebene werden diese Server für verschiedene Zwecke eingesetzt. Während Server in den oberen Hierarchieebenen Aufgaben des Büroumfeldes, wie etwa die Funktion als Datenbankserver, übernehmen, werden Server der Produktionsleitebene und speziell der Prozessleitebene für Kontroll- und Überwachungsfunktionen des Produktionsnetzes eingesetzt.

Eine Anwendungsmöglichkeit ist es, den Server als SCADA-Server oder -System zu nutzen. SCADA steht für Supervisory Control and Data Acquisition und beschreibt ein wahlweise zentrales oder dezentrales Softwaresystem, welches Informationen der prozessnahen Komponenten (PNK) sammelt, die Informationen zu Analyse- und Überwachungszwecken verarbeitet und steuernd in den Prozess eingreifen kann. Dabei werden die PNKs als sogenannte Datenpunkte betrachtet, die auf Basis von bestimmten E/A-Werten und mathematischer Berechnungen bestimmt werden. Zu den Betriebssystemen, die die Ausführung eines SCADA-Systems erlauben, zählen u.a. DOS-basierte, Windows NT-basierte, Unix-basierte und Linux-basierte Betriebssysteme. Abhängig von der Größe des SCADA-Systems können mehrere tausende bis hunderttausende E/A-Kanäle erfasst werden[38]. In Richtung der oberen Hierarchieebenen kann ein SCADA-System u.a. mit ERP & MES Systemen kommunizieren[1].

Ein SCADA-Server kommuniziert auf verschiedenen Ebenen. Zu Bedien- und Beobachtungszwecken kann ein SCADA-Client mit einem SCADA-Daten-Server kommunizieren per TCP/IP-Schnittstelle. Die Daten-Server kommunizieren je nach Kommunikationspartner entweder über ein bestimmtes Feldbussystem (z.B. Profibus) mit PNKs oder untereinander über die Ethernet-Schnittstelle. Für die Kommunikation zwischen verschiedenen SCADA-Systemen kann das SCADA-Protokoll „Sinaut“

verwendet werden[38].

Der Zugriff von SCADA- und Prozessleitsystemen auf Elemente der Feldebene kann sowohl direkt oder über Prozesskontrollserver erfolgen. Ein Prozesskontrollserver ist ein Server mit zusätzlichen E/A-Schnittstellen, die für die Anbindung herstellerspezifische PNKs genutzt werden können. Um die herstellerspezifischen Datensätze in einem einheitlichen Format an die höheren Hierarchieebenen weiterleiten zu können, bedient man sich der OPC-Schnittstelle. Diese Schnittstelle erlaubt es mithilfe eines spezifischen OPC-Treibers die Datensätze in OPC-Objekte umzuwandeln und OPC-Funktionen zu unterstützen. Die OPC-Schnittstelle dient also der Sammlung von Daten aus einem Netzwerk von Sensoren, Aktoren und SPSen unterschiedlicher Hersteller. Abhängig von der OPC-Spezifizierung basiert OPC auf der DCOM-Schnittstelle von Microsoft und erfordert somit, dass der Server ein Windows-Betriebssystem verwendet. Diese Spezifizierung, das klassische OPC, umfasst Spezifizierungen für die Übertragung von Echtzeitdaten, Alarmen und Events, Zugriff auf historische Datensätze und die direkte Kommunikation zwischen OPC-Servern. Der neuere OPC-Standard wird OPC-UA (Unified Architecture) genannt. Das Ziel dieses Standards ist es u.a., einen Standard unabhängig von DCOM und einem bestimmten Betriebssystemtyp für die zyklisch gesteuerte Kommunikation mit PNK-Netzwerken zu setzen.

In industriellen Produktionsnetzen werden auch PCs in Anlagen verwendet. Diese unterscheiden sich im Vergleich zu herkömmlichen Büro-PCs durch erhöhte Hardware- und Softwareanforderungen. Zu den Hardwareanforderungen zählen u.a. eine gewisse Robustheit gegenüber Einflüssen der Umgebung wie extreme Temperaturen, Schmutz, Feuchtigkeit oder elektromagnetische Felder und Vibrationen, aber auch eine gute Abschirmung des Gehäuses und der Anschlüsse (in Umgebungen, die empfindlich auf Störgrößen dieser Art reagieren). Unabhängig von der Umgebung müssen diese Computer jedoch vor allem ein hohes Maß an Zuverlässigkeit bieten und eine schnelle und einfache Wartung ermöglichen (z.B. die Möglichkeiten eine Festplatte innerhalb von Sekunden auszutauschen). Dies erfordert auch, dass die gleiche Hardware über einen langen Zeitraum betrieben werden kann, da Änderungen der Hardware zu Fehlern bzw. zur Unbrauchbarkeit von gesicherten Speicherabbildern führen können.

Zu den wichtigen Softwareanforderungen gehören vor allem die zuverlässige, fehlerfreie Ausführung sowie die Verarbeitung von Daten in Echtzeit. Zu diesem Zweck wird etwa die Hauptplatine eines PCs durch ein speziell angepasste Platine mit mehr Steckplätzen für Busschnittstellen verwendet.

Ein Fertigungssystem umfasst noch weitere Elemente wie etwa Systeme, die die Produktion von der Produktionsleitebene steuern (Manufacturing Execution System (MES)) oder die Überwachung und Verwaltung auf dieser Ebene ermöglichen über einen Leitstand, der alle Informationen des Fertigungsprozess zentral sammelt und darstellt. Auf der Feldebene existieren zudem noch speziellere Maschinen, etwa für die Qualitätssicherung des Produktes, den Transport des zu verarbeitenden Materials oder für das Austauschen verwendeter Werkzeuge. Da jedoch eine tiefere Beschreibung dieser Elemente für die Kategorisierung der verfügbaren Informationsmenge nicht erforderlich sind, sei an dieser Stelle jediglich darauf verwiesen, dass diese Elemente in diesen Netzwerken existieren [36].

2.6.3 Kommunikationstechnologien

In industriellen Netzwerken herrschen abhängig von der Hierarchieebene unterschiedliche Anforderungen an die Kommunikationswege zwischen den Komponenten. Während in den oberen Hierarchieebenen die Anforderungen der Unternehmensnetzwerke gelten, sind für die Kommunikation in den Bereichen der Prozessleitebene und Feldebene die Anforderungen der Echtzeitfähigkeit und der deterministischen Eigenschaft des Kommunikationssystems oder -protokolls zu erfüllen. Unter der Echtzeitfähigkeit versteht man die Fähigkeit eines Kommunikationssystems, Daten innerhalb weniger Millisekunden oder sogar unterhalb einer Millisekunde zu übertragen. Dies ermöglicht eine schnelle Erkennung von Veränderungen innerhalb des physikalischen Prozesses und schnelle Reaktionen auf diese Veränderungen. Die notwendige Übertragungsgeschwindigkeit ist abhängig von der notwendigen Reaktionszeit. Die Anforderung des Determinismus beschreibt die *Garantie*, dass Daten innerhalb eines bestimmten Zeitraumes übertragen werden. Diese Anforderung ergibt sich daraus, dass die Steuerungen garantiert mit den aktuellen Daten versorgt werden müssen, um Reaktionszeiten einzuhalten und die Sicherheit der Anlage und der Mitarbeiter zu gewährleisten. Durch die technologischen Eigenschaften des Ethernetstandards ist dieser Determinismus nicht gegeben.

Die Kommunikationssysteme in diesen Ebenen verwenden entweder Kommunikationsprotokolle, die den vorhandenen Ethernetstandard um diese Fähigkeiten erweitern, oder verwenden andere Datenbustechnologien. Für die Kommunikation innerhalb der Prozessleitebene zwischen den OPC-Servern und hierarchisch höher liegenden Systemen wie etwa dem SCADA-System oder dem Leitstand wird der Ethernetstandard durch Kommunikationsprotokolle erweitert. Diese werden unter dem Begriff „Industrial Ethernet“ Protokolle zusammengefasst. Für die Kommuni-

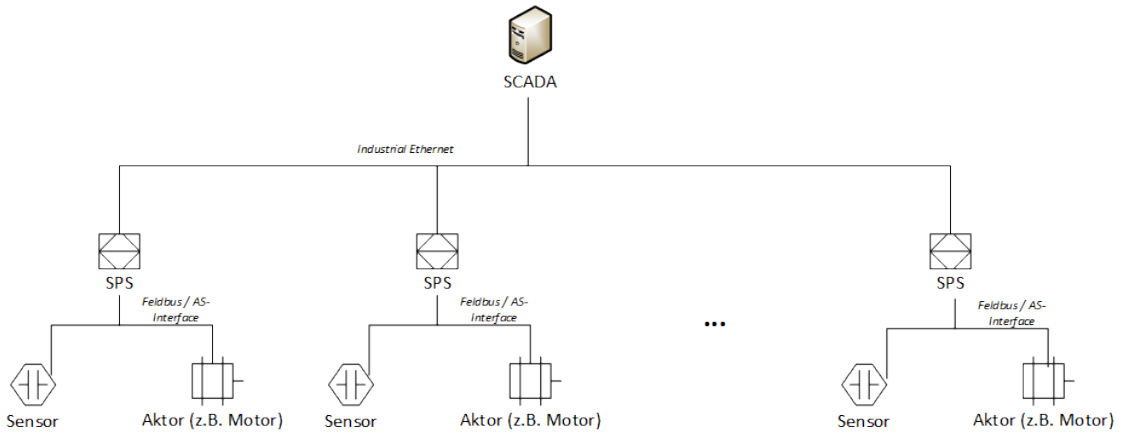

Abbildung 2.6: Kommunikationssysteme in industriellen Fertigungsnetzwerken

kation mit den Steuerungen innerhalb der Feldebene werden historisch bedingt Feldbusse eingesetzt. Allerdings finden auch hier vermehrt Industrial Ethernet Protokolle Einzug. Die Basis für die Verwendung der Feldbusse ist die taktbezogene Arbeitsweise der speicherprogrammierbaren Steuerungen. Für die Steuerung der Sensoren und Aktoren werden wiederum andere Busse wie etwa der AS-I-Bus oder der HART-Bus verwendet. Netzwerke, die aus Sensoren und Aktoren bestehen für die bessere und redundante Überwachung des Prozesses, werden als Aktor-Sensor-Netzwerk bezeichnet.

Feldbus

„Unter Feldbus versteht man ein Bussystem, dass in rauer Umgebung (Feld) eingesetzt wird. Neben besonderen Anforderungen an die mechanische Ausführung sind insbesondere auch Anforderungen an die Robustheit (Störempfindlichkeit) des Datenprotokolls gegeben.“[33].
Ein Bussystem ist ein Kommunikationssystem, bei dem mehrere Kommunikationsteilnehmer die gleiche Datenleitung (Datenbus) verwenden. Um die Kommunikation möglich zu machen, darf nur ein Teilnehmer pro Zeiteinheit als Sender agieren, während die anderen Teilnehmer den Datenbus abhören. Man spricht in diesem Zusammenhang auch von einer Punkt-zu-Gruppe Kommunikation[33]. Für die Freigabe bzw. das Recht eines Teilnehmers, Daten zu senden, existieren verschiedene Modelle, deren Ausprägung abhängig sind von der verwendeten Topologie und der benötigten Funktionalität. Eine Form ist das sogenannte Master-Slave-Prinzip. Bei diesem Prinzip fragt ein autorisierter Teilnehmer (Master) Informationen der anderen Teilnehmer (Slaves) an und erteilt für den Empfang der Daten die temporäre Sendeerlaubnis. Dieses Prinzip wird in vielen Feldbussystemenen verwendet wie etwa

PROFIBUS[33]. Für die Feldebene gilt, dass die Zuverlässigkeit der Kommunikation und die Verfügbarkeit der Daten garantiert werden muss. Ein entscheidender Faktor ist dabei die Behandlung von Kollisionen, welche auftreten, wenn mehrere Teilnehmer gleichzeitig auf dem Datenbus senden. Zu diesem Zweck wird entweder CSMA/CD (Kollisionseerkennung) und CSMA/CA (Kollisionsvermeidung) verwendet. Bei der Kollisionserkennung stoppen alle Teilnehmer das Senden von Daten und der Teilnehmer mit der Sendeerlaubnis wiederholt das Senden der Nachricht. Durch diese Wiederholung kann es zu unterschiedlich großen Zeitfenstern kommen, in denen eine Nachricht gesendet wird. Dieser Zustand kann zu Problemen in einer taktgesteuerten Umgebung wie der Feldebene führen, wenn eine Nachricht innerhalb eines bestimmten Zeitfensters gesendet werden muss, d.h. die Kommunikation muss „deterministisch“ sein. Zu diesem Zweck wird die Methode der Kollisionsvermeidung verwendet, bei der Datentelegramme höherer Priorität Datentelegramme niedrigerer Priorität bei den Empfängern überschreiben, sodass die Daten nicht erneut gesendet werden müssen. In Kombination mit dem Master-Slave-Prinzip, durch das quasi eine temporäre Punkt-zu-Punkt Verbindung zwischen Master und Slave hergestellt wird, kann eine deterministische Kommunikation implementiert werden[33]. Feldbusnetzwerke sind standardisierte, aber auch proprietäre Netzwerke. In den Standards IEC 61158 und IEC 61784 existieren zehn verschiedene Konzepte. Sieben dieser Konzepte stellen eine eigene Protokoll Suite zur Verfügung, die anderen drei Konzepte basieren auf Ethernet Funktionalität. Beispiele für Protokolle sind etwa PROFIBUS oder DeviceNet.

Industrial Ethernet

Zusätzlich zu der Feldbustechnologie existieren in industriellen Netzwerken auch abgewandelte Formen des Ethernet Standards. Im Folgenden wird diese Abwandlungskategorie als „Echtzeit-Ethernet“ bezeichnet. Der intendierte Einsatz dieser Technologie hat mehrere Vorteile. Das Echtzeit-Ethernet ermöglicht die Verbindung zwischen Büro- und Produktionsnetzwerken. So können etwa Echtzeitdaten aus dem Produktionszyklus sowie IT-relevante Daten nahezu zeitgleich und über dasselbe Medium übertragen werden, die Anzahl der Teilnehmer in einem Netzwerk kann erhöht und der gleichzeitige Zugriff auf den Datenbus möglich gemacht werden [22]. Neben der Echtzeitfähigkeit spielt auch die funktionelle Sicherheit, d.h. Schutz gegen Unfälle durch Fehler im Kommunikationsprotokoll, eine wichtige Rolle[63]. Der Ethernet-Standard für sich erfüllt allerdings auf Grund der Verwendung des Prinzips der Kollisionsvermeidung (s. Feldbus) und der damit nicht vorhandenen Deterministik nicht die Echtzeit-Anforderungen der Feld- und Steuerebene. Aus diesem

Grund wurden von verschiedenen Herstellern verschiedene Echtzeit-Ethernet Protokolle entwickelt um die Echtzeit-Anforderungen in den verschiedenen Ebenen zu ermöglichen. Die Klassifizierung der Protokolle kann anhand der vorgegebenen Reaktionszeit und Deterministik nachvollzogen werden[63]:

- „Weiche“Echtzeit (TCP/IP Mechanismen mit ModbusTCP oder PROFINET CBA), keine harten Zeitgrenzen
- Deterministisch (Harte Echtzeit), 1 bis 10ms
- Isochron, 250 μs bis 1ms

Wie für Feldbusse existiert bisher auch kein fester Standard für Industrial Ethernet-Protokolle, wodurch die Kommunikation zwischen Feldbus und Industrial Ethernet mit komplementären Protokollen durchgeführt werden muss. In der in dieser Arbeit folgenden Analyse der industriellen Netzwerke wird die Umsetzung dieser Klassifizierungen am Beispiel von Profinet näher beleuchtet [23].

2.6.4 Kommunikationsprotokolle

Industrial Kommunikationsprotokolle: Wird dieser Unterteil benötigt? (Abhängig von der Analyse und notwendigen Definitionen)

2.6.5 Bekannte Angriffsvektoren

Wie die Systeme in Unternehmensnetzwerken sind auch Systeme der industriellen Netzwerke Angriffen ausgesetzt. Dabei müssen diese Netzwerke nicht zwangsläufig an das globale Netz (Internet) angeschlossen sein, wie etwa das Beispiel Stuxnet belegt, bei dem eine Infektion der Anlage und mehrerer Komponenten über die angeschlossene USB-Speichereinheit eines Mitarbeiters erfolgte.

Eine Liste des Bundesinstitut für Sicherheit in der Informationstechnik (BSI) [30] umfasst in einer Übersicht eine Reihe von Angriffsszenarien:

- Social Engineering & Phishing Angriffe
- Einschleusen von Schadsoftware über Wechseldatenträger und andere externe Hardware
- Infektion mit Schadsoftware über einen Zugriff aus dem Internet oder Intranet
- Einbruch über Schnittstellen, die eine Fernwartung von Systemen ermöglichen

- Menschliches Fehlverhalten und Sabotage
- Internet-verbundene Steuerungskomponenten
- Technisches Fehlverhalten und höhere Gewalt
- Kompromittierung von Extranet und Cloud-Komponenten
- (D)DoS Angriffe
- Kompromittierung von Smartphones im Produktionsumfeld

Diese Szenarien lassen sich in die bereits beschriebenen Kategorien der Angriffsvektoren von Unternehmensnetzwerken einordnen und sind teilweise bedingt durch das Vorhandensein von Komponenten der Unternehmensnetzwerke (wie etwa Webserver oder Benutzer-PCs). Das Ziel des Angreifers ist es, direkt oder indirekt Zugriff auf den Netzwerkverkehr zu erhalten und/oder Netzwerkteilnehmer innerhalb des Fertigungssystems zuzugreifen, um Informationen zu sammeln und Daten zu stehlen, zu manipulieren oder Schäden an der Infrastruktur und damit verbundene Ausfälle des Fertigungssystems zu verursachen. Eine weitere Methode ist die Auslösung eines Notfallprozederes, das die temporäre Abschaltung des gesamten Systems oder eines Teilsystems zur Folge hat.

Kapitel 3

Analyse der Informationsquellen in eines Unternehmensnetzwerkes

SIEM Systeme sind, vor allem in großen Unternehmensnetzwerken (siehe SANS Paper Proactive)

Unternehmensanalyse: Quelle einfügen für Aussage bzgl. SIEM Einsatz (SANS Paper Proactive)!

, zu einem Standard geworden, um Informationen aus Protokollen verschiedener Systeme und anderen Kontextdaten zu gewinnen und diese in Szenarios einzuordnen. Um das Zwischenziel einer Bewertung der aktuellen Informationsmenge eines industriellen Netzwerkes zu erhalten, bietet es sich durch diesen Zustand an, einen Abgleich zwischen dem, in Bezug auf SIEM Technologie, etablierten Informationsstand in Unternehmensnetzwerken und industriellen Netzwerken auszuführen. In diesem Kapitel soll es deshalb um eine akkurate Beschreibung der Informationsmenge eines typischen Unternehmensnetzwerkes gehen, die aus einer Analyse der gängigen Informationstypen und -kategorien erfolgt. Dazu wird im folgenden zunächst der Analyseansatz geschildert, der das Vorgehen schildert und die Analyse nachvollziehbar gestalten soll.

3.1 Verwendete Analysemethode

Die erste Frage für die Wahl des Analyseansatzes stellt sich sowohl in der gewählten Tiefe als auch in einer zielgerichteten Methodik. Das Ziel der Analyse liegt darin, die sicherheitsrelevante Infortmationsmenge zu erfassen. Informationen werden als sicherheitsrelevant betrachtet, wenn diese Informationen genutzt werden können, um die Nutzung eines bestimmten Angriffsvektors erkennen zu können. Da

Angriffsvektoren und die möglichen Folgeschritte sehr zahlreich und spezifisch für ein bestimmtes System oder eine Applikation sein können, ist eine allumfassende Analyse aller Angriffsvektoren nicht möglich. Deshalb wird in dieser Analyse von einem vereinfachten Modell ausgegangen, um eine Einschätzung der möglichen Informationsmenge zu erhalten.

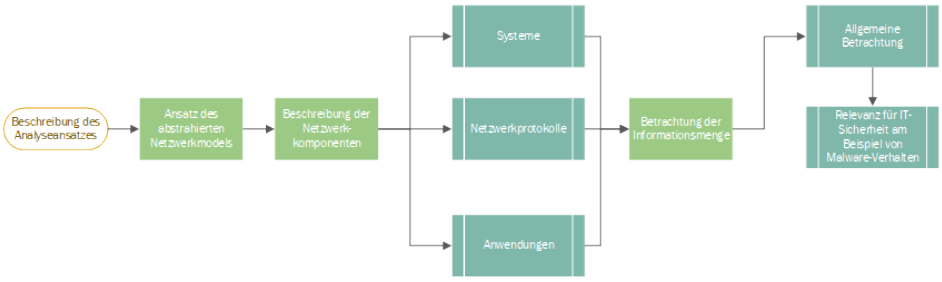

Abbildung 3.1: Analysestruktur (Placeholder)

Basierend auf diesen Praktiken ergibt sich die Durchführung der Analyse: Zunächst werden das benutzte Modell und dessen Komponenten beschrieben. Das Ziel der Beschreibung der Komponenten ist es, einen Überblick über die verfügbaren Informationsquellen zu definieren. Darauf folgend wird auf Basis der Informationsmenge eine Einteilung der Informationen in Kategorien vorgenommen, die eine Basis für den Vergleich der Informationsmengen bilden. Um die Auswahl der Kategorien zu verdeutlichen, wird desweiteren anhand des Beispiels einer Malware-Analyse aufgezeigt, welche Indikatoren auf das Eindringen einer Malware in den verschiedenen Schritten hinweisen können.

3.2 Beschreibung des Modells

Das Ziel des Modells ist es eine repräsentative, vereinfachte Darstellung eines Unternehmensnetzwerkes mit typischen Komponenten für den Betrieb darzustellen. Der Aufbau des Netzwerkes orientiert sich an typischen Elementen, die in einem Unternehmensnetzwerk zu finden sind. Dabei besteht die Notwendigkeit, das Modell in seiner Größe und Komplexität einzugrenzen, um eine Analyse in sinnvollem Maße zu ermöglichen, gleichzeitig aber ein möglichst vollständiges Bild erhalten zu können.

Die Architektur und Einteilung in verschiedene Netzwerkzonen ist dabei von üblichen Sicherheitszonen abgeleitet. Während die Positionierung der Elemente keine besondere Rolle spielt in Bezug auf die ermittelbaren Informationen pro Element, ist jedoch anzumerken, dass auch diese Informationen, z.B. in Form der Auswertung von Netzwerkadressen, einem SIEM-System nützliche Informationen zur Verfügung

stellen können. Aus diesem Grund werden die Zonen beispielhaft in das Modell integriert. Die Architektur des Modells beginnt an seinem „Rand“, dem Zugang zum WAN (Internet), über einen Gateway-Router. Dieser Router ermöglicht die Weiterleitung von Netzwerkpaketen in und aus der angrenzenden Netzwerkzone, der de-militarisierten Zone (DMZ). In der DMZ des Modells ist ein Apache Webserver auf der Basis des Betriebssystems „CentOS (Linux)“ platziert. Dieses Element bietet die Möglichkeit, Datenverkehr über das Hypertext Transport Protocol (HTTP) sowie die verfügbaren Protokolleinträge eines Webservers und eines Linux-basierten Servers zu untersuchen. Alle Elemente werden durch „Managed Switches“ miteinander verbunden.

Die DMZ wird von der nächsten Zone, dem Extranet, durch eine Firewall überwacht, die eingehende initiale Kommunikation blockiert. Firewalls gehören zu den Grundlagen der Sicherheit von Unternehmen und werden häufig platziert, um Datenverkehr in und aus bestimmten Netzwerkzonen zu überwachen. Innerhalb der Netzwerkzone werden zwei Benutzer-PCs eingesetzt. Diese werden platziert, um Abweichungen und andere Benutzerinteraktionen in die erhaltbare Informationsmenge zu integrieren. Die beiden Computer unterscheiden sich wie auch die Server der DMZ im Betriebssystem, sodass ein PC auf Basis des Windows-Betriebssystem enthalten ist.

Als dritte Netzwerkzone wird als „Restricted Area“ bezeichnet. Innerhalb dieser Netzwerkzone wird ein Windows-basierter Server eingesetzt. Auf diesem Server wird ein Microsoft SQL Datenbankserver ausgeführt, um kritische Unternehmensdaten abzubilden. Diese Elemente ermöglichen die Untersuchung eines Windows-basierten Betriebssystems sowie die Verwendung eines Datenbankservers und der dazugehörigen Protokoll-Suite. Zusätzlich ist in dieser Zone als Ergänzung auch der SIEM-Server gesetzt.

Zwischen den verschiedenen Elementen des Netzwerkes werden Informationen ausgetauscht, z.B. für die Abfrage einer Ressource, Herstellung einer Kommunikation oder Übermittlung von Authentifizierungsinformationen. Dieser Austausch wird durch verschiedene Protokolle gesteuert. Der Inhalt der gesendeten Datenpakete (Payload) wird mit verschiedenen Header-Informationen gekapselt. Neben den grundlegenden Header-Daten der Protokolle auf niedrigeren Ebenen (Ethernet, IP, TCP/UDP) sollen in diesem auch verschiedene Protokolle der Ebene 7 betrachtet werden. Die zugehörigen Header-Information sind spezifisch für das entsprechende Protokoll und können z.B. Informationen über den Status der Kommunikation beinhalten. Diese Informationen sind bei der Analyse von Netzwerkdaten nützlich, um den Kontext der ausgetauschten Daten zu verstehen und einen Ablauf der Kommuni-

kation nachzuvollziehen. In dem verwendeten Modell werden verschiedene Protokolle bei der Kommunikation zwischen den verschiedenen Komponenten betrachtet. Die Wahl des Protokolls hängt von der spezifischen Schnittstelle ab. Eine Auflistung der Schnittstellen für das Ansprechen der entsprechenden Komponente werden in der folgenden Tabelle aufgelistet:

Kommunikationsschnittstellen:

- Microsoft SQL Datenbank: SQL Protocol Suite
- Fernzugriff auf ein System: SSH, RDP
- WebServer: HTTP

Die Betrachtung der Informationen von den Netzwerkelementen und der Kommunikation zwischen den Elementen ergibt das Gesamtbild der ermittelbaren Informationsmenge in diesem Modell. Die folgenden Schritte betrachten beide Teile für die jeweils zu untersuchenden Elemente.

3.3 Beschreibung der Systeme

3.3.1 Windows

Das Windows Betriebssystem von Microsoft ist das am Weitesten verbreitete Betriebssystem in der Industrie. Wollen wir die Informationsmenge eines Windowssystemes beschreiben, können wir auf verschiedene Punkte zurückgreifen. In Bezug auf Angriffsanalysen und forensische Analyseverfahren werden zu diesem Zweck drei grundlegende Teile betrachtet: Protokolldateien des Betriebssystems sowie installierter Software, die Windows Registry und ausgeführte Prozesse. Die Überwachung und Dokumentation dieser Bereiche ermöglicht, es ein Bild über den aktuellen Zustand des Systems zu erhalten. Für die Ausführung von Prozessen und Änderungen der Registry kann eine installierte Überwachungssoftware in Form einer lokalen, systemfokussierten Lösung genutzt werden. Mit Hilfe einer solchen Lösung ist es möglich, die Ausführung von Dateien oder das Laden dynamischer Bibliotheken zu überwachen und Sicherheitsereignisse, im Falle der Ausführung/des Ladens aus ungewöhnlichen oder für diesen Zweck gesperrten Verzeichnissen des Dateisystems, zu generieren. Selbiges gilt für Änderungen von Schlüsseln innerhalb der Registry.

Der dritte Teil besteht aus den Protokolldateien des Betriebssystems.

Das Windowsbetriebssystem beinhaltet zwei Kategorien für Protokolldateien: Windows Protokolle und Dienst- und Anwendungsprotokolle. Die Windowsprotokolle beinhalten Ereignisse, die durch das Betriebssystem protokolliert werden. Diese

werden einer von fünf Protokolldateien zugeordnet: *Anwendung*, *Sicherheit*, *Installation*, *System* und *Weitergeleitete Ereignisse*.

Das Anwendungsprotokoll beinhaltet Ereignisse, die von installierten Anwendungen protokolliert werden und etwa Fehler mit Bezug auf das Dateisystem signalisieren. Welche Ereignisse konkret protokolliert werden, wird von den Entwicklern der Anwendung bestimmt.

Das Sicherheitsprotokoll beinhaltet Ereignisse bzgl. sicherheitsrelevanter Elemente wie z.B. Fehlern bei der Anmeldung eines Benutzers oder bzgl. der Ressourcenverwendung bei der Erstellung, Öffnung und Löschung von Objekten. Die Administratoren des Betriebssystems entscheiden, welche Ereignisse dieser Kategorie protokolliert werden.

Das Systemprotokoll enthält Ereignisse, die von Systemkomponenten des Betriebssystems protokolliert werden, während das Setupprotokoll Ereignisse sichert, die bei der Installation von Anwendungen auftreten können [56].

Die zweite Kategorie, Anwendungs- und Dienstprotokolle, beinhaltet Protokolle, deren Ereignisse im Kontext von einzelnen Programmen auftreten und keine systemweiten Auswirkungen haben. Diese Kategorie wird in vier Unterkategorien unterteilt: *Verwaltung*, *Betrieb*, *Analyse* und *Debug*.

Verwaltungsprotokolle enthalten Ereignisse mit Problemen und vordefinierten Lösungspfaden für Administratoren.

Betriebsprotokolle enthalten Ereignisse, deren Daten für die Analyse und Diagnose von auftretenden Problemen sowie für das Auslösen von installierten Werkzeugen oder Programmen genutzt werden können.

Die Analyse- und Debugprotokolle sind standardmäßig deaktiviert und müssen für die Verwendung aktiviert werden. Dabei enthält das Analyseprotokoll Ereignisse zu Programmoperationen und Problemen, die nicht vom Benutzer behoben werden können, während das Debugprotokoll weitere Daten für Entwickler beinhaltet.

Die Protokolldateien sind in einem spezifischen Format geschrieben, sodass eine Anzahl an Feldern vorgegeben ist, die durch den bereitstellenden Service bzw. den bereitstellenden Prozess gefüllt werden können. Die Protokolldateien werden im EVT (alt) bzw. EVTX (neu) Format dargestellt. In der, vom Betriebssystem bereitgestellten, Ereignisanzeige können die Daten sowohl in benutzerfreundlicher Formatierung als auch in einer XML-basierten Form dargestellt werden. Die folgenden Felder werden für diese Protokolle bereitgestellt:

- Quelle
- Ereignis-ID
- Ebene
- Benutzer
- Vorgangscode
- Protokoll
- Aufgabenkategorie
- Schlüsselwörter
- Computer
- Datum und Uhrzeit
- Zusätzliche Felder: Prozess-ID, Thread-ID, Prozessor-ID, Sitzungs-ID, Kernelzeit, Benutzerzeit, Prozessorzeit, Korrelations-ID und relative Korrelations-ID

Bild für Windowslog Eventfelder einfügen zur Verdeutlichung

Diese Felder können durch die jeweilige Quelle und das Betriebssystem mit verfügbaren Daten versehen werden.

Die Quelle gibt die Software(-komponente) oder die Komponente des Betriebssystems an, die das Ereignis protokolliert hat. Die zugehörige Ereignis-ID benennt den Ereignistypen, der z.B. das erfolgreiche Starten eines spezifischen Dienstes darstellt. Weitere Identifikatoren geben spezifischere Informationen über den auslösenden Prozess und zugehörige Elemente an.

Jedes Ereignis wird zu einer bestimmten Kategorie zugeordnet, der Ebene. Die Ebene eines Ereignisses bezeichnet den zugeordneten Schweregrad des Ereignisses. Für alle Protokolldateien stehen die folgenden Ebenen zur Verfügung: *Informationen*, *Warnung*, *Fehler* und *Kritisch*.

Ereignisse der Ebene *Informationen* enthalten Daten über Änderungen an Anwendungen oder Komponenten. Der erfolgreiche Start bzw. die erfolgreiche Beendigung eines Dienstes, sofern dieser nicht die Systemfunktionalität o.ä. gefährdet, seien hier als Beispiel genannt.

Die Ebenen *Warnung* und *Fehler* weisen auf Ereignisse hin, die das Auftreten eines Problems signalisieren. Die Ebene *Warnung* beschreibt Ereignisse, die das Auftreten eines Problems anzeigen, durch das ggf. ein Fehler ausgelöst oder ein Dienst beeinträchtigt werden könnte. Ein Beispiel ist die Verzögerung der Ausführung des Herunterfahrens des Betriebssystems durch einen Prozess, dessen Beendigung verzögert oder nicht durchgeführt werden kann. Die Ebene *Fehler* beschreibt Ereignisse, die potentiell die Funktionalität außerhalb der protokollierenden Quelle beeinträchtigen können. Die Bewertung der Ereignisse dieser Ebene sind somit als schwerwiegender anzusehen als Ereignisse der Ebene Warnung.

Die letzte Ebene *Kritisch* umfasst Ereignisse, die Fehler signalisieren, jedoch nicht automatisch von dem Betriebssystem behoben werden können.

In Protokolldatei *Sicherheit* treten dazu noch zwei weitere Ebenen auf: „Erfolgsüberwachung“ und „Fehlerüberwachung“. Diese Ebenen umfassen Ereignisse, die mit der Anwendung der Rechte des ausführenden Benutzers zusammenhängen. Ereignisse der Ebene Erfolgsüberwachung beinhalten die Dokumentation der erfolgreichen Anwendung der Rechte, Ereignisse der Ebene Fehlerüberwachung beinhalten Fehlermeldungen, die bei der Anwendung aufgetreten sind [55].

Ereignisse der gleichen Quelle und der gleichen Event-ID können abhängig vom Schweregrad in den verschiedenen Ebenen eingeordnet werden.

In der Protokolldatei „Sicherheit“ konnten bei einer Untersuchung eines in der Produktion eingesetzten Windowsservers die mit Abstand größte Zahl an Ereignissen festgestellt werden. Im Sicherheitsprotokoll werden Ereignisse festgehalten, die verschiedene Komponenten bzgl. der Sicherung des lokalen Servers oder Computers als auch Zugriffe auf geteilte Resourcen innerhalb einer Windowsdomäne oder mehrere Domänen betreffen. Im Detail können diese Kategorien einen guten Einblick über die Merkmale der überwachten Elemente durch das Betriebssystems geben:

- Account Logon
- Account Management
- Detailed Tracking
- DS (Directory Service) Access
- Logon/Logoff
- Object Access
- Policy Change

- Privilege Use
- System

Grundsätzlich lassen sich die Unterkategorien bzw. Ereignisse in zwei Bereiche unterteilen: *Domänen- bzw. Directory Service basierte* Ereignisse und *lokale* Ereignisse. Erstere Ereignistypen beziehen sich auf den Zugang zu Domänen und allgemeine Zugriffs- und Rechteverwaltung sowie auf die technisch darunterliegenden Protokolle und Services. Lokale Ereignistypen beziehen sich auf lokale Ereignisse bzgl. des Zugangs zu dem Betriebsystem und die Benutzung von Privilegien und die damit verbundenen Sicherheitsrichtlinien[54].

Zu den Kategorien der Directory Services zählen *Account Logon*, *Account Management* und *DS Access*.

Die Kategorie „Account Logon“ bezeichnet nicht die Authentisierung eines Benutzers an einem Windowsbetriebssystem, sondern die Funktionalität des Kerberosservices. Kerberos ist ein verteilter Authentifizierungsdienst, der für die Anmeldung an einer Windowsdomäne verwendet wird. Daher beziehen sich Ereignisse aus dieser Kategorie auf Operationen bzw. Eigenschaften des Kerberosprotokolls.

Die verwandte Kategorie „Account Management“ bezieht sich auf die Verwaltung von Distribution Groups (Verteilungsgruppen für E-Mail Services) und Security Groups (Zuordnung von Benutzerrechten und Zugriffsrechten auf geteilte Resourcen). Desweiteren enthält diese Kategorie auch Informationen zu der Erstellung von Accounts sowie Zugriffsversuchen auf Passworthashs und Anfragen an die Passwortrichtlinienschnittstelle.

Eine technische Kategorie des Verzeichnisdienstes wird durch „DS Access“ gebildet. Diese Kategorie enthält Ereignisse bzgl. Änderungen, Zugriffen und Replikationen des Verzeichnisdienstes bzw. der im Verzeichnisdienst enthaltenen Daten [54].

Die Kategorien des lokalen Ereignisbereiches sind *Detailed Tracking*, *Logon/Logoff*, *Object Access*, *Policy Change*, *Privilege Use* und *System*.

„Detailed Tracking“ weißt auf Ereignisse bzgl. der Erstellung und Vernichtung von Prozessen hin sowie Aktiviten bzgl. der Data Protection Schnittstelle „DPAPI“ und Anfragen auf die RPC (Remote Procedure Call)-Schnittstelle.

Die Kategorie „Logon/Logoff“ kann als äquivalente lokale Kategorie gesehen werden, da diese Ereignisse bzgl. der Anmeldung/Abmeldung als lokaler Benutzer an einem Betriebssystem gesehen werden kann. Allerdings enthält diese Kategorie auch Ereignisse bzgl. der Nutzung des IPSec Protokolls und die Interaktion eines

Benutzers mit einem Network Policy Server.

Die Kategorie „Object Access“ stellt Ereignisse bzgl. des Zugriffes und der Änderung auf systemrelevante Objekte dar. So sind Ereignisse bzgl. der Verbindung zur Windows Filtering Plattform, darunter Ereignisse der Windows Firewall. Die Windows Filtering Platform ist eine Sammlung aus Schnittstellen und Systemdiensten, die für die Erstellung von Programmen zur FIlterung und Modifikation von Netzwerkdatenverkehr genutzt werden kann. Die Windows Firewall basiert auf dieser Sammlung [57]. Desweiteren werden dieser Kategorie Ereignisse zugeordnet bzgl. Änderungen der Windows Registry Keys, des Component Object Models (COM+), Zugriffe auf das Dateisystem und geteilte Verzeichnisse sowie die Manipulation von Zugriffsoptionen auf Systemressourcen und Änderungen am Certification Service [54].

Die Kategorie „Policy Change“ beschreibt Ereignisse, die mit der Änderungen von Richtlinien-Objekten zusammenhängen. Die entsprechenden Richtlinien gehören zu den Bereichen Authentisierung, Authorisierung, Überwachung, Windows Filtering Platform sowie des MPSSVC (Teil der Windows Firewall, welcher vor nichtautorisiertem Zugriff von Benutzern aus dem Internet oder einem Netzwerk schützt) und anderer Richtlinien (z.B. in Bezug auf kryptografische Operationen) [**?**].

Die Kategorie „Privilege Use“ beinhaltet Ereignisse zu der (nicht-)sensiblen Benutzung von Privilegien im Kontext des Betriebssystems [54]. Schlussendlich zeigen Ereignisse aus der Kategorie „System“ Änderungen am (Sicherheits-)Zustand des Systems sowie der Sicherheitssubsysteme (Local Security Authority und Security Account Manager) [54].

Ereignisse der genannten Quellen können auch, abhängig von der ID, also dem Ereignistypen, in anderen Protokollen wie etwa dem Anwendungsprotokoll oder dem Systemprotokoll aufgeführt werden.

Windows Protokolle: Ggf. weitere Beispiele nennen, kurz beschreiben

Neben den fundamentalen Protokolldateien können weitere Protokolldateien von Applikation erstellt werden. Neben Microsoft-Produkten wie dem Webserver IIS, Microsoft Office oder der Benutzerverwaltung Active Directory können auch Protokolle von Microsoft-fernen Produkten wie z.B. einer Anti-Virensoftware oder proprietäre Netzwerkdienste durch die Applikationen zur Verfügung gestellt werden.

Loggt das Betriebssystem Elemente aus diesem Bereich? Wie funktioniert die Anbindung der Protokolldateien an das Betriebssystem?

3.3.2 Linux

Für die Extraktion der Informationsmenge aus einem Linuxsystem wird die gleiche analytische Basis wie für das Windowsbetriebssystem vorausgesetzt. Die Betriebssysteme unterscheiden sich von ihrem Aufbau und ihren Mechanismen teilweise deutlich, jedoch lässt sich der Grundsatz ähnlich ableiten. Das Ziel ist es, alle verfügbaren Informationen zu erhalten, die bei der Ausführung des Systems entstehen. Dies schließt die Analyse von Protokollen ein sowie die Überwachung der Ausführung von Diensten (Services) und die Ausführung von (System-)Prozessen.

In Bezug auf das Betriebssystem Linux sollen daher die vorhandenen Protokolle sowie grundlegende Elemente, wie etwa zugehörige Informationen zu Diensten und Informationen über die Ausführung von Befehlen, untersucht werden.

Bei Linux unterscheidet man zwischen verschiedenen Distributionen. In Bezug auf Protokolldateien differenziert man in der Literatur bzgl. der Namensgebung zwischen Debian-basierten Distributionen wie etwa Ubuntu und CentOS/RedHat. In Linux existieren vier typische Kategorien für Protokolldateien [51]:

- Application Logs
- Event Logs
- Service Logs
- System Logs

Unter Linux wird das Protokollieren von Systemmeldungen durch „syslogd“ übernommen, den „system logging daemon“. Desweiteren protokolliert der Daemon „klogd“ Ereignismeldungen aus dem Kernel (Betriebssystemkern). Diese beiden Prozesse, die im Hintergrund ausgeführt werden, schreiben Meldungen in Protokolldateien, die sich in dem Unterverzeichnis „syslog“ (Debian-basiert / Ubuntu) bzw. „messages“ (CentOS / RedHat) des Standardverzeichnis für Protokolldateien befinden (/var/log/). Dabei werden die Meldungen als Ereignisse durch Regeln den verschiedenen Protokolldateien zugeordnet, abhängig von ihrer „Facility“ sowie ihrer Priorität.

Für rsyslogd bestehen die folgenden Facilities:

- auth/authpriv: Security/authorization messages (private)
- cron: Clock daemon (crond & atd)
- Daemon Messages from system daemons

- kern: Kernel messages
- local0-local7: Reserved for local use
- lpr: line printer subsystem
- mail: Messages from mail daemons
- news: USENET news subsystem
- syslog: Messages generated internally by system log daemon
- User: Generic user-level messages
- UUCP: UUCP subsystem

Für jedes Ereignis werden, ähnlich der Ebene für Windowsprotokolle, Prioritäten vergeben:

- emerg: System is unusable
- Alert: Action must be taken immediately
- crit: critical conditions
- err: error conditions
- warning: Warning conditions
- notice: normal but significant importance
- info: informational messages
- debug: debugging messages

Basierend auf diesen Parametern werden die Ereignisse in die Protokolldateien geschrieben, deren Namen auf diesen Parametern basieren (z.B. „mail.info“) [2, 13].

Neben den Protokolldateien des syslog Verzeichnisses gibt es noch weitere wichtige Protokolldateien im Verzeichnis „/var/log/ “, die für die Erhebung weiterer Informationen nützlich sein können. Zu den wichtigsten Protokolldateien bzw. -verzeichnissen zählen:

- auth.log (Debian) / secure (CentOS): Ereignisse bzgl. der Authentifizierung von Benutzern
- boot.log: Ereignisse während des Boot-Vorgangs

- dmesg: Nachrichten bzgl. Hardware und Hardwaretreibern
- kern.log: Ereignisnachrichten des Betriebssystemkerns
- faillog: Dokumentation von gescheiterten Login-Versuchen
- cron: Dokumentation der Ausführung und ggf. Fehlermeldungen von Cronjobs

Abhängig von der Verwendung des Servers stehen auch standardmäßig Protokolldateien zu den jeweiligen Servertypen (etwa E-Mailserver, Webserver (typischerweise Apache) oder Datenbankserver (etwa MySQL) zur Verfügung [51, 8, 10, 41].

Beschreibe das Linux Auditing Framework um einen Überblick zu geben, wie Linux Auditing funktioniert und was überwacht wird

3.3.3 Applikationen

Apache Webserver

Der Apache Webserver ist eine freie Software als Teil der Apache Lizenz. Es werden viele Betriebssysteme unterstützt und mithilfe der APR (Apache Portable Runtime) Bibliothek wird eine Verallgemeinerungsschicht zwischen den Webserver und die Systemaufrufe gesetzt, um die individuellen Stärken des Betriebssystems besser nutzen zu können. Der Webserver ist modular aufgebaut und kann um viele Funktionalitäten erweitert werden, u.a. zusätzlich zu der Unterstützung der serverseitigen Skriptsprachen PHP, Perl und Ruby kann ein Modul für Python, Lua, Tcl und .NET geladen werden. Weitere Modulfunktionalitäten sind etwa Verschlüsselungen, Authentifzierung, Proxyfunktionalitäten, WebDAV-Unterstützung oder HTTP-Rewrite. Diese Module können jederzeit aktiviert und deaktiviert werden. Als Ansprechschnittstelle dient die CGI Schnittstelle.

Ein Fehler in der Webserverkonfiguration oder auf dem Webserver aufbauenden Webanwendungen kann potentiell von einem Angreifer provoziert und ausgenutzt werden. Aus diesen Gründen wird von Apache eine Liste an Sicherheitselementen bereit gestellt, die als Anhaltspunkte für die Härtung und Sicherung der Webserver-Installation dienen sollen. Neben Sicherungshinweisen bzgl. der Aktualisierung des Servers und Abwehrmaßnahmen gegen DDoS Angriffe werden auch u.a. die folgenden Elemente genannt:

- Rechte bzgl. des ServerRoot Directories (Wurzelverzeichnis)
- Server Side Includes

- Hinweise zum Umgang mit der CGI Schnittstelle (Generell, Non-Script Aliases, Script Aliases)
- Umgang mit anderen Quellen für dynamische Webinhalte
- Schutz der Systemeinstellungen
- Standardschutz der Serverdateien
- Überwachung der Protokolldateien (Logs)

Im Zuge dieser Sicherheitsbedenken stellt der Webserver verschiedene Protokolldateien -und Funktionalitäten zur Verfügung. Da verschiedene Module unterschiedlich kritische Auswirkungen auf den Webserver haben können, existieren Protokolldateien pro Modul, die individuell anpassbar sind. Neben dem Error-Protokoll und den Modul-Protokollen ist das Access-Protokoll als eine der wichtigste Protokolldateien ausgewiesen, welche Zugriffe auf den Webserver bzw. das Webserververzeichnis protokolliert. Das Verzeichnis und der Zugriff auf das Access-Protokoll werden von der „Custom Log Directive“ verwaltet. Das übliche Protokollformat ist dabei wie folgt strukturiert: „IP-Adresse, Request Teil (falls vorhanden), Zeitstempel, HTTP Kommando, HTTPStatusCode, Größe der Antwort“.

Bild des Apache Default-Logformates einfügen anstatt des Text-Placeholders

Es ist zudem möglich, mehrere Zugriffs-Protokolle zu führen sowie „Conditional Logs“ (Protokolldateien in denen vorkonfiguriert bestimmte Event-Typen nicht protokolliert werden). Innerhalb des Apache Webserver existieren verschiedene *Log-Level*, die die Ausführlichkeit der Eventdokumentation in einer Protokolldatei widerspiegeln. Die Log-Level unterscheiden sich in: „

- emerg: Notfall - das System ist unbenutzbar
- alert: Maßnahmen müssen unverzüglich ergriffen werden
- crit: Kritischer Zustand
- error: Fehlerbedingung
- warn: Warnung
- notice: Normaler, aber signifikanter Zustand
- info: Information
- debug: Debug-Level-Nachrichten

...
Es wird empfohlen, mindestens den Level crit zu verwenden.“[9]

Neben den genannten Protokolldateien gibt es noch ein paar weitere, potentiell signifikante Elemente:

- „mod_ log_ forensic“: ein Modul das forensischen Protokollierungsfunktionalität von Client-Anfragen bietet, mit zwei Einträgen pro Anfrage (davor und danach)
- „PID file“: Speichert die ParentID des Webserverdaemons / -services
- „Script Log“: Protokolliert Ein- und Ausgabe von CGI Skripten

Microsoft SQL Server

Der Kern eines Unternehmens sind Daten über das Unternehmensgeschäft. Die Aufbewahrung, Sicherung und der Zugriff zu diesen Daten ist daher von essentieller Bedeutung. Daher werden diese Daten in Datenbanken abgelegt, die die strukturierte Darstellung und Sicherung von Daten ermöglichen. Der Microsoft SQL Server ist ein relationales Datenbankverwaltungssystem (Relational Database Management System (RDBMS)). Es ermöglicht die Verwaltung mehrerer Datenbanken und steuert den parallelen Zugriff auf eine Datenbank, d.h. das parallele Abrufen und Editiern von Daten durch mehrere Personen. Zu der Verwaltung der Datenbanken werden weitere Funktionalitäten hinzugefügt, die die Analyse der Daten, Integration anderer Dienste und Anwendungen, Reporting und Sicherheit der Datenbanken und des Servers. Zudem existiert eine Client-Anwendung, das Microsoft SQL Management Studio, für den verwaltenden Zugriff auf den SQL Server.

Die Komponenten des Servers werden in zwei Kategorien eingeteilt. Die erste Kategorie umfasst Komponenten für *Business Intelligence* Zwecke, also Komponenten, die bei der Entscheidungsfindung für das Unternehmensgeschäft helfen können. Die zweite Kategorie, *Database Engine* , umfasst Dienste, die für die Operation des Servers notwendig sind, etwa für die Verbindung und den Austausch von Daten über Transact-SQL (T-SQL) Statements [47].

In der Kategorie Database Engine gehören neben der primären Komponente, der Storage Engine, die folgenden Komponenten [48]:

- T-SQL programming interface
- Replication Services

- SQL Server Agent
- High Availability and disaster recovery tools
- SQL Server Integration Services
- SQL Server Management Tools
- Sicherheitssubsystem

Das Sicherheitssubsystem erlaubt den kontrollierten Zugriff zum SQL Server, den verwaltenden Datenbanken, anderen Serverobjekten und Datenbanktabelleneinträgen. Zudem gewährleistet es die Verschlüsselung von Datenbankobjekten und fügt Werkzeuge für das Server Auditing hinzu [49].

Die Server Audit Komponente erlaubt das Verfolgen und Protokollieren von Ereignissen innerhalb der Database Engine. Für den Zweck der Protokollierung werden Objekte (Audit Objects) angelegt. Dies kann sowohl auf der Serverebene als auch auf der Datenbankebene durchgeführt werden. Die protokollierten Ereignisse können entweder in eigens dafür angelegte Dateien, in eine Windows Anwedungsprotokolldatei oder das Windows Sicherheitsprotokoll geschrieben werden. Für die Protokollierung wird unter anderem „Extended Events“ verwendet, ein Überwachungswerkzeug für die Serverperformanz, welches Konzepte des Windows Event Tracing nutzt. Dies erlaubt u.a. die Korrelation von Ereignisdaten innerhalb des SQL Servers. Unter bestimmten Bedingungen ist es zudem möglich, Ereignisdaten des Servers mit weiteren Daten von Anwendungen und dem Betriebssystem zu korrelieren [34].

Die Server Audit Komponente unterteilt die zu protokollierenden Ereignisse in „SQL Server Audit Action Groups and Action“[15].

Diese enthalten, zusammengefasst, die folgenden Elemente:

- Änderungen (Erstellung, Löschung, Veränderung) von
  - Objekten des Servers (z.B. Schemata) oder der Datenbank
  - Zugriffsrechten und Inhaberrechten
  - Zugriffsoperationen
  - Ausführung von T-SQL Statements
- SQL-Aktionen (SELECT, UPDATE, INSERT, DELETE, EXECUTE, RECEIVE, REFERENCE)
- Audit-Aktionen CREATE, ALTER und DROP von Audit Objekten (Server Audit, Server Audit Specification, Database Audit Specification)

3.3.4 Interaktionen der Systeme (und Anwender)

Dieser Abschnitt soll die Interaktion der Netzwerkteilnehmer beschreiben, gehört ggf. in den Architekturteil mit Bild

3.3.5 Netzwerkprotokolle

HTTP (Hypertext Transfer Protocol)

Das Hypertext Transfer Protocol (HTTP) ist ein zustandsloses Protokoll, das zur Übertragung von Daten, meist für das Laden von Webinhalten aus dem Internet, genutzt wird. Die Kommunikation zwischen Client und Server wird in Form eines Nachrichtenaustausches vollzogen, der aus zwei Elementen besteht: Anfrage (durch den Client) und Antwort (durch den Server). Das Protokoll wird als „zustandslos“ bezeichnet, da die aufeinander folgenden Anfragen des Clients unabhängig voneinander versendet werden. Daher ist HTTP u.a. auf ein Transportprotokoll wie TCP auf der Transportebene angewiesen, HTTP selbst wird zur Anwendungsebene zugeordnet.

Die Nachrichtenpakete werden in Kopf (Header) und Rumpf (Body) unterteilt. Der Header enthält Informationen über das gesendete Nachrichtenpaket. Eine typische Anfrage ist wie folgt strukturiert:

<Methode> <URL> <Protokollversion>
<Host>
<Payload>

Dummypräsentation, Bilder von HTTP Request und Response müssen später eingefügt werden

Die Methode stellt die Art der Anfrage dar. Üblicherweise werden entweder die Methoden „GET“ (Anforderung einer Resource per URI (Uniform Resource Identifier)) oder „POST“ (Senden von Daten an den Server zur Verarbeitung) verwendet. Weitere Methoden sind HEAD, PUT, PATCH, DELETE, TRACE, OPTIONS und CONNECT. Die URL gibt den Server an, an den die Anfrage gesendet wird. Das Host-Feld wird genutzt, um mehrere DNS-Namen, die unter der gleichen IP-Adresse erreichbar sind, zu unterscheiden. Der Payload enthält Informationen zu der angeforderten Resource.

Die Antwortnachricht wird in folgendem Format gesendet:

<Protokollversion> <HTTPStatusCode> <Beschreibung>
Server: <Webserverversion> <PHP Version>

Content-Length: <Größe der Resource in Byte>
Content-Language: <Sprachkürzel> (z.B. „de“)
Connection: <Verbindungsstatus>
Content-Type: <Resourcentyp> (z.B. HTML)
<Payload>

Die Antwortnachricht enthält Informationen bzgl. der angefragten Resource sowie Informationen über den liefernden Webserver. Der HTTPStatusCode repräsentiert einen dreistelligen Code, der die erfolgreiche Bearbeitung der Anfrage repräsentiert oder einen Fehlercode anzeigt. Die Codes werden wie folgt kategorisiert:

- 1xx: Informationen
- 2xx: Erfolgreiche Bearbeitung
- 3xx: Umleitung der Anfrage (wenn bspw. eine Resource verschoben wurde)
- 4xx: Clientseitiger Fehler
- 5xx: Serverseitiger Fehler

Secure Shell (SSH)

Secure Shell (SSH) ist ein Netzwerkprotokoll bzw. ein System, welches für die sichere Kommunikation zwischen zwei Computern verwendet wird. Ein SSH-Server erlaubt SSH-Clients Anfragen zu senden, um sich etwa über das Netzwerk auf dem Server zu anzumelden, Daten zu senden oder Kommandos auszuführen. Dabei sollen grundsätzlich die Ziele der Vertraulichkeit der Kommunikation, der Integrität der Nachrichten und sicheren Authentisierung der Kommunikationsteilnehmer sowie der authorisierte Zugriff sicher durchgeführt werden. SSH kann zudem genutzt werden, um weiteren Datenverkehr auf der Basis von TCP/IP zu „tunneln“, d.h. die Nachrichten zu verschlüsseln und verschlüsselt über die vorhandene Sitzung weiterzuleiten. Es existieren verschiedene Versionen dieses Protokolls. Die folgende Beschreibung beschränkt sich auf die Version SSH-2, die als sicherere Variante im Vergleich zu SSH-1, gilt [61].

Für die Kommunikation zwischen dem Client und dem Server werden verschiedene Verschlüsselungsschlüssel für die Herstellung der Kommunikation und für die Dauer der Sitzung (Session) verwendet. Die Herstellung der Kommunikation erfolgt über das Public-Key-Verschlüsselungsverfahren, welches die privaten und öffentlichen Schlüssel des Clients und des Servers nutzt, um eine Verschlüsselungsmethode sowie einen Sitzungsschlüssel für die Verschlüsselung auszuhandeln. Durch diese

Methode wird gleichzeitig die Authentisierung des Benutzers sowie des Servers gegenseitig gesichert [61].

Grundsätzlich gehören zu den Paketfeldern die Gesamtlänge des Paketes, ein gewisser Padding-Bereich, der Payload, ein weiterer zufälliger Paddingbereich und der Message Authentication Code (MAC).

SSH: Bild einfügen für Paketschema?

Das SSH-2-Protokoll wird in verschiedene Layer unterteilt:

- SSH Transport Layer Protocol
- SSH Authentication Layer Protocol
- SSH Connection Layer Protocol

Das SSH Transport Layer Protocol dient als Basis. Mithilfe dieses Layers wird die intiale Verbindung aufgebaut (Prüfung der Server Authentizität, Aushandlung der zu verwendenden Verschlüsselungsmethode, Intialisierung der Sitzung) und somit die grundlegende Funktionalität für die Verschlüsselung der Pakete und die Sicherung der Nachrichtenintegrität mithilfe von kryptografischen Hash-Funktionen [61].

Im Speziellen wird die Kommunikation mit der Nachricht „SSH_ MSG_ KEXINIT“ ausgelöst. Die Felder der Nachricht enthalten neben einem Byte-Array (Cookie) eine Reihe von weiteren Arrays, in denen Listen der unterstützten Algorithmen für Server Host Key, Verschlüsselung, MACs und Komprimierung gesendet werden. Durch einen Abgleich und weitere Protokollregeln ergeben sich die verwendeten Methoden für die Kommunikation. Sollte bei diesem Ablauf ein Fehler oder sonstige Störungen auftreten, wird eine „SSH_ MSG_ DISCONNECT“ Nachricht gesendet. Diese enthält u.a. einen sogenannten „reason code“, der eine Grund für den Verbindungsabbruch angibt.

Darauf folgend kann mithilfe des SSH Authentication Layer Protocol die Authentisierung des Benutzers vorgenommen werden. Für die Authentisierung stehen drei Methoden zur Verfügung („Public Key“, „hostbased“ und „password“). Nach der erfolgreichen Authentisierung besteht im Folgenden die Möglichkeit über das SSH Connection Layer Protocol die weiterreichenden Funktionalitäten von SSH zu nutzen (inklusive Port Forwarding, Remote Access und Remote Program Execution)[61].

Für den Zweck der Authentifzierung des Servers existieren sogenannte „Host Keys“. Diese werden genutzt, um die Authentizität der Serveridentität zu bescheinigen. Mit SSH-2 wird auf einem Server pro Netzwerksockel ein individueller Host Key verwendet. Die Benutzerauthentisierung findet u.a. per Passwort statt. Um die Rechtezuordnung pro Account bzw. pro Server zu konfigurieren, werden Konfigurationsdateien verwendet (etwa /.ssh2/authorization) [61].

SSH: Ist es sinnvoll noch genauer auf die Abläufe einzugehen?

Remote Desktop Protocol (RDP)

Das Remote Desktop Protocol ist ein Protokoll, das die Darstellung und Kontrolle des Bildschirminhaltes eines anderen Computers mit dem Windows Betriebssystem ermöglicht. Es basiert auf dem T-120 Standard der ITU (International Telecommunication Union), der eine Sammlung an Kommunikations- und Anwendungsprotokollen enthält.

Das Protokoll regelt u.a., wie die Dienste und Anwendungen auf der entfernten Windows-Maschine (Terminalserver) angesprochen und verwendet werden können. Dabei ist es möglich, mehrere virtuelle Kanäle zu unterschiedlichen Maschinen zu öffnen und Kommunikations- sowie Präsentationsdaten zu übertragen. Für die Kommunikation notwendige Umgebungsvariablen werden aus den RPC-TCP Einstellungen ermittelt.

Die grafischen Daten werden auf dem Server von einem RDP-zugehörigen grafischen Treiber in Netzwerkpakete verpackt und über das Netzwerk gesendet. Der Client empfängt diese Daten und wandelt sie über Aufruf der Graphic Device Interface (GDI) Programmierschnittstelle in eine Darstellung um. Eingabedaten von Tastatur- und Maus werden über das RDP vom Client auf den Server umgeleitet[58].

Der Ablauf einer RDP-Verbindung kann in verschiedene Teile unterteilt werden. Für den Aufbau der Verbindung werden u.a. die folgenden Schritte durchgeführt. Für die Kommunikation werden in TCP/IP verpackte X.224 Paketeinheiten (Protocol Data Unit, PDU) verwendet:

- Initierung der Verbindung
- Austausch der grundlegenden Basiseinstellungen
- Erstellung eines dedizierten Kommunikationskanals
- Austausch von Sicherheitseinstellungen

- Versendung der Verbindungslizenz des Servers zum Client zu Validierungszwecken während der Verbindung
- Finalisierung der Verbindungsdetails

Für den Abbruch der Verbindung existieren verschiedene Szenarien, die unterschiedlich behandelt werden. Zu diesen zählen die vom Benutzer-initierte Beendigung der Verbindung auf der Clientseite (durch Beenden der RDP-Anwendung) sowie auf der Serverseite (durch Beenden der Sitzung). Als weiterer Fall wird das erzwungene Beenden der Verbindung durch einen Administrator genannt [59].

Microsoft SQL Server Protocol Suite

Für die Kommunikation mit dem Microsoft SQL Server werden verschiedene Protokolle verwendet, die für bestimmte Anwendungszwecke und Funktionen verwendet werden. Zu den Anwendungsbereichen zählen:

- Netzwerkverbindungen und Anwendungsentwicklung
- Verwaltung
- SQL Server Services
  - Master Data Service
  - Reporting Services
  - Analysis Services
- Database Engine
- Complex Event Processing (CEP) Engine

[60] Im Folgenden wird eine Untergruppe dieser Protokolle beschrieben, die für die Netzwerkverbindungen verwendet werden.

Der Verbindungsverlauf mit einer Client-Anwendung erfolgt in den folgenden Schritten [60]:

1. Authentifizierungs Handshake zwischen Client und Server mit einem ausgewählten Schema (SQL Server Authentifzierung oder Windows Authentifzierung)
2. Der SQL Server verifiziert die übermittelten Anmeldedaten.

- Sind die Anmeldedaten korrekt, erfolgt die Bestätigung der Verbindung durch den Server
- Sind die Anmeldedaten nicht korrekt, wird die Verbindung durch den Server abgebrochen und eine entsprechende Rückmeldung gesendet

3. Nach erfolgreicher Anmeldung werden Befehle gesendet
4. Der Server beantwortet diese Anfrage mit einem Ausführungsstatus und einer Antwort
5. Die Verbindung wird durch die Client-Anwendung beendet

Dieser Ablauf gilt für die Verwendung des Native Web Services (SSNWS) Protokolls und des Tabular Data Stream (TDS bzw. SSTDS (TDS v4.2) Protokolls.

Das Native Web Services Protokoll ist ein Netzwerkprotokoll, welches für die Verbindung der Database Engine mit WebService-basierten Anwendungen genutzt wird. Auf der Basis von SOAP 1.1 und 1.2 definiert es Kommunikationslogik und Nachrichtenformate für den Austausch von T-SQL Abfragen.

Das TDS Protokoll, ein Protokoll der Anwendungsebene, regelt die Übertragung von T-SQL Anfragen und Antworten zwischen Client-Anwendungen und Datenbanken auf dem SQL Server. Die Version 4.2 (SSTDS) fügt dem Protokoll weitere Eigenschaften hinzu (u.a. Authentifizierung, Kanalverschlüsselungsverhandlung, Spezifikationen für SQL-Anfragen und RPC-Integration) [60].

Neben diesen Protokollen werden desweiteren die folgenden Protokolle für die Netzwerkverbindungen verwendet:

- Session Multiplex Protocol (SMP): Dieses Protokoll wird für die Kommunikation zwischen Datenbankanwendungen und der SQL Server Database Engine verwendet. Es ermöglicht darüber hinaus Multiplex-Datenbankkommunikation über eine einzelne, stabile Transportverbindung.
- SQL Server Resolution Protocol (SSRP): Dieses Protokoll dient der Namensauflösung von SQL-Serverinstanzen im Netzwerk, sowie für die Auflistung der erreichbaren Serverinstanzen.

[60]

3.4 Analyse des Informationspools

In Folge der Beschreibung der Elemente des zu analysierenden Modells stellt sich die Frage der Relevanz in Bezug auf die Nutzbarkeit für die Erkennung von Sicherheitsvorfällen, u.a. mit Hilfe eines SIEM-Systems. Zu diesem Zweck wird im Folgenden zunächst der Informationspool als Gesamtmenge betrachtet und daraufhin unter dem Aspekt der Analyse von Angriffsszenarien.

3.4.1 Betrachtung der Datenmenge

Die Untersuchung der Netzwerkteilnehmer ergibt, dass auf jedem System eine Art Protokolldatei zur Verfügung steht, die Informationen über Veränderungen des Systems enthält. Diese Informationen umfassen u.a. Daten zu den Zeitpunkten der Veränderung, dem Objekt der Veränderung sowie abhängig von der Protokolldatei und der protokollierenden Software weitere Details wie etwa Verzeichnisdaten. Zudem erlaubt die Integration der Elemente im Netzwerk basierend auf dem einheitlich verwendeten Ethernet-Standard eine Betrachtung der Kommunikationsflüsse innerhalb des Netzwerkes durch die Analyse der Protokolldatein der Firewalls, Switche und Router sowie der zugehörigen Protokolldateien der Server.

Die grundlegenden Protokollierungsmechanismen der Betriebssysteme erlauben eine Übersicht über alle laufenden Prozesse und deren Abhängigkeiten sowie der ausgeführten grundlegenden Operationen (z.B. die Erstellung eines Prozesses). Mit dieser Funktionalität werden zudem Events über Veränderungen von Dateien und Änderungen von Betriebssystemkomponenten (wie etwa Änderungen an der Registry-Komponente des Windows Betriebssystems) dokumentiert. Desweiteren ist mit der vorhandenen Grundlage die Erstellung anwendungsspezifischer Protokolle von auf dem Betriebssystem aufsetzenden Appplikationen möglich. Die Protokollierung der Betriebssysteme umfasst zudem Daten über die Zugriffszeitpunkte, Dauer des Zugriffes und Rechtevergabe für Benutzer der Systeme. Die Limitierung der Dokumentation der Systemveränderungen ergibt sich aus der Verwendung der Protokollierungsplattform durch Entwickler der installierten Anwendungen, etwa in Form der Nutzung der vorhanden Datenfelder und dem Definitionsgrad der Fehlermeldung und anderer Events in Bezug auf die Funktionalität der jeweiligen Anwendung.

Neben der Protokollierungfunktionalität der Betriebssysteme können zusätzlich weitere Überwachungs- und Analysewerkzeuge, wie etwa Endpoint Security Lösungen, verwendet werden, um zusätzliche Informationen durch Auswertung der unter-

liegenden Datenlage sowie der Überwachung definierter Elemente des Systems im Rahmen der vorhanden Resourcen bzgl. Rechenleistung und Zeitkosten zu gewinnen.

Zusätzlich zu der systembezogenen Überwachung und Analyse können die Protokolldateien von Netzwerkkomponenten (Firewalls, Switche, Router) genutzt werden, um Informationen über die Kommunikation innerhalb des Netzwerkes sowie der Kommunikation mit Kommunikationsteilnehmern außerhalb des Netzwerkes nachzuvollziehen und auf Unregelmäßigkeiten zu prüfen. Zu den verfügbaren Daten gehören Daten über den Ablauf der Kommunikation, das Ziel der Kommunikation (etwa die Anfrage und Übermittlung von Datenbankdaten), die Dauer der Kommunikation und auch die Anzahl an Kommunikationsfehlern, Verbindungsabbrüche und andere Störungen.

3.4.2 Kategorisierung der Informationsmenge

Um einen Vergleich der Informationsmengen zu ermöglichen, ist es notwendig, die Informationsmengen so zu definieren, dass Informationen aus beiden Netzwerktypen von verschiedenen Systemtypen miteinander verglichen werden können. Zu diesem Zweck werden die verfügbaren Informationen in Kategorien eingeteilt, um eine konkrete Abgrenzung der Informationsmenge vornehmen zu können. Die Kategorien werden auf Basis des Unternehmensnetzwerkmodels erstellt und repräsentieren die überwachten Bereiche. Dabei wird zwischen Systemen und Netzwerkprotokollen unterschiedenen auf Grund der unterschiedlichen Informationsträger. So ergeben sich wie bereits zuvor beschrieben die Informationen von Systemen aus protokollierten Ereignissen. Während die Kommunikation zwischen verschiedenen Systemen ebenfalls in diese Kategorie fällt, ergeben sich jedoch die Informationen aus Netzwerkpaketen hauptsächlich aus den Informationen, die aus der Zerlegung der Nachrichten gewonnen werden. Daher werden in dieser Arbeit unterschiedliche Kategorien für Systeme und Netzwerkprotokolle angewendet.

Systeme

Für die Erstellung der Systemkategorien werden die beschriebenen Protokolldateien und die damit verbundenen Elemente verwendet.

Die erste Kategorie bildet die Kategorie System. Die Protokolldateien von Windows- wie auch Linux-basierten Betriebssystemen ergeben protokollierten Ereignistypen, die Veränderungen an der Betriebssystemfunktionalität, des Installationsprozesses, Systemanalysen und -fehlern sowie Start und Stopvorgängen, dokumentieren. Zu-

sätzlich werden Ereignisse von Hardwarekomponenten des Systems (z.B. Änderungen am Zustand einer Komponente oder ein Ausfall) protokolliert. Diese Kategorie deckt also Änderungen des Systemzustands im Sinne der Bereitschaft des Systems eine bestimmte Funktion auszuführen, der zugehörigen Hardware und der zugehörigen Software.

Die zweite Kategorie bildet die Datenablage bzw. der Datenspeichers. Die Erstellung, Veränderung und Löschung von Daten sind Aktivitäten, welche kontinuirlich von Prozessen und Anwendern durchgeführt werden. Veränderungen in funktionskritischen Verzeichnissen können Änderungen am Programm-/Prozessablauf und oder den Ausfall einer Software- oder Hardwarekomponente zur Folge haben. Desweiteren liefert die Überwachung Hinweise auf Fremdeinwirkungen und auf potentielle Verhaltensnormalitäten von Benutzern oder Prozessen, wie am Beispiel eines Angriffes mit Hilfe von Schadsoftware im Folgenden noch erläutert wird. Aus diesem Grund wird das Dateisystem durch Anti-Malware-Programme und die Betriebssystemprotokollierung überwacht und die dokumentierten Aktivitäten analysiert.

Die dritte Kategorie bildet die Überwachung von Prozessen und Diensten. Die Erstellung / Terminierung von Prozessen und zughörige Informationen, wie etwa die Identifikationsnummer des Elternprozesses oder die Laufzeit des Prozesses, ermöglichen die Verfolgung von Aktivitäten auf der Systemebene. Die Ausführung von Diensten, etwa zum Start- oder Stopvorgang des Betriebssystems sowie die Eingliederung oder Entfernung von Diensten und die Protokollierung von Dienstaktivitäten stellt einen weiteren Kernpunkt dar, wodurch sich die Kategorie ergibt.

Die vierte Kategorie betrifft den Zugriff auf das System durch einen Benutzer. In diese Kategorie fallen alle protokollierten Ereignisse, die mit (erfolgreichen und nicht erfolgreichen) Zugriffsversuchen, der Verwaltung von Benutzerkonten auf dem System und Informationen zu Anmeldedauer -und zeitpunkt in Verbindung stehen. Die Dokumentation und Analyse dieser Informationen ergeben wichtige Hinweise für präventive Maßnahmen sowie für Reaktionen auf verdächtige Aktivitäten. Daher bildet sich aus diesen protokollierten Elementen die vierte Kategorie.

Neben den Veränderungen des Betriebssystems werden auch Veränderungen und Aktivitäten der aufliegenden Anwendungen protokolliert. Diese werden zum Teil in die Betriebssystemprotokolle integriert, jedoch werden auch anwendungsspezifische Ereignisse wie etwa der Zugriff auf eine bestimmte Resource innerhalb der Anwen-

dung oder Änderungen an den Anwendungseinstellungen bzw. der Zugriff auf diese Einstellungen oder Verzeichnisse durch Anwendungsfunktionen und Anwendungsfehler protokolliert. Aus diesem Grund wird eine eigene Kategorie für diese Ereignistypen erstellt.

Im Folgenden werden die Kategorien und die konkreten Informationsträger für diese Kategorien aufgeführt.

Unternehmensanalyse: Erstelle die Tabelle mit den Systemkategorien mit Latex und ersetze den Placeholder

	System	Dateisystem	Prozesse und Dienste	Benutzer	Anwendungen
Windows	- Installation-Log - System-Log - Analyse&Diagnose Logs - Sicherheit-Log (Kategorie System)	- Sicherheit-Log (Kategorie Object Access)	- Sicherheit-Log (Kategorie Detailed Tracking)	- Sicherheit-Log (Kategorien: Account Logon, Account Management, DS Access, Logon/Logoff, Policy Change)	(Microsoft SQL Server) - Protokollierung in Sicherheit-Log oder SQL Server Logs - Protokollierte Elemente: Änderungen an Serverobjekten und Datenbanken, Zugriffs- und Inhaberrechten, Zugriffsoperationen, SQL- und Audit-Aktionen
Linux	- Logs des "kern" Verzeichnisses - Logs des DaemonMsgs Verzeichnisses - Logs des „syslog" Verzeichnisses - Logs des „dmesg"		- Logs des "cron" Verzeichnisses	- auth.log - faillog.log - Logs des „User" Verzeichnisses	- Error.log - Logs der eingesetzten Module - Access.log - PID file - Script.log

Abbildung 3.2: Placeholder: Darstellung der verfügbaren Informationen der Systeme

Netzwerkprotokolle

Bzgl. der Netzwerkprotokolle ergeben sich aus der Betrachtung zwei grundlegende Kategorien:

Aus den Informationen, die aus den Header-Daten abgeleitet werden können, bildet sich die erste Kategorie. Dies betrifft z.B. die Adressen der Kommunikationspartner, den Nachrichtentyp und anderen Informationen über die Kommunikationsverbindung. Diese Informationen können u.a. verwendet werden, um Hinweise auf die Kommunikation zwischen einem internen System und einem externen Kommunikationspartner nachzuvollziehen oder etwa ungewöhnliche Resourcenanfragen zu erkennen.

Die zweite Kategorie bezieht sich auf Informationen, die sich aus dem Mitschnitt einer Kommunikation mithilfe eines Zwischenelementes (z.B. eines Routers, eines Switches, einer Firewall oder einem NIDS) ergeben. Diese Informationen erlauben es, den Ablauf der Kommunikation zu analysieren und das Verhalten der Kommunikationsteilnehmer abzubilden. Sie werden im Folgenden genutzt, um Unregelmäßigkeiten im Protokollablauf, etwa beim Verbindungsaufbau, zu erkennen.

Die folgende Tabelle zeigt die Kategorien im Bezug auf die in diesem Kapitel analysierten Protokolle:

Unternehmensanalyse: Tabelle aktualisieren und in Latex erstellen

	Header	Verhalten
Ethernet	- MAC-Adressen (Quelle & Ziel) - Frame-Typ	/
IP	- IP-Adressen (Quelle & Ziel) - TTL (Time-To-Live) - Protokollversion - Längeninformationen - Priorität & Service-Typ - Header Prüfsumme	/
TCP / UDP	- Ports (Quelle & Ziel) - Längeninformationen - Prüfsumme - Sequenznummer (TCP) - Bestätigungsnummer (TCP, ACK) - Header Length (TCP)	- Verbindungsaufbau (TCP, Handshake) - Senden & Quittierung des Empfangs (TCP) - Verbindungsabbau (TCP)
HTTP	- Methoden-Typ (z.B. GET) - Protokollversion - Informationen über die angefragte Ressource - Weitere Inhaltsinformationen	- Zugehörigkeiten von Anfrage & Antwort
SSH System	- Gesamtlänge der Nachricht - Message Authentication Code (MAC) - Algorithmenlisten für MACs, Verschlüsselung und Komprimierung (SSH TLP) - Server Host Key (SSH TLP) - Authentisierungsmethode (SSH Auth)	- Protokolltyp (Nachrichtenzugehörigkeit TLP, Authentication oder Connection) - Ablauf des Verbindungsaufbaus & -abbaus (SSH TLP) - Ablauf der Funktionalitätsverwendungen (SSH Connection)
RDP	/	- Verbindungsaufbau - Grund für die Beendigung der Verbindung
MSSQL Suite	- Protokolltyp - Authentisierungsmethoden, Verschlüsselungsmethoden (MSSQL TDS) - SQL-Anfrage Informationen (MSSQL TDS)	- Information über Verwendungszweck durch Protokolltyp (TDS, SMP, SSRP)

Abbildung 3.3: Placeholder: Darstellung der verfügbaren Informationen der Netzwerkprotokolle

3.4.3 Informationsanalyse am Beispiel einer Malwareinfektion

Um die Relevanz der dargestellten Kategorien zu verdeutlichen, wird anhand eines Beispieles, die Analyse von Angriffsszenarien bzgl. der Infektion von Systemen durch eine Malware beschrieben. Die Aufgabe eines SIEM-Systems ist es, wie bereits im Grundlagenkapitel beschrieben, Daten über den Status der Systeme im Unternehmensnetzwerk zu sammeln, zu speichern, auszuwerten und zu verknüpfen. Die zu diesem Zweck erstellten und fortlaufend aktualisierten Regeln basieren u.a. auf der Kenntnis des Verhaltens von Malware.

Bei der Infektion, Ausführung und Verbreitung von Malware auf einem System werden Teile des Systems verändert, um eine möglichst lange Ausführung der Malware zu ermöglichen. Diese Veränderungen werden durch das System an verschiedenen Stellen dokumentiert. Diese Daten werden im Rahmen einer Untersuchung und Überwachung eines Systems als Indikatoren für die Kompromittierung eines Systems

(IoC, „Indicator of Compromise“) bezeichnet. Für die Erkennung dieser Indikatoren wird bspw. für die Überwachung der normale Betriebszustand möglichst präzise definiert sowie eine Analyse des infizierten Systems und der darauf befindlichen Malware durchgeführt, durch die potentielle Erkennungsmerkmale festgestellt und zu Indikatoren zusammengestellt werden. Diese werden von einem Sicherheitssystem wie etwa einem Softwareagenten eines SIEM-Systems oder einer Anti-Malware Software für die Erkennung und/oder Schutzmaßnahmen verwendet.

Der Ablauf eines Angriffsszenarios, bei dem eine Malware für einen oder mehrere Zwecke auf einem System installiert wird, lässt sich grob in die folgenden Schritte einteilen:

1. Erlangen des Zugriffs auf das System über einen Angriffsvektor (z.B. Phishing, infizierte externe Speicher, infizierte Webinhalte)

2. Erstellung eines Zugriffspunktes, Sicherstellung der Persistenz und Verschleierung der Aktivitäten

3. Hinzufügen weiterer Werkzeuge von einem internen oder externen Speicherpunkt

4. Ausführung der Malware auf dem System

5. Ggf. Informationssammlung über weitere Systeme im Netzwerk und Verbreitung der Malware

Quelle [32] beschreibt anhand eines Beispieles die Erstellung von IoCs. In diesem Beispiel wird zunächst der Angriffsvektor in Form einer Phishing-E-Mail analysiert. Durch das Herunterladen und den Versuch der Öffnung einer PDF-Datei im Anhang der E-Mail wurde die erste Komponente der Malware platziert, durch die eine Hintertür geöffnet und dem Angreifer Zugriff auf das System verschafft wurde. Bei der Betrachung des Vorfalles durch den zuständigen Analysten konnte dieser Prozess durch dokumentierte Operationen des Betriebssystems nachvollzogen werden. Im Speziellen wurden Operationen im Dateisystem protokolliert, die die Erstellung der PDF-Datei durch Herunterladen des Anhangs (inklusive des dazugehörigen Zeitstempels, Verzeichnispfades und der Betriebssystemoperation) sowie die Erstellung einer Anwendungsdatei (.exe) zum vermuteten Zeitpunkt des Öffnungsversuches durch den Benutzer zeigen. Im nächsten Schritt konnten protokollierte Änderungen von Einträgen innerhalb der Windows Registry entdeckt werden sowie weitere Spuren in Form von Dateien und Dateisystemoperationen. Diese zeigten das Hinzufügen weiterer Funktionalität in dem Verzeichnispfad „C:\$RECYCLE.BIN“

(verstecktes Verzeichnis des Papierkorbes), etwa eines Werkzeuges für die Durchsuchung des Dateisystems, sowie die Verschleierung der Aktivitäten. Ein typischer Fund konnte etwa das Hinzufügen der Malware in den Autostart-Schlüsseln der Windows Registry verzeichnet werden, durch das die Ausführung der Malware beim Start des Systems sichergestellt werden soll. Zudem wurde der Aufruf des Verzeichnises per Internet Browser dokumentiert sowie weitere Aktivitäten mit dem Ziel, Informationen über Systeme im lokalen Netzwerk zu sammeln.

Andere Quellen und Beispiele der Untersuchung von Malware-Typen, u.a. Keylogger-Werkzeugen, Ransomware und trojanischen Pferden ergeben ähnliche Funde [35]. Darüber hinaus werden diese Indikatoren auch für die Herstellung von Malware-Signaturen verwendet, die von Anti-Malware-Programmen genutzt werden, um Malware-Infektionen zu erkennen und zu verhindern oder einzudämmen.

Diese Beispiele zeigen, dass die Dokumentation jeglicher Systemprozesse sowie die kontinuierliche Auswertung dieser Dokumentationen essentiell sind, um Angriffe durch Malware-Infektionen zu verhindern. Daraus ergibt sich die Relevanz der Informationsmenge für den Einsatz eines SIEM-Systems.

Kapitel 4

Analyse der Informationsquellen in einem industriellen Produktionsnetzwerk

In diesem Kapitel soll in Folge der Analyse der relevanten Informationsmenge in Unternehmensnetzwerken eine Betrachtung des Zustandes in industriellen Produktionsnetzwerken erfolgen. Als industrielles Produktionsnetzwerk werden dafür in diesem Kapitel die Netzwerkstrukturen in der Produktionsleit- und Feldebene betrachtet.

Wie bereits im vorherigen Kapitel ist das Ziel der Analyse, die sicherheitsrelevante Infortmationsmenge zu erfassen. Informationen werden als sicherheitsrelevant betrachtet, wenn diese Informationen genutzt werden können, um die Nutzung eines bestimmten Angriffsvektors erkennen zu können. Da automatisierte Anlagen abhängig vom Industriezweig und selbst von Unternehmen zu Unternehmen zweckgebunden unterschiedlich sind, wird ein stark vereinfachtes Modell ausgewählt, welches die Elemente eines Automatisierungsnetzwerkes zeigt. Da das grundlegende Ziel die Ermittlung der Informationsmenge ist, werden Elemente wie etwa die weite Verteilung / Verbreitung von Fertigungssystemen in diesem Kapitel nur am Rande berücksichtigt.

Äquivalent zu der Betrachtung der Unternehmensnetzwerke werden das benutzte Modell und dessen Komponenten beschrieben. Das Ziel der Beschreibung der Komponenten ist eine grobe Klassifizierung der verfügbaren Informationsquellen der Komponenten. Da in industriellen Netzwerken konkurrierende Standards und viele proprietäre Protokolle und Geräte verwendet werden, muss eine Einschränkung stattfinden. Deswegen fällt die Auswahl für die Modellkomponenten auf populäre Elemente des europäischen Raum. Desweiteren wird basierend auf der individuellen Natur der Programmierungen und Parameteranpassungen auf den zu steuernden

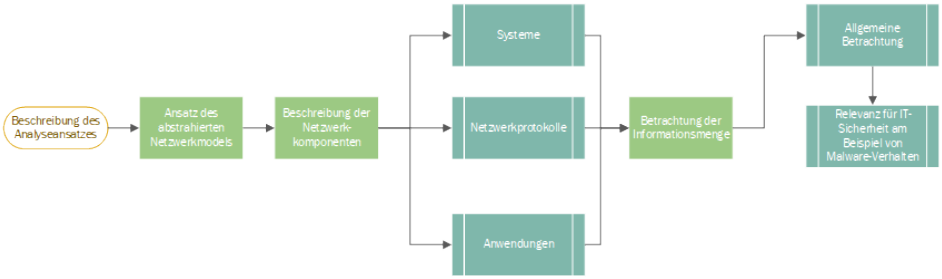

Abbildung 4.1: Analysestruktur (Placeholder)

Prozess der Fokus der Beschreibungen basierend auf der betrachteten Komponente angepasst.

Danach werden die ermittelten Informationsquellen zu den im letzten Kapitel etablierten Informationskategorien zugeordnet. Dies ermöglicht den Vergleich der Informationsmengen.

4.1 Beschreibung der Unternehmensarchitektur (Beispiel)

Industrienetzwerk: Bild zum Model einfügen

Das verwendete Modell bezweckt die Darstellung des Pfades vom Leitsystem (inklusive eines SCADA Systems) zu den Elemente der Feldebene (SPS und Aktoren/Sensoren). Das Leitsystem wird mit einem Industrie-PC in der Funktion eines Servers für die Datenabfrage aus der Feldebene verbunden. Das SCADA System wird durch das Siemens WinCC System dargestellt. Für die Kommunikation zwischen Server und der Kontrolleinheit des Prozesschrittes (SPS) wird das Industrial Ethernet Protokoll PROFINET eingesetzt. Für die Kommunikation mit den Pheripheriegeräten wird das Feldbussystem PROFIBUS genutzt. Als SPS wird eine Siemens S7 eingesetzt.

4.1.1 Systeme

SIMATIC S7-1200 (SPS)

Die SIMATIC S7-1200 wird als Beispiel für eine gebräuchliche SPS herangezogen. Wie im Grundlagenkapitel bereits erläutert, wird die Funktionsweise der SPS durch die Firmware für die unterliegenden Funktionen des Ansprechens der Baugruppen und der Zentralen Einheit (CPU) und das Anwenderprogramm bestimmt. Basierend auf diesem Zustand ist es nicht möglich, eine allgemeine Aussage darüber zu treffen,

welche Daten grundsätzlich protokolliert und verfügbar gemacht werden können. Aus diesem Grund widmet sich diese Beschreibung genauer den verfügbaren Befehlen und Optionen der SPS und der SPS-Programmierung, um eine Abschätzung der verfügbaren Datenmenge und Art der Daten vorzunehmen.

Grundlegend besteht die S7-1200 aus den folgenden Baugruppen:

- Zentralbaugruppe (CPU): Ausführung des Anwenderprogrammes
- Signalbaugruppen: Schnittstellen zu den Aktoren und Sensoren
- Kommunikationsbaugruppen: Erweiterung der SPS um weitere Kommunikationsschnittstellen
- Technologiebaugruppen: Erweiterung der SPS um spezielle Funktionen (z.B. das Messen von Energiedaten)

Basierend auf dieser Auflistung wird für eine gemeinsame Grundmenge im Folgenden auf die Zentralbaugruppe eingegangen [6]. Für diese Beschreibung wurde als Quelle das Betriebshandbuch [5] verwendet.

Grundlegend nutzt man für die Programmierung einer SPS Programmbausteine. Diese dienen dem Zweck der Steuerung und Ausführung des Anwenderprogrammes. Grundsätzlich wird zwischen den folgenden Bausteinen unterschieden:

- Organisationsbausteine (OBs): Diese Bausteine legen die Struktur des Programmes fest und fungieren für den Aufruf und die Ausführung bestimmter Unterprogramme und Alarme
- Funktionsbausteine (FBs) und Funktionen (FCs): Funktionen werden etwa durch die CPU bereitgestellt. Sowohl FBs als auch FCs enthalten ausführbare Programmcodes, der für die Ausführung einer bestimmten Aktion und die Definition von E/A-Parametern im Umgang mit bestimmten Modulen genutzt wird.
- Datenbausteine (DBs): Datenbausteine werden für die temporäre Speicherung von Daten (z.B. Rechnungsergebnissen) verwendet. DBs können auf verschiedenen Ebenen eingesetzt werden, wodurch der Zugriff auf die Daten durch andere Funktionsbausteine oder Programme möglich wird.

Diese Bausteine werden relevant u.a. im Bezug auf die Interaktion mit den Betriebszuständen der CPU. Die Steuerung der Programmausführung durch die OBs

umfasst u.a. die folgenden Bereiche: Strukturierung des Programmzyklus, Strukturierung des Anlaufes, Strukturierung des Aufrufes von Alarmen (Weckalarm, Prozessalarm, Zeitfehler und Diagnosefehler). Dies ist u.a. deshalb relevant, weil Fehlerereignisse in den Diagnosepuffer eingetragen und über eine Schnittstelle abgerufen werden können.

Für die CPU sind die folgenden Betriebszustände definiert:

- STOP: Programm wird nicht ausgeführt, Laden eines neuen Programmes ist hier möglich
- STARTUP: Einmalige Ausführung von OBs, die für den Anlauf des Programmes notwendig sind
- RUN: Ausführung des Programmes in einem sich wiederholenden Zyklus

Basierend auf diesen Zuständen kann die Ausführung eines Anwenderprogrammes gesteuert werden. Treten Fehler auf, können diese Ereignisse u.a. in den Diagnosepuffer geschrieben werden. Ein Diagnoseereignis wird durch Datum, Uhrzeit, Kategorie und Beschreibung des Ereignis beschrieben. Ereigniskategorien sind bspw. Systemereignisse, CPU-Fehler und Modulfehler, die durch Diagnosefunktionen ermittelt werden. Zudem wird jede Änderung des Betriebszustandes protokolliert. Der Diagnosepuffer speichert die neuesten fünfzig Ereignisse.

Für die Parametrierung und den Zugriff auf Daten ist es essentiell, den Datenspeicher und seine Bereiche zu betrachten. Für den Zugriff auf bestimmte Daten ermöglicht die Programmierumgebung für die Steuerungslogik, „STEP7“, das Binden von Datenadressen an symbolische Namen, äquivalent zu Variablen in höheren Programmiersprachen. Der Speicher wird unterteilt in die folgenden Bereiche:

- E: und A: - Prozessabbild der Eingänge und Ausgänge
- M: Merker (Elemente, die für das temporäre Speichern von Zwischenergebnissen genutzt werden)
- DB: Zugriff auf einen Datenbaustein

Bzgl. des Zugriffes auf die Ein- und Ausgänge ist es zudem möglich, direkt auf die physischen Eingänge und Ausgänge zuzugreifen. Das Schreiben von Daten ist jedoch nur für die Ausgänge möglich, da die Eingänge direkt mit den Sensoren und

Aktoren verbunden sind und die enthaltenen Zustände dementsprechend nicht überschrieben werden dürfen.

Informationen und Ereignisse

Im Folgenden wird basierend auf den verfügbaren Anweisungsmöglichkeiten und Funktionen eine Auflistung der potentiellen Daten vorgenommen. Für diese Arbeit wurden die folgenden Kategorien definiert:

- Potentielle Systemereignisse basierend auf Änderungen der Systemkonfiguration
- Fehlerereignisse basierend auf Funktionalitäten
- Fehlerereignisse basierend auf Kommunikationsaktivitäten

Eine Auswahl an potentiell wichtigen Informationen wird in der folgenden Tabelle dargestellt. Die Limitierung durch die maximale Größe des Anwendungsprogrammes und des Speichers wird in diesem Abschnitt noch nicht berücksichtigt.
SIMATIC S7: Tabelle erstellen und Listen ersetzen

- Kommunikation:
  - PROFINET / PROFIBUS Analysefunktionen
    - Verbindungsinformationen (MAC-Adresse, IP-Adresse oder Busadresse, Spezifikationen des Gerätes)
    - Auslesen der Diagnoseinformationen mit GET_ DIAG
    - Abrufen der Betriebszustände von Peripheriegeräten oder Modulen
    - Abrufen von LED-Zuständen
  - Fehler bei Zugriffen auf die Peripherie
- System:
  - Änderung der System- oder Lokalzeiten
  - Weck- und Verzögerungsalarme (Auslöser für die Ausführung eines Programmteils)
  - (De-)Aktivierung von Unterprogrammen (durch Fehlerereignisse von Alarm-OBs)
  - Ziehen / Stecken-Ereignisse beim Entfernen / Hinzufügen von Baugruppen

  - Fehler bei Baugruppenträgern
  - Gemeinsame Fehlercodes für erweiterte Anweisungen bzgl. E/A-Ereignissen und Zugriffe auf DBs
- Funktionalitäten:
  - Diagnosefehler
  - Ereignisse bzgl. des Zugriffsschutz der CPU und bestimmter Bausteine

SIMATIC S7: Auflistung in Tabellenform überführen

Basierend auf diesem Ausschnitt wird das Potential der Verfolgung von funktionalen Fehlern im Prozess sowie bei der Prozessausführung dargestellt. Ein besonderes Augenmerk soll desweiteren auf die Möglichkeit der Datenprotokollierung gelegt werden.

Die S7-1200 und andere Siemens SPSen unterstützen die Erstellung von Protokolldateien im CSV-Datenformat. Für die Einträge von Daten in diese Protokolldateien werden verschiedene Funktionen für die Erstellung, Schreiben und Schließung dieser Dateien bereitgestellt. Jede Protokolldatei kann maximal 256 Einträge speichern. Über interne Ereignisse lässt sich der Zustand und die Erstellung und Schließung der Dateien kontrollieren. Dabei können mehrere Protokolldateien, maximal jedoch acht verschiedene Dateien, gleichzeitig geöffnet sein. Die Gesamtmenge an Protokolldateien wird durch die Größe des Ladespeichers bzw. einer hinzugefügten Speicherkarte definiert. Die Gesamtmenge darf maximal 25% des Speichers einnehmen (basierend auf der Größe des Ladespeichers 250-500 KByte / Speicherkarte bis zu 6 MByte). Der Zugriff auf die Protokolldateien kann entweder über die Aktivierung des Webservers oder den direkten Zugriff auf Protokolldateien über einen Webbrowser („http://<IP-Adresse>/DataLog.html?FileName=<Name der Protokolldatei>“) erfolgen, sofern eine geeignete Verbindung (z.B. über PROFINET) vorhanden ist. Protokolldateien werden im Ladespeicher oder auf einer Speicherkarte abgelegt, dürfen jedoch maximal 25% des jeweiligen Speichers belegen.

4.1.2 Kommunikationsmedien

PROFIBUS

PROFIBUS ist ein serielles Feldbussystem, welches u.a. in der Fertigungs-, Prozess- und Gebäudeautomatisierung verwendet wird. Es ermöglicht die zeitkritische, deterministische Kommunikation von Steuer- und Peripheriegeräten (Aktoren und Sen-

soren) sowie HMIs und anderen Elementen. Das Kommunikationsprinzip basiert auf dem Multi-Master-Prinzip. Dies soll den gemeinsamen Betrieb mehrerer Systeme für den Zweck der Automatisierung, des Engineerings oder der Visualisierung der Anlage, mit den verteilten Pheripheriegeräten der Anlage ermöglichen. Zu diesem Zweck werden die Kommunikationsteilnehmer in zwei Gerätearten unterteilt: Master- und Slave-Geräte. Master-Geräte bestimmen die Kommunikation und senden auf dem Feldbus, wenn sie im Besitz des Tokens sind, welches das Zugriffsrechte auf den Datenbus darstellt. Slave-Geräte hingegen können lediglich empfangene Nachrichten bestätigen oder auf Anfrage des Master-Gerätes Daten senden[18].

Für die Datenübertragung werden drei Versionen von PROFIBUS DP verwendet. DP-V0 dient der zyklischen, DP-V1 und DP-V2 der azyklische Kommunikation.

Fieldbus Data Link: Datenübertragung in PROFIBUS

Für die Zuweisung der Buszugriffsrechte, Verwaltung und Durchführung der Datenübertragung wird das *Fieldbus Data Link* (FDL) Protokoll verwendet. Dieses verbindungslose, Layer-2-Protokoll bestimmt, zu welchem Zeitpunkt ein bestimmter, aktiver Kommunikationsteilnehmer (Master) das Recht erhält, Daten zu senden. Die dadurch etablierte Buszugriffskontrolle (MAC, Medium Access Control) führt die folgenden Aufgaben aus:

- Etablierung des Token-Rings während des Hochfahrens des Systems
- Hinzufügen/Entfernen von Teilnehmern
- Kontrolle der Weitergabe des Tokens

[19]

Der Token-Ring wird durch aufsteigende Busadressen der Master-Geräte realisiert. Die Tokeninformationen werden durch ein spezielles Telegram weitergegeben. Abhängig von der Definition und Größe des Systems muss das Token in einer maximalen Token-Umlaufzeit einmal an jeden aktiven Kommunikationsteilnehmer weitergesendet werden. Dieses Verfahren ermöglicht die Erstellung von Master-Slave-, Master-Master- und Misch-Kommunikationsformen. Die Adressierung wird durch einen Byte-Wert zwischen 0 und 127 kodiert. Dabei werden bestimmte Bereiche für etwa Programmiergeräte, Slaves und Master bestimmt [19].

Desweiteren wird durch eine Unterscheidung der verschickten Telegramme (Nachrichten) eine logische Datensicherung etabliert. Diese wird durch bestimmte Start- und Endzeichen sowie Paritätsbits und Kontrollbytes hergestellt [19].

Für die Übertragung von Daten werden verschiedene Dienste verwendet. Abhängig von der benutzen Version des PROFIBUS DP (V0, V1, V2) Protokolls stehen unterschiedliche Dienste zur Verfügung, dazu zählen u.a.: „

- SDN (Send Data with No acknowledge, alle Versionen)
- SRD (Send and Request Data, alle Versionen)
- CS (Clock Synchronisation, DP-V1 und DP-V2)
- MSRD (Send and Request Data with Multicast Reply, DP-V2)

“[26]

Für die PROFIBUS-Protokolle der Anwendungsebene (DP-Varianten) werden verschiedene Telegramformate bereitgestellt:

- Keine Daten (SD1)
- Daten variabler Länge (SD2)
- Daten fester Länge (SD3)
- Token (SD4)
- Kurzquittung (SC)

[27]

Das Format der Nachricht ist abhängig von dem verwendeten Telegramtyp. Die grundlegenden Header sind:

- Telegrammformat
- Empfängeradresse (außer SC)
- Senderadresse (außer SC)
- Function Code (Telegramtyp, der die Funktion festlegt)

[27]

Zusätzlich zu der Diensten der Datenübertragung existieren die Fieldbus Management (FMA) Dienste für die Verwaltung der Schichten 1 (Übertragungstechnik) und 2 (FDL). Diese Dienste können unterteilt werden in lokale Dienste (bzgl. der Station) und stationsübergreifende Dienste. Zu den lokalen Diensten zählen „Reset“, „Set Value“, „Read Value“, „(R)SAP (De-)Activate“und „Event“. Event ist ein Dienst, der Anwender über Ereignisse oder Fehler in den Schichten 1 und 2 informieren kann. Zu diesen Events zählen:

- Duplicate adress (Master)
- Faulty transceiver (Master)
- Time_put (Master / Slave)
- Not_syn (Master / Slave)
- Out_of_ring (Master
- GAP_event (Master)

Zu den stationsübergreifenden Diensten zählen „Ident (Versionsdaten von Hardware und Software)“, „LSAP Status (Informationen zu einem SAP)“und „Live-List (Liste aller erreichbaren Teilnehmer)“[25].

PROFIBUS DP-V0: Zyklische Kommunikation

PROFIBUS DP dient als Protokoll für den zyklischen Datenaustausch zwischen einer Kontrolleinheit (SPS) und Peripheriegeräten. Bei der Verwendung von DP wird zwischen drei Gerätetypen unterschieden:

- DP-Master 1 (DPM1): Dieser Kommunikationsteilnehmer regelt den zyklischen Datenaustausch (typischerweise SPS oder PC)
- DP-Master 2 (DPM2): Ein zusätzliches Gerät, etwa für Bedienungs- oder Engineering-Zwecke, die Kommunikation erfolgt azyklisch
- Slave: Pheripheriegeräte (Aktoren/Sensoren)

PROFIBUS DP unterstützt Mono- und Multi-Mastersysteme. Ein Multi-Mastersystem besteht aus verschiedenen Subsystemen (Mono-Master, d.h. DPM1 mit mehreren Slaves) und weiteren DPM2-Geräten. Die Kommunikation wird mit Hilfe des FDL Protokolls geregelt [20].

Das Verhalten eines (Sub-)Systems hängt von dem Betriebszustand des DPM1 ab. Es wird zwischen den folgenden Betriebszuständen unterschieden:

- Stop: Kein Datenverkehr zwischen DPM1 und Slaves
- Clear: DPM1 liest Eingangsinformationen der Slaves und schaltet die Ausgänge in einen sicheren Zustand
- Operate: Zyklische Kommunikation zwischen DPM1 und Slaves (Datentransferphase)

Bei der zyklischen Kommunikation im „Operate“-Zustand sendet (schreibt) der Master per Aufruf-Telegramm Ausgangsdaten an den jeweiligen Slave und empfängt (liest) die Eingangsdaten des Slaves. Dieser Vorgang wiederholt sich in dieser Reihenfolge konsequent. Wird ein neuer Slave in das System eingefügt, werden drei Phasen durchlaufen: Parametriesierungs-, Konfigurations- und Datentransferphase. Die ersten Phasen dienen einem Ist-Soll-Abgleich zwischen DPM1 und Slave, bei dem der DPM1 Gerätetyp, Format- und Längeninformationen sowie die Anzahl der Ein- und Ausgänge prüft. Nach erfolgreicher Prüfung geht die Kommunikation in die Datentransferphase über [20].

Neben dem Datenaustausch zwischen einem DPM1 und einem einzelnen Slave besteht auch die Möglichkeit eines Multicasts, bei dem Steuerungsbefehle an einen, eine Gruppe oder alle Slaves des von dem DPM1 verwalteten System gesendet werden können. Diese Funktionalität ermöglicht die Betriebsarten Sync und Freeze. Bei der Sync-Betriebsart werden zunächst die Ausgangszustände der Slaves eingefroren, d.h. sie können nicht mehr verändert werden. Darauf werden Ausgangsdaten vom DPM1 auf dem Slave gespeichert, die Zustände jedoch nicht verändert. Durch einen erneuten Sync-Befehl werden die Zustände überschrieben. Ein Unsync-Befehl beendet den Sync-Betrieb. Bei der Freeze-Betriebsart werden die Eingangszustände der Slaves eingefroren. Die Eingangszustände werden aktualisiert, sobald ein weiterer Freeze-Befehl gesendet wurde. Analog zu Sync wird der Freeze-Betrieb durch den Unfreeze-Befehl beendet [20].

Für den Schutz gegen Fehlparametrierung, d.h. falscher Setzung der Operationsparameter, und Ausfall des Übertragungsmediums, werden für DP-Master als auch DP-Slaves „Data_ Control_ Timer“ ausgelöst, sobald innerhalb eines festgelegten Zeitintervalls keine ordnungsgemäße Kommunikation stattgefunden hat. Im Fall des DP-Masters kann zu diesem Zweck der Parameter „Auto-Clear“ aktiviert (True) sein. In diesem Falle werden die Ausgänge aller Slaves in einen sicheren Zustand geschaltet und daraufhin der Clear-Betriebszustand eingenommen [20].

Neben den Betriebszuständen der PROFIBUS-Master-Geräte nehmen auch die PROFIBUS-Slave-Geräte verschiedene Zustände an: Eingeschaltet, WPRM (Warten auf Parameter), WCFG (Warten auf Konfiguration) und DXCHG (Datenaustausch)[28].

Damit Daten mit einem PROFIBUS-Slave-Gerät ausgetauscht werden können, muss das Gerät in den DXCHG-Zustand wechseln. Um diesen Zustand zu erreichen, wird eine Initialisierungsphase durchgeführt. Zunächst wird die Existenz des Slaves über eine Diagnoseanfrage bestimmt. Daraufhin werden Parameter von dem zuständigen PROFIBUS-Master übertragen und die Konfiguration und Parameter

geprüft. Während dieses Prozesses werden die genannten Zustände durchlaufen, bis der Datenaustausch möglich ist[28].

Für die Analyse von Fehlern stehen Diagnose SAPs (Service Access Points) zur Verfügung. Mit Hilfe dieser Dienste lassen sich Informationen zu Fehlern codiert abfragen. Die Diagnosetelegramme enthalten u.a. drei Statusfelder (je 8 Bit), welche Kommunikationsfehler (Status1), Gerätefehler (Status2) und den Überlauf des Diagnosepuffers (Status3) kodieren[24].

PROFIBUS DP-V1: Azyklische Kommunikation

PROFIBUS DP kann durch PROFIBUS DPV1 erweitert werden. DPV1 dient der azyklischen Kommunikation und ermöglicht die Kommunikation von DPM2-Geräten und Slaves. Die azyklische Kommunikation dient der Übertragung von Bedarfsdaten. Bedarfsdaten sind Parameter (z.B. Grenzwerte) oder Optionen (z.B. Fehlerlisten), die von den Slaves abgerufen werden können. Zu diesem Zweck erweitert DPV1 DP um die folgenden Dienste: „Read“und „Write“[21].

Die azyklische Kommunikation wird parallel zur zyklischen Kommunikation durchgeführt. Der Ablauf kann am Beispiel Read demonstriert werden:

- (Für DPM2: Aufbau einer C2-Verbindung)
- DP-Master sendet den Aufruf an den DP-Slave
- Der DP-Slave bestätigt den Erhalt der Anfrage und beginnt die interne Bereitstellung der Daten (Quittierung)
- DP-Master führt zyklische Prozesskommunikation durch
- Wiederholend:
  - DP-Master sendet Poll-Request
    - Daten stehen noch nicht bereit? DP-Slave quittiert Poll-Request
    - Daten stehen bereit: DP-Slave sendet Daten an Stelle der Quittierung

Die aufgerufenen Betriebsdaten werden über das Modul („Slot“) und den jeweiligen Parameter („Index“) definiert. Soll nur ein Teilwert des Parameters gelesen werden, kann über die Länge („Length“) dies definiert werden. Die Indexnummern und Datentypen werden in PROFIBUS Profilen festgelegt[21].

Für den azyklischen Datenaustausch müssen zunächst bestimmte Parameter gesetzt und als Konfiguration bestätigt werden. Die Unterstützungsmöglichkeit der azyklischen Kommunikation wird in der GSD-Datei des Gerätes definiert.

Die Read- und Write-Dienste von PROFIBUS DP-V1 greifen über einen Index auf die Daten zu. Das zugehörige Telegramm enthält den Telegrammformat („Kurzquittung“-Format), die Slot-Nummer des Moduls, die Indexnummer des Parameters sowie die Länge der Nachricht. Bei einer Antwort werden die Daten angehängt [24].

Da die Leistungsmerkmale der Geräte, d.h. Busparameter und Funktionalitäten (z.B. Anzahl der E/A-Signale und Diagnosemeldungen), abhängig von Gerätetyp und Hersteller unterschiedlich sind, werden GSD-Dateien für die Definition dieser Elemente verwendet. Eine GSD-Datei stellt dabei eine Art elektronische Gerätedatenblatt da und kann während der Konfiguration, z.B. durch ein Projektierungswerkzeug, eingelesen werden. Die GSD-Dateien sind in drei Abschnitte eingeteilt:

1. Allgemeine Festlegungen
2. Master-bezogene Festlegungen
3. Slave-bezogene Festlegungen

Zudem muss in der GSD-Datei eine Identnummer vermerkt sein. Diese Identnummer muss bei der PROFIBUS-Nutzerorganisation beantragt werden und dient der schnellen Identifzierung eines angeschlossenen Gerätetyps durch einen DP-Master [21].

Alarm- und Statusmeldungen von PROFIBUS DP-V1 werden im Vergleich zu V0 umfassender beschrieben. Dazu werden die Alarmmeldungen in verschiedene Typen unterteilt, die Sequenz der Telegramme jeweils quittiert und die Daten in komplexeren Strukturen verpackt. Die GSD Datei legt u.a. fest, welche Alarmtypen von dem Gerät unterstützt werden. Für die Übertragung werden die Alarme in Diagnose-Nachrichten verschickt. Diese bestehen aus dem Typ-Header, dem Alarmtyp, Speicherindex, eine 8bit-Spezifizierung über den Zustand des Alarms sowie Diagnosedaten.

In der Erweiterung PROFIBUS DP-V2 wird die Uhrensynchronisierung durch den DPM1 und isochrone Kommunikation hinzugefügt. Zum Zweck der Uhrensychronisation wurde der Gerätetyp DP-Master 3 (Uhrenmaster) eingefügt.

PROFINET

Profinet ist ein offener Industrial-Ethernet-Standard für die Integration von IT-Systemen in den Automatisierungsprozess auf Basis von TCP/IP. Für die nachfol-

gende Beschreibung wird auf die Systembeschreibung der PROFIBUS Nutzerorganisation (PNO)[17] zurückgegriffen.

Wie bei PROFIBUS werden Kommunikationsteilnehmer in einem Profinet-IO-System in verschiedene Gerätekategorien eingeteilt: IO-Controller (typischerweise eine SPS), IO-Device (Gerät der Feldebene) und IO-Supervisor (Programmiergerät, PC, HMI oder anderes zusätzliches Werkzeug (etwa für Engineering-Zwecke)). Der IO-Supervisor entspricht dem PROFIBUS DP-Master 2.

Zusätzlich werden wie bei PROFIBUS Geräte- bzw. Anwendungsprofile in Form von GSDML-Dateien verwendet. Diese werden durch den Hersteller erstellt und beschreiben Eigenschaften, Leistungsmerkmale und Verhaltensweise der Geräte. Dabei wird zwischen allgemeinen Anwendungsprofilen mit verschiedenen Einsatzmöglichkeiten und spezifischen Anwendungsprofilen unterschieden.

Geräte der Feldebene werden durch ihr Gerätemodell beschrieben, welches die technischen und funktionellen Möglichkeiten darlegt. Das Gerätemodell wird definiert durch den Zugriffspunkt (Device Access Point (DAP)) und die für eine bestimmte Gerätefamilie definierten Module (Baugruppen über die die Kommunikation der Prozessdaten abläuft). Der DAP dient als Schnittstelle für die Ethernet-Kommunikation und Verarbeitungsprogramme. Die Module erlauben die Adressierung der E/A-Daten über die Parameter Slot (Steckplatz/Baugruppe), Subslot (Prozess-Schnittstelle, definiert durch den Hersteller), den Index (gibt den Parameter an) sowie den Application Profile Identifier. Diese Daten werden azyklisch per Read/Write ausgelesen. Der Ausbaugrad eines Anwendungsprofils wird kategorisiert als „kompakt“ (nicht-veränderbar) oder „modular“. Die GSDML-Datei wird auf einer XML-Basis erstellt.

Die Kommunikationswege innerhalb eines PROFINET-Systems werden auf Basis dieser GSDML-Dateien erstellt während der Projektierung. Der Datenaustausch zwischen Kommunikationsteilnehmern wird innerhalb einer Application Relation (AR) durchgeführt, die verschiedene Communication Relations (CRs) spezifiziert:

- Recorded Data CR: Standardkanal für Konfigurationsdaten
- IO Data CR: Kanal für die zyklische Übertragung von Echtzeitdaten
- Alarm CR: Kanal für Alarme und Fehlermeldungen

Sollen mehrere IO-Controller auf die gleichen Daten eines IO-Device zugreifen, muss dieses während der Projektierung angegeben werden. Jeder IO-Controller kann

genau eine AR zu einem bestimmten IO-Device aufbauen. Jedes Feldgerät erhält zudem einen symbolischen Namen als Identifier, welcher als Schlüssel für die Zuordnung von MAC- und IP-Adresse verwendet wird. Die Zuordnung dieses Namens kann über das „Discovery and basic Configuration (DCP)“ Protokoll, alternativ aber auch über die Topologie und Nachbarschaft zu einem bestimmten IO-Controller, durchgeführt werden. Die Zuweisung der IP-Adresse wird per DHCP oder einen herstellerspezfischen Mechanismus durchgeführt. Die verwendbaren Möglichkeiten sind innerhalb der GSDML-Datei definiert.

Für die Adressierung innerhalb des PROFINET-Systems werden eindeutige 48-Bit-MAC-Adressen verwendet. Die MAC-Adresse wird zusammengesetzt aus der Firmenkennung und einer Organization Unit ID (OUI), welche gegen eine Gebühr von IEEE-Departments vergeben werden.

Der Funktionsumfang von Profinet ist flexibel anpassbar abhängig von der notwendigen Funktionalität. Für eine Differenzierung der Funktionalitäten teilt man Profinet in die folgenden Konformitätsklassen ein:

- CC-A: Grundfunktionen
- CC-B: CC-A wird um Netzwerkdiagnose und Topologieinformationen erweitert
- CC-B(PA (Process Automization)): CC-B wird um Systemredundanzen erweitert
- CC-C: Basisfunktionen für Geräte und hardwareunterstützte Bandbreitenreserver für isochrone Echtzeitkommunikation (Basis für taktsynchrone Anwendungen)

Grundfunktionen

Die Grundfunktionen umfassen den zyklischen Austausch von E/A-Daten mit Echtzeiteigenschaften sowie azyklische Kommunikation (Read/Write) von bedarfsorientierten Daten (Parameter, Diagnose) sowie Identifizierungs- und Verwaltungsfunktionen (Identification and Maintenance (I&M)) und eine Alarmfunktionalität für die Signalisierung von Geräte- und Kommunikationsfehlern. Für die Identifzierung eines Gerätes werden die folgenden Elemente verwendet: MAC-Adresse, Hardware-Version, Software-Version, Produkttyp und Hersteller-ID. Neben diesen Informationen lassen sich weiter Informationen auf Geräte- oder Modullevel abrufen[64].

Der zyklische Datenaustausch wird über eine IO Daten CR auf Layer 2 des OSI-Modells in einem Zeitfenster zwischen 250 μs und 512ms durchgeführt. Die gesende-

ten Daten werden durch den Empfänger nicht bestätigt. Die Telegramme enthalten neben den Prozessdaten weitere Informationenen für die Bestätigung der Gültigkeit der Daten, Redundanz- und Diagnosedaten sowie Informationen über den Taktzyklus des Providers. Wie auch bei PROFIBUS werden die Zeitintervalle zwischen den Kommunikation überwacht und Mechanismen ausgelöst, falls der Zeitabstand überschritten werden sollte.

Der azyklische Datenaustausch wird über die Record Data CR auf Basis von TCP/IP durchgeführt. Die Read/Write-Rahmen werden verwendet, um Diagnose- und Identifzierungsinformationen über das Netzwerk und die Kommunikationsteilnehmer abzurufen. Die Diagnosedaten enthalten u.a. mehrstufige Alarmereignisse. Das Zustandsmodell unterscheidet zwischen den Kategorien „gut“, „Wartungsbedarf (etwa bei einem Ausfall der Medienredundanz-Funktionalität)“, „Wartungsanforderung“ und „fehlerhaft“. Zudem wird zwischen Diagnose- und Prozessalarmen unterschieden. Diagnosealarme umfassen Fehler oder Ereignisse, wie etwa das Abziehen oder Aufstecken einer Baugruppe, innerhalb des IO-Devices und im Bezug auf zusammenhängende Komponenten. Prozesslarame werden durch den Anwender definiert und signalisieren etwa die Überschreitung eines Grenzwertes.

Netzwerkdiagnose und -verwaltung (CC-B)

Die Konformitätsklasse (CC) B erweitert PROFINET um weitere Diagnoseinformationen und Topologieinformationen. Diese Informationen werden durch die Erweiterung des Link Layer Discovery Protocols (LLDP-MIB EXT) und innerhalb einer Management Information Base (MIB) gesichert. Die Informationen sind abrufbar über die Verwendung des Netzwerkprotokolls SNMP oder die Verwendung azyklischer Profinet-Dienste.

Die Erweiterung um Informationen der Topologie des Netzwerkes wird durch die Nachbarschaftserkennung ermöglicht. Durch diese Erweiterung tauschen PROFINET-Feldgeräte über LLDP vorhandene Adressierungs-Informationen aus, wodurch der phyikalische Aufbau auf Basis der Port-Nachbarn erschlossen wird. Diese Funktionalität ermöglicht einen Soll-Ist-Vergleich der Topologie sowie die Prüfung des korrekten Anschlusses für den Falls des Austausches eines Gerätes mithilfe eines Software-Werkzeuges.

Für die Verwendung dieser Erweiterung müssen die Switche als IO-Device eingesetzt werden können, um die Übertragung der Alarme über die Alarm CR zu ermöglichen.

Synchrone Echtzeit (CC-C)

Um PROFINET für zeitkritische, deterministische Kommunikation nutzen zu können, müssen weitere Funktionalitäten hinzugefügt werden. Dies umfasst eine netzwerkweite Synchronisierungsfunktion und zeitliche Übertragungsschwankungen („Jitter“) von weniger als 1 ms. Zu diesem Zweck wird eine definierte Bandbreite für das Übertragen dieser Daten reserviert, während die restliche Bandbreite für den restlichen Datenverkehr genutzt wird. Die synchrone Kommunikation erfordert, dass alle Kommunikationsteilnehmer den gleichen Takt verwenden. Dies wird über einen Clock-Master für alle lokalen Taktgeneratoren innerhalb des Taktystems (IRT Domäne) umgesetzt. Zwischen den beteiligten Geräten dürfen keine asynchronen Geräte verwendet werden. Um die Schnelligkeit der Übertragung zu verbessern, verwendet man verschiedene, zusätzliche Mechanismen (Fragmentierung in kleinere TCP/IP-Pakete sowie Dynamic Frame Packing (DFP)).

4.2 Applikationen

4.2.1 WinCC (SCADA / HMI)

Als Beispiel für das SCADA-System wird in diesem Modell das System „WinCC“ von Siemens verwendet. WinCC ist ein pc-basiertes System, das sowohl als eigenständiges SCADA-System als auch als HMI für Prozessleitsysteme eingesetzt werden kann. WinCC wird auf einer modernen Version des Betriebssystem Windows ausgeführt (z.B. Windows 7, 8.1 oder 10 sowie Windows Server 2008 oder 2016). Mit Hilfe dieses Systems ist es möglich, die Durchführung und Überwachung eines Fertigungsprozesses durchzuführen[3]. Dies beinhaltet im Kern die folgenden Funktionalitäten[7]:

- Meldung und Bestätigung von Ereignissen innerhalb des Fertigungsprozesses
- Archivierung von Meldungen und Messwerten
- Protokollierung von gesendeten Prozessdaten und Konfigurationsdaten
- Grafische Darstellung des überwachten Netzwerkes

Hinzu kommen Elemente wie etwa eine Benutzerverwaltung für den geregelten Zugriff auf das System.

Im Bezug auf die für diese Arbeit relevante Informationsmenge ist die Protokollierung und Überwachung des Systems und des Netzwerkes interessant. Da das Betriebssystem bereits im vorherigen Kapitel ausführlich beschrieben wurde, wird im Folgenden das System „WinCC“ selbst betrachtet und die potentiellen Quellen

für Informationen bzgl. des Systems selbst. Das SCADA-System empfängt und sendet Prozessdaten und definierte Bedarfsdaten (Parameterdaten der Steuerungen). Deshalb werden außerdem die Schnittstellen für das Senden und Empfangen der Daten betrachtet.

In WinCC lassen sich die Daten in die folgenden Gruppen unterteilen: Tags, Nachrichten und Alarm Logging.

In WinCC werden „Tags“ als Elemente benutzt, um Daten eines Projektes zu lesen, zu schreiben oder weiterzuleiten. Jeder Tag wird mit einer Datenadresse, einem symbolischen Namen und einem Datentyp sowie weiteren Eigenschaften definiert.

Es wird zwischen Prozesstag und internen Tags unterschieden. Ein Prozesstag ist mit einem bestimmten Parameter eines Automatisierungssystems (z.B. einer SPS) verknüpft und definiert u.a. den Kommunikationstreiber, der die Details der Kommunikation definiert. Diese Verbindung wird genutzt, um Daten auszulesen und zu schreiben. Ein interner Tag wird genutzt, um innerhalb des Projektes Daten zu verwalten und dient zudem als Schnittstelle zur Archivierungsfunktion. Interne Tags, die für die Verwaltung des Projektes notwendig sind, werden Systemtags genannt[4].

Für die Feststellung, Darstellung und Archivierung von Fehlern wird das Nachrichtensystem (Message System) verwendet, um erkannte Fehler visuell darzustellen und an das Archivierungssystem weiterzuleiten. Die Nachrichten werden in drei Teile unterteilt: Systemblocks (Datum, Zeit, Dauer,...), User Text Blocks (vor-definierte Beschreibungen) und Process Value Blocks (Tag-Werte). Zudem wird jede Nachricht einem Typ zugeteilt: Operationsnachrichten (Status eines Prozesses), Fehlernachrichten (Fehler in einem Prozess) und Systemnachrichten (Fehler von anderen Anwendungen). Dokumentierte Ereignisse schließen binäre Ereignisse (Statusänderungen von Tags) sowie „Monitoring Events“ (Archivfehler, Serverfehler, Fehler in der Prozesskommunikation) ein. Für die Archivierung werden Prozessdaten und Nachrichten für den Status der Operation und von Fehlern in einem Microsoft SQL Server gesichert [4].

Beschreibung der WinCC Daten ggf. weiter ausführen? (prüfen)

4.3 Analyse der Informationsmenge

Nach der Beschreibung der Elemente des Modells wird numehr eine Zusammenfassung der gesammelten Informationsquellen erfolgen. Um diese Sammlung der Informationsquellen vergleichbar zu machen, werden sie den Kategorien des Unternehmensnetzwerkes zugeordnet (vgl. Analyse des Unternehmensnetzwerkes). Damit

schafft man eine Vergleichsbasis der Informationsmengen.

4.3.1 Betrachtung der Datenmenge

Die Betrachtung der ausgewählten Elemente zeigt eine Datenmenge, die sich hauptsächlich auf die Dokumentierung und Analyse des Fertigungsprozesses bezieht. Die verfügbaren Diagnosefunktionalitäten durch die genutzten Kommunikationssysteme PROFINET und PROFIBUS sowie die Diagnose und Alarmierungsfunktionalitäten der SPS und die Abfrage entsprechender Parameter-Tags sowie die Möglichkeit der Speicherung und statistischen Analyse der Prozessdaten ermöglichen eine präzise Kontrolle des Prozesses. Basierend auf den verfügbaren Bausteinen und Parametern der SPS und der Protokollierung von Parametern besteht die Möglichkeit genau festzulegen, welche Daten wichtig für die Überwachung eines Prozessschrittes sind. IT-Sicherheitsfunktionen werden in Form von Zugriffskontrollen der Benutzerverwaltung und Passwortschutz auf wichtige Komponenten umgesetzt. Desweiteren bieten sich die bereits diskutieren Möglichkeiten für die Systemüberwachung des SCADA-Servers durch das Windows-Betriebssystem an. Die Protokollierung von SCADA-Ereignissen ist eingeschränkt möglich, eine genauere Betrachtung ist u.U. im späteren Verlauf zu untersuchen.

4.3.2 Kategorisierung der Informationsmenge

Bei der folgenden Kategorisierung der Informationsmenge sollen die beschriebenen Informationsquellen und Informationen in die im vorherigen Kapitel beschriebenen Kategorien übertragen werden. Daher wird auch hier unterschiedenen zwischen Netzwerkprotokollen und Systemen. Die gewählten Zuweisungen und die zugehörigen Begründungen werden nun dargelegt.

4.3.3 Systeme

WinCC (SCADA-System) Zunächst soll hier die SCADA-Anwendung betrachtet werden. Diese wird auf der Basis von Windows ausgeführt. Aus diesem Grund gelten für das zugrundeliegende Betriebssystem die bereits im vorherigen Kapitel beschriebenen Zuweisungen. Die Anwendung selbst ermöglicht die Abfrage von Prozessdaten aus dem Netzwerk sowie die Protokollierung von empfangenen und gesendeten Prozessdaten und Konfigurationsdaten. Diese Informationen im Bezug auf das ausführende System werden der Anwendungskategorie zugeordnet. Die Entscheidung über die Zuordnung zu der Anwendungskategorie basiert in diesem Falle darauf,

dass diese Informationen von WinCC verwendet werden und keine Informationen über das System selbst enthalten. Die internen Tags werden ebenfalls der Anwendungskategorie zugeordnet.

Industrieanalyse Kategorisierung: Sind gesendete und empfangene Konfigurationsdaten wirklich als Anwendungsdaten zu betrachten? Warum? Warum nicht?

Siemens S7-1200 (Speicherprogrammierbare Steuerung)

Die Informationen der beschriebenen SPS lassen sich in mehrere Kategorien einordnen. So lassen sich potentiell verschiedene Informationen über das System selbst ermitteln, inklusive des Betriebszustandes, Systemzeit und Fehlerereignissen aus dem Diagnosepuffer, etwa Nachrichten bzgl. des Hinzufügens oder Entfernens einer Baugruppe, CPU-Fehler und Modulfehler. Daher werden diese Informationen der Systemkategorie zugeordnet.

Alarmereignissen der Zugriffschutzfunktionen können mit der Kategorie *Benutzer* verbunden werden. Diese Informationen bezeichnen ein Ereignis bzgl. des Zugriffs auf abgelegte Daten des Systems. Während diese Informationen auch der Systemkategorie zugeordnet werden könnten, wird hier der Zugriff auf ein „Objekt“ betrachtet, welches Ähnlichkeiten etwa zu Ereignissen des Sicherheitsprotokolls von Windows aus der Kategorie „Object Access“ aufweist.

Weckalarmdaten aus den Organisationsbausteinen werden der Prozesskategorie zugeordnet. Diese geben Informationen über den Start eines Programmes, welches äquivalent zu dem Start eines Prozesses bzw. einer Anwendung in einem Betriebssystem betrachtet wird.

Die Prozessdaten aus den Prozessabbildern, die u.a. vom SCADA-System abgefragt werden können, sowie Fehlermeldungen, die im Rahmen der Programmierung definiert werden können, beschreiben Informationen im Rahmen der Programmausführung. Daher werden diese Informationen der Anwendungskategorie zugeordnet.

Zuletzt sind Informationen zu betrachten, die durch PROFIBUS / PROFINET Analysefunktion abgerufen werden können. Zu diesen Informationen zählen Informatioen über die Systemkonfiguration sowie Daten des Diagnosepuffers und Betriebszuständen von anderen Peripheriegeräten und Modulen. Während die Daten des Diagnosepuffers, wie ausgeführt, verschiedenen Kategorien zugeordnet werden können, werden Informationen bzgl. der Systemkonfiguration der Systemkategorie zugeordnet.

Im Falle der Betriebszustandsdaten von anderen Peripheriegeräten und Modulen ist die Betrachtung nicht eindeutig. Aus der Perspektive des betreffenden Systems sind diese Informationen Systeminformationen, allerdings können diese Informationen auch durch das laufende Programm ausgelesen und verwendet werden, wodurch im Kontext des Programmes diese Informationen als Anwendungsinformationen betrachtet werden könnten. Da jedoch im Kontext dieser Arbeit diese Systeme nicht direkt diskutiert werden, sind diese Informationen zunächst der Anwendungskategorie zuzuweisen. Die Verwendung dieser Informationen kann im Vergleich konkreter diskutiert werden.

Im Folgenden werden die Kategorien und die konkreten Informationsträger für diese Kategorien aufgeführt.

Unternehmensanalyse: Erstelle die Tabelle mit den Systemkategorien mit Latex und ersetze den Placeholder

	System	Dateisystem	Prozesse und Dienste	Benutzer	Anwendungen
Server	(s. Unternehmensanalyse)	(s. Unternehmensanalyse)	(s. Unternehmensanalyse)	(s. Unternehmensanalyse)	- Prozessinformationen des Fertigungssystems - Konfigurationsinformationen der verbundenen Geräte
SPS	- Betriebszustand - Systemzeit - Fehlerereignisse (CPU Fehler oder Modul Fehler) - Systemkonfiguration	/	- Weckalarmereignisse	- Alarmereigniss bei fehlgeschlagener Anmeldung für den Zugriff auf CPU oder Programmierbaustein	- Prozessdaten der verbundenen Aktoren und Sensoren - Betriebszustände der verbundenen Aktoren und Sensoren

Abbildung 4.2: Placeholder: Darstellung der verfügbaren Informationen der Systeme

4.3.4 Netzwerkprotokolle

Zunächst wird das PROFINET Protokoll betrachtet. PROFINET basiert auf der Basis des Ethernet-Standards und abhängig von der verwendeten Konfiguration auf dem Internetprotokoll (IP) und einem Transportprotokoll (TCP oder UDP). Dementsprechend werden die im vorherigen Kapitel beschriebenen Kategorien für diese Protokolle verwendet und der Fokus bei der Kategoriezuweisung auf die Funktionalitäten gelegt. PROFINET unterscheidet verschiedene Konformitätsklassen. Für die Zuordnung werden jedoch zunächst alle potentiell verfügbaren, vorher beschriebenen Funktionen kategorisiert. Bei der späteren, vergleichenden Betrachtung werden die klassenspezifischen Funktionalitäten entsprechend markiert.

Zu der Header-Kategorie werden die Informationen aus den Headern hinzugezählt. Im Falle von PROFINET gehören zu diesen Informationen neben den Ebenen des TCP/IP-Stacks die Operationen Read und Write für Bedarfsdaten, Identifizie-

rung und Verwaltung sowie Alarme für Geräte- und Kommunikationsfehler.

Bzgl. der Verhaltenskategorien werden außerhalb des Verhaltens der unterliegenden Standards die Kommunikation über die verschiedenen Kommunikationskanäle gewertet. Diese Zuweisung deckt einen Bereich ab, in dem potentiell Nachrichten mit für den Kommunikationskanal ungültigen Nachrichtenkopfeinträgen gesendet werden. Diese Information kann potentiell auf eine Manipulation der gesendeten Pakete hindeuten, wodurch die korrekte Verwendung relevant wird.

Industrieanalyse Kategorisierung: Gehören Informationen von Geräteprofildateien bzgl. verwendbarer Dienste auch zur Protokollkategorie Verhalten?

Für PROFIBUS gelten die oben gesetzten Vorbedingungen der verwendeten Standards nicht, da das Kommunikationssystem nicht auf diesen Standards beruht.

Eine PROFIBUS-Nachricht definiert im Nachrichtenkopf verschiedene Informationen. Dazu gehören u.a. die verschiedenen Nachrichtentypen (z.B. Quittung oder Token) sowie Funktionstypen, Telegrammformate und die Empfänger- und Senderadresse. Dementsprechend werden diese Informationen der Header-Kategorie zugeordnet.

Zu der Verhaltenskategorie werden im Bezug auf PROFIBUS der Vorgang, das Einflechten und der Ablauf der azyklischen Kommunikation gewertet. Ein weiteres Verhaltensmerkmal ergibt sich aus dem Prozess des Hinzufügens und Entfernens von Teilnehmern aus dem Kommunikationsbus und weitere definierte Sequenzen (z.B. Slave-Initialisierung oder Alarmsequenzen). Eine interessante Frage stellt sich bzgl. der Änderung eines Betriebszustandes eines DPM1 Gerätes. Da der DPM1 Zustand propagiert bzw. ein entsprechender Ablauf durchgeführt wird, lässt sich diese Information aus einem Mitschnitt der Kommunikation auf dem Kommunikationsbus erhalten und dementsprechend zur Verhaltenskategorie zuordnen. Allerdings lässt sich der Zustand auch von dem entsprechenden Gerät abfragen. Da jedoch die Information zuerst durch den Kommunikationsablauf erhältlich wird und der Betriebszustand zur PROFIBUS Funktionalität gehört, werden die Betriebszustände bzw. Betriebszustandswechsel der Verhaltenskategorie zugeordnet.

Die folgende Tabelle zeigt die Kategorien im Bezug auf die in diesem Kapitel analysierten Protokolle:

Industriesanalyse: Tabelle aktualisieren und in Latex erstellen

	Header	Verhalten
PROFINET	- Ethernet-Informationen (s. Unternehmensanalyse) - IP-Informationen (CC-A, CC-B, CC-B(PA)) - TCP-Informationen (CC-A, CC-B, CC-B(PA)) - Read / Write (verschiedene Abfragen)	- Korrekter Nachrichtentyp in korrektem Kommunikationskanal - Korrekte Addressierung für die richtige Verbindung
PROFIBUS	- Nachrichtentyp - Diensttyp - Busadresse des Empfängers	- Hinzufügen / Entfernen eines Teilnehmers - Azyklisches Kommunikationsverhalten

Abbildung 4.3: Placeholder: Darstellung der verfügbaren Informationen der Netzwerkprotokolle

Kapitel 5

Vergleich der Analysen

In diesem Kapitel wird der Vergleich der Informationsmengen durchgeführt. Als Grundlage für den Vergleich werden die Informationsmengen verwendet, die in den Analysen (Kapitel 3 und 4) ermittelt und in Tabellenform dargestellt sind. Das Ziel dieses Kapitels ist die Erstellung einer Liste, mit Unterschieden zwischen den ermittelten Netzwerkmodellen, die im nächsten Kapitel einer Bewertung unterzogen wird. In diesem Kapitel wird zunächst die verwendete Vergleichsmethode beschrieben, worauf die Anwendung dieser Methode erfolgt und das Ergebnis des Vergleiches dargestellt wird.

5.1 Vergleichsmethode

Die Basisannahme für die Vergleichsmethode ist, dass die Informationsmengen zwischen dem Unternehmens- und dem Industrienetzwerkmodell unterschiedlich sind und Differenzen auftreten. Diese Annahme ergibt sich aus der Tatsache, dass etwa in einem industriellen Netzwerk Systeme verwendet werden, die spezielle Anforderungen des Umfeldes erfüllen müssen und damit bestimmte Limitierungen aufweisen, welche üblicherweise in einem Unternehmensnetzwerk nicht zu finden sind. Dies schließt sowohl Systeme als auch die verwendeten Kommunikationstechnologien ein. Deswegen werden die Analysen angefertigt. Um den Vergleich unterschiedlicher Systemtypen zu ermöglichen, ist es notwendig, eine gemeinsame Basis zu schaffen. Hier wird dies mit der Unterteilung in System und Kommunikationsprotokolle und auch mit der Unterteilung der Informationen in Kategorien erreicht.

Diese Basis ermöglicht einen mehrschichtigen Vergleich. In einem ersten Schritt wird untersucht, ob für jede Kategorie jeweils Informationsquellen vorhanden sind. Durch die Informationsquellen des Unternehmensnetzwerkes (hauptsächlich in Form

von Protokolldateien) ergeben sich gewisse Unterkategorien, für die die Existenz äquivalenter Informationsquellen im Industrienetzwerkmodell geprüft werden. Im letzten Schritt folgt ein Vergleich dieser äquivalenten Informationsquellen. In diesem werden Unterschiede dieser äquivalenten Informationsquellen herausgearbeitet und Unterschiede notiert.

5.2 Vergleich

Die beschriebene Methodik wird zunächst für den Vergleich zwischen Server und SPS angewandt. Das Ziel ist es, die Unterschiede zwischen den beiden Systemtypen zu zeigen. Zusätzlich werden desweiteren die Anwendungen verglichen und Unterschiede aufgezeigt. Es folgt der Vergleich zwischen die Ethernet/IP/TCP-basierte Kommunikation und der Feldbuskommunikation am Beispiel von PROFIBUS. Dadurch werden die grundlegenden Unterschiede herausgestellt. Desweiteren wird PROFINET mit seinen verschiedenen Konfigurationen und Diensten mit den Protokollen der Anwendungsebene verglichen.

5.2.1 Systeme

Im Folgenden werden die Betriebssysteme mit dem SPS-Modell anhand der ermittelten Informationsquellen verglichen.

Kategorie: System

Bei der Betrachtung der Kategorie „System“ sind in allen drei Systemen Informationen über deren Zustand verfügbar. Im Detail protokollieren sowohl die Betriebssysteme als auch die SPS Ereignisse bei Start- und Stop-Vorgängen des Systems. Während jedoch bei den Betriebssystemen die einzelnen Schritte dokumentiert werden, ist die Dokumentation des Betriebszustandwechsels der SPS von STOP zu Anlauf zu RUN limitiert. Die dokumentierten Ereignisse im Diagnosepuffer beschränken sich auf protokollierte Ereignisse der Anlauf-OBs (Standardereignisse) und der damit verbundenen Anzahl der Anlauf-OBs sowie Fehlerereignisse, die während des Anlaufes auftreten.

Bei Hardwarefehlern werden Fehlerereignisse protokolliert, die durch das Betriebssystem erkannt werden (etwa bei Schnittstellen-Fehlern) sowie Fehlern, die durch die Gerätetreiber an das Betriebssystem gemeldet werden. Daraus ergibt sich, dass die dokumentierten Fehler beschränkt sind durch Treiberdefinition und Be-

triebssystemkennwerte. Äquivalent werden im Diagnosepuffer der SPS von der CPU erkannte Fehler bzgl. der CPU selbst, des Gerätes, eingebauter Module und angeschlossener Peripheriegeräte dokumentiert. Die Alarme der (Pheripherie-)Geräte sind in der entsprechenden GSD Datei dokumentiert. Desweiteren dokumentiert die SPS auch Kommunikationsfehler zu den Pheripheriegeräten (etwa Kurzschlüsse oder Drahtbrüche).

Änderungen an Betriebssystemkomponenten (inklusive Verzeichnissen, Softwarekomponenten und Dateien) inklusive Sicherheits(sub-)systemen werden in System- bzw. Sicherheitsprotokollen dokumentiert. Erweiterte Ereignisse etwa bzgl. Änderungen an der Windows Registry oder System Daemons (Linux) lassen sich mit zusätzlicher Sicherheitssoftware überwachen. Die S7-1200 dokumentiert äquivalent Änderungen des Betriebssystemzustandes (wie bereits für die Installationen erwähnt). Darüber hinaus können in neueren Versionen weitere Systeminformationen durch die System- und Taktmerker gewonnen werden. Desweiteren werden Sicherheitsereignisse wie etwa die Wiederherstellung einer CPU-Konfiguration, Änderungen an den Schutzeinstellungen der CPU und Änderungen an der Aktualisierungsdatei der Firmware gewonnen. Es ist nach Wissen des Verfassers nicht genau geklärt, ob dabei auch konkrete Änderungen der Konfiguration bzw. der Firmware dokumentiert werden können.

Als letzter Vergleichspunkt werden Fehlermeldungen des Systems betrachtet. Diese Unterkategorie überschneidet sich teilweise mit dem Punkt der Hardwarefehlermeldungen, jedoch werden zudem softwareseitige Fehler bei der Systemausführung mit einbezogen. Diese werden bei den Betriebssystemen in System- und Diagnoseprotokollen festgehalten. Die S7-1200 dokumentiert in diesem Zusammenhang funktionelle Fehler während des Betriebs wie etwa E/A-Fehler durch Unerreichbarkeit oder fehlerhaften Zugriff auf E/A Speicherplätze oder Datenbausteine. Desweiteren werden auch Überschreitungen der maximalen Zykluszeit protokolliert.

Kategorie: Datenspeicher

Änderungen an Dateien werden auf Betriebssystemen durch das verwendete Dateisystem durch Zeitstempel definiert. Neben dem Eigentümer/Ersteller der Datei werden in moderenen Dateisystemen der Zeitpunkt der Erstellung, der letzten Änderung und des letzten Zugriffes in Protokollen festgehalten sowie der Zugriff auf kritische und bestimmte benutzerdefinierte Verzeichnisse. Bei der S7-1200 ist diese Dokumentation deutlich stärker limitiert. So werden im Diagnosepuffer Fehlerereignisse von Zugriffsoperationen (etwa beim Lesen/Schreiben von Datensätzen) dokumentiert. Quellen, die die Protokollierung weiterer Ereignisse belegen, konnten

nicht gefunden werden. Das einzige Sicherheitsevent in dieser Kategorie protokolliert, dass eine Änderung von Daten auf einer Speicherkarte der S7-1200 stattgefunden hat.

Kategorie: Prozesse und Dienste

Windows und Linux dokumentieren den Lebenszyklus der durchgeführten Prozesse und (Hintergrund-)Dienste. Dabei können detaillierte Informationen wie etwa das Ausführungsverzeichnis, Laufzeit, Speicherverbrauch oder der Elternprozess ermittelt werden. Mithilfe von zusätzlicher Software, wie etwa einer Anti-Malware-Software, lassen sich darüber hinaus verdächtige Prozesse oder Prozessketten erkennen und Sicherheitsereignisse protokollieren. Zusätzlich werden Ereignisse bzgl. Ausführung und Fehler der Hintergrunddienste (Services / „cron jobs“) protokollieren. Bei der S7-1200 SPS werden für das Aufrufen und Ausführen des Programmes und Unterprogrammen verschieden Organisationsbausteine verwendet, die durch von der CPU erkannte Ereignisse angestoßen und teils protokolliert werden. Für jeden Organisationsbausteintyp wird eine limitierte Liste an Standard- und Fehlerereignissen definiert. Diese Form der Dokumentation schließt die Ausführung und Ausführungspausen von (Unter-)Programmen sowie die zeitgesteuerte Ausführung von Unterprogrammen ein.

Kategorie: Benutzerzugriff

Sowohl für Windows- als auch Linux-basierte Betriebssysteme werden Zugriffsversuche auf das System in verschiedenen Protokolldateien dokumentiert. Bei Windows werden u.a. auch Loginversuche von lokalen und domänen-abhängigen Logins unterschieden. Im Detail werden fehlgeschlagene als auch erfolgreiche Anmeldungen protokolliert. Desweiteren wird in einem gewissen Rahmen die Nutzung von privilegierten Benutzerrechten sowie Änderungen an den Sicherheitseinstellungen des Systems bzgl. Authentifizierungs- und Authorisierungsregeln festgehalten. Bzgl. der betrachteten SPS sind für den direkten Zugriff auf die CPU bzw. den Zugriff auf den integrierten Webserver verschiedene Sicherheitsereignisse definiert. Die Liste dieser Ereignisse beinhaltet zunächst Ereignisse bzgl. der erfolgreichen oder fehlgeschlagenen Anmeldung an dem Gerät (spezieller: der Online CPU) bzw. dem Webserver (etwa durch Eingabe eines Passwortes). Desweiteren werden weitere Auffälligkeiten dokumentiert wie etwa Zeitablauf einer Online-Verbindung durch Inaktivität, multiple gleichzeitige Zugriffe und Änderungen an den Schutzeinstellungen, speziell des CPU-Schutzes. Die Genauigkeit der Protokollierung dieser Ereignisse (speziell bzgl. der Häufigkeit) kann durch Einstellungen des Diagnosepuffers limitiert wer-

den. Zusätzlich zu dieser Liste könnten weitere Sicherheitsereignisse existieren, zu denen jedoch zum Zeitpunkt des Rechercheabschlusses keine Dokumentation gefunden werden konnte.

Kategorie: Anwendungen

An dieser Stelle soll noch ein kurzer Blick auf die verwendeten Anwendungen sowohl im Unternehmensnetzwerk als auch im Fertigungsnetzwerk geworfen werden. Alle drei Anwendungen (Apache Webserver, Microsoft SQL Server, Siemens WinCC) setzen auf Windows oder Linux auf. Der Apache Webserver und Microsoft SQL Server dokumentieren beide verschiedene Aspekte in verschiedenen Protokolldateien. Dabei werden Ereignisse teilweise in Systemprotokolle integriert. Zu den dokumentierten Bereichen zählen u.a. die Dokumentation von Benutzerzugriffen, Datenanforderungen über die jeweilige Schnittstelle, Senden von Befehlen und Anwendungsfehler. Damit werden verschiedene Bereiche überwacht und protokolliert, die genutzt werden können, um den Zugriff auf gespeicherte Daten in der jeweiligen Anwendung nachzuvollziehen. Für WinCC ergeben sich ähnliche Fähigkeiten. Der Begriff „Protokollierung“ wird in WinCC als Ausdruck für die Möglichkeit verwendet, verschiedene Datensätze auszudrucken. Neben der Dokumentation von ein- und ausgehenden Prozessdaten, Projektdaten und Änderungen an diesen Projektdaten kann mit Hilfe der WinCC/Audit Option eine Dokumentation der Benutzeraktionen erreicht werden. Dabei werden sämtliche Aktionen des Benutzers vom erfolgreichen/-fehlerhaften Login (inklusive Methode) über die Ausführung von Aktionen innerhalb des Projektes bis zur Logout-Methode nachvollzogen werden.

5.2.2 Kommunikationsprotokolle

Für den Vergleich der Kommunikationsprotokolle werden die Kategorien „Header“ und „Verhalten“ verwendet. Da jedoch die Protokolle unterschiedliche Funktionalitäten erfüllen, wird im Folgenden darüber hinaus zwischen Kommunikations- und Anwendungsprotokollen unterschieden. Die Kommunikationsprotokolle ermöglichen die Datenübertragung und die Anwendungsprotokolle setzen darauf auf.

Für die Unterteilung werden daher zunächst PROFIBUS und die Kombination von Ethernet, IP und TCP/UDP verglichen. Darauf folgend werden die Anwendungsprotokolle des Unternehmensnetzwerkes und die Gemeinsamkeiten mit Informationen über Anwendungen bzw. Diensten von PROFIBUS und PROFINET jeweils verglichen.

Vergleich von PROFIBUS und Ethernet-basierter Kommunikation

Header

Bei dem Vergleich von PROFIBUS und Ethernet-basierter Kommunikation sind zunächst die Gemeinsamkeiten zu nennen. Diese bestehen in einer Sender- und Empfängeradresse der Nachricht, einer Kennung der Nachricht für einen bestimmten Typ sowie einer Kennung für einen bestimmten Dienst bzw. Service-Typ.

Der fundamentale Unterschied ergibt sich aus der Menge der möglichen Kodierungen bzw. Detailtiefe der Informationen über die verschickte Nachricht selbst. So finden sich Daten, wie etwa die Gültigskeitsdauer einer Nachricht oder Informationen über die Länge, in jeder Ethernet-basierten Nachricht, während die Gültigkeitsdauer überhaupt nicht und die Länge der Nachricht nur für ein bestimmtes Nachrichtenformat in der PROFIBUS-Kommunikation versendet wird. Bei PROFIBUS werden teilweise Paritäts-Bits oder CRC-Daten innerhalb der Nutzdaten mitgesendet.

Eine weitere Gemeinsamkeit ergibt sich aus der Übermittlung von Port-Daten bzw. SAP (Service Access Point) Daten. Diese Informationen definieren äquivalent den Zugangspunkt einer Anwendung bzw. eines Dienstes am System.

Informationen durch Verhaltensanalyse

Bei der Betrachtung der Verhaltens der Kommunikationssysteme im laufenden Betrieb lassen sich Gemeinsamkeiten finden in Form der Nachvollziehbarkeit bei der Eingliederung von neuen Teilnehmern. In der Ethernet-basierten Kommunikation wird von verschiedenen Protokollen wie etwa dem Adress Resolution Protocol (ARP) eine Zuweisungsliste am Switch bzw. Router des jeweiligen Netzwerkes erzeugt. Diese Aktivität kann dokumentiert werden. Bei der PROFIBUS Kommunikation kann durch das FDL das Hinzufügen bzw. Entfernen von Kommunikationsteilnehmern durch Mitschneiden der Kommunikation ermittelt werden und durch eine Option eines PROFIBUS-Master-Gerätes lassen sich die Kommunikationsteilnehmer im Feldbus durch Abfrage ermitteln.

Bei der Kommunikation zwischen Geräten lassen sich zusätzliche Daten über den Kommunikationsaufbau bei der Verwendung von TCP gewinnen. So wird bei der Kommunikation zwischen Sender und Empfänger eine Sequenz- bzw. Bestätigungsnummer verschickt. Dieser Punkt ist erwähnenswert, da bei PROFIBUS ebenfalls Bestätigungen für den Empfang von Nachrichten durch die PROFIBUS-Slaves gesendet werden, jedoch die Bestätigung allgemein gehalten wird und keine Daten über eine bestimmte Nachricht in der Bestätigung vorhanden ist. Bei der Verwendung von UDP werden nach der Protokollspezifikation keine Bestätigungen versendet und keine Sequenznummern übertragen.

Bei der Verwendung von TCP durch den TCP-Handshake und das Austauschen von Informationen für die Herstellung einer Verbindung. Die Aufzeichnung dieses Verhaltens ergibt Informationen über den konkreten Verbindungsaufbau zwischen zwei Kommunikationsteilnehmern. Bei PROFIBUS lassen sich diese Informationen aus der Analyse des Nachrichtenverlaufs und die Ermittlung der jeweiligen Empfängeradresse ermitteln. Da genau ein Kommunikationsteilnehmer zur Zeit auf dem Kommunikationsbus senden darf und ein PROFIBUS-Slave-Gerät eine temporäre Sendeerlaubnis vom PROFIBUS-Master für das Senden von Prozessdaten bzw. Bestätigungen erhält, lässt sich auf diese Art rudimentär ein ähnliches Verhalten nachvollziehen. Abhängig von dem verwendeten Übertragungsdienst kann jedoch nicht ermittelt werden, ob die Nachricht erhalten wurde, wenn eine Quittierung nicht verlangt wird. Hier lässt sich eine Parallele zu UDP ziehen.

Fehler innerhalb der Kommunikation lassen sich durch Fehlermeldungen der Kommunikationsteilnehmer in Ethernet-basierten Netzwerken nachvollziehen (etwa in Form von ICMP-Nachrichten). Bei PROFIBUS-Systemen können durch Dienste des FDMA-Protokolls Funktionsfehlernachrichten verschickt werden.

PROFIBUS-Dienste und Anwendungsprotokolle des TCP/IP Stacks

Gemeinsamkeiten der Kommunikationsprotokolle

Bei erster Betrachtung verbindet die ausgesuchten Protokolle ihre grundlegende Eigenschaft der Client-Server-Anwendung, d.h. jedes Protokoll dient dazu, den Lebenszyklus der Verbindung von einem Client-Gerät zu einem Server-Gerät zu verwalten und den Austausch zu kontrollieren.

An dieser Stelle hebt sich HTTP in der Form von den restlichen Netzwerkprotokollen derart ab, dass es für die Verbindungsverwaltung und -abwicklung nicht mehrere Subprotokolle verwendet. SSH, RDP und MSSQL benutzen Protokollsuiten mit dedizierten Protokollen für verschiedene Schritte. So werden Protokolle verwendet, um die Verbindung zu initialisieren oder die Verbindungskonfiguration zu regeln. Ein weiteres Subprotokoll dient dann der Ausführung der eigentlichen Anwendung durch den Nutzer. Auffallend sind auch die vorhanden Authentisierungsmechanismen, die durch ein eigenes Subprotokoll durchgeführt werden.

Desweiteren lassen sich die Protokolle zunächst in zwei Arten unterteilen. Die Protokolle der MSSQL Suite und das HTTP dienen der Anfrage von Ressourcen und Diensten an den Server und den Erhalt einer Bestätigung und/oder Ressource. SSH und RDP hingegen dienen dem Zugriff auf das Serversystem.

Bei der Betrachtung der Header der versendeten Nachrichten kann von jedem

Protokoll Name und -version sowie Nachrichtentyp ermittelt werden. Darüber hinaus werden weitere Informationen über die gewählte Verbindung mitgesendet, wie etwa die verwendeten Verschlüsselungsmethoden (für HTTP über TLS (Schicht 5 im OSI Modell)) und Informationen über die Nachricht in Form von Längendaten und weiteren Daten über den versendeten Payload. Alle Protokolle haben die Möglichkeit, verschiedene Verschlüsselungs- und Authentifizierungsmechanismen einzusetzen.

Der Aufbau der Verbindung für SSH, RDP und MSSQL erfolgt in festgelegten Sequenzen. So wird eine initiale Verbindung erzeugt, die Authentizität des Servers geprüft, Konfigurationsdaten für die Verbindung ausgetauscht und eine Authentizitätsprüfung des Benutzers durchgeführt. Für HTTP gilt dies nicht, da die Kommunikation zustandslos verläuft, d.h. das Anfragen und Antworten ausgetauscht werden ohne Kenntnis des vorhergegangenen Austausches.

Vergleich mit PROFIBUS-Diensten

Für den Vergleich der Anwendungsprotokolle werden im Folgenden PROFIBUS DP-V0 und DP-V1 betrachtet. Diese setzen auf der Kommunikationsfunktionalität auf, die durch das FDL-Protokoll bereitgestellt wird. Dies bildet einen Äquivalenzpunkt zu der Art und Weise, wie die Anwendungsprotokolle auf der Ethernet-basierten Kommunikation aufsetzen.

Als ersten Vergleichspunkt wird der Verbindungsaufbau betrachtet. Ähnlich wie bei einem Client-Server Modell werden über das Protokoll Anfragen geschickt und und Antworten erhalten. Für die Initialisierung eines PROFIBUS-DP-Slave-Gerätes wird, wie bei SSH, RDP und MSSQL, eine Verbindung hergestellt und Konfigurationsdaten gesendet. Im Unterschied ist der Austausch der Daten allerdings einseitig, da die Informationen nur von dem Master-Gerät an das Slave-Gerät gesendet und der Empfang durch den Slave quittiert wird. Desweiteren findet keine Prüfung der Authentizität des Masters statt. Zwar wird eine Identifikationsnummer gesendet, anhand derer das Slave-Gerät prüft, ob die über den gemeinsamen Datenbus gesendete Konfiguration auf dem Gerät angewendet werden soll. Darüber hinaus wird allerdings keine weitere Prüfung durchgeführt. Es werden zudem keine Verschlüsselungsmethoden ausgetauscht. Die Definition der Parameter wird über die Daten der GSD-Datei definiert, die dem Master-Gerät zur Verfügung stehen. In Folge dessen wird die Konfiguration zwar innerhalb der Sequenz geprüft, es findet jedoch im Vergleich zu SSH, RDP und MSSQL keine Aushandlung der gemeinsamen Konfiguration für die Verbindung statt.

Eine Gemeinsamkeit der Headerdaten zwischen PROFIBUS DP-V0/V1 und den Anwendungsprotokollen besteht in der Information über die Länge der Anfrage/Antwort-

Nachricht sowie des Nachrichtentyps. Desweiteren werden bei der DP-V1 Kommunikation äquivalent zu HTTP Informationen über die angefragte Ressource übertragen. Bei Verwendung der synchronen Kommunikation mit DP-V0 werden bei der Initialisierung im Zuge der Parametrisierung Daten über die verwendbaren Alarmtypen, die Gruppenzugehörigkeit für die Anwendung der Broadcast-Funktionalität durch einen Master der Klasse 1 sowie weitere Benutzerparameter übertragen. Im Vergleich zu den SSH, RDP und MSSQL werden diese Konfigurationsdaten unverschlüsselt übertragen.

Eine weitere Gemeinsamkeit besteht in der Existenz eines spezifizierten Fehler-Nachrichtentyps, der einen Fehler der Kommunikation beschreibt. Im Unterschied zu den Anwendungsprotokollen des Unternehmensnetzwerk variiert jedoch die Anzahl der Typen abhängig von dem Kommunikationspartner. Während dies eingeschränkt auch für unterschiedliche Servertypen zutrifft, wird jedoch eine Grundmenge über das Protokoll definiert. Darin liegt der entscheidende Unterschied.

Vergleich PROFINET und PROFIBUS

Als Grundlage basiert PROFINET zunächst auf dem Ethernet-Standard. Abhängig von dem gewählten Kommunikationskanal werden jedoch UDP/IP bzw. TCP/IP verwendet oder nur auf Basis von Ethernet. In diesem Fall besteht eine Nachricht aus einem Rahmen, der eine PROFIBUS-Nachricht zwischen den Ethernet-Headern und einem CRC-Feld einkapselt. Auf der Anwendungsebene betrachtet man die PROFINET Dienste und Protokolle [29]. Die Dienste beziehen sich prinzipiell auf die bereits verglichenen Read/Write-Dienste von PROFIBUS. Zusätzlich wird durch die verwendete Kommunikationbeziehung (CM, IO data oder Alarm) eine zusätzliche Information über den allgemeinen Zweck der Nachricht gewonnen. Diese Beziehung und die zusätzlichen Protokolle DCP und LLDP (vgl. Industrieanalyse/PROFINET) erlauben eine genauere Eingrenzung des Nachrichtenzweckes, äquivalent zu den Subprotokollen von SSH, RDP und MSSQL. Die verwendeten Protokolle ermöglichen es im Vergleich zu einer PROFIBUS-Kommunikation, zusätzliche Informationen über die Identität eines Kommunikationsteilnehmers zu erhalten sowie Prozesse der Adressvergabe nachzuvollziehen. Dies ermöglicht die genauere Segmentierung des Nachrichtenverkehrs und in der Folge zusätzliche Informationen über den Nachrichtenaustausch im Netzwerk.

5.3 Ergebnis

Als Ergebnis des Vergleiches ist festzustellen, dass grundlegende Ähnlichkeiten der Informationsmengen zu erkennen sind. Dies ist ein natürlicher Schluss aus dem Zweck der Anwendung. Es ergeben sich jedoch auch Unterschiede.

Zunächst kann sowohl für die verglichenen Systeme als auch Kommunikationsprotokolle festgehalten werden, dass die Elemente des industriellen Netzwerkes nach dem vorliegenden Vergleich eine eingeschränktere Informationstiefe vorweisen. Im Speziellen ergeben sich bei der S7-1200 Limitierungen in der Dokumentation von Ereignissen im Diagnosepuffer, die hauptsächlich funktionelle Fehlermeldungen und Systemereignisse zeigen. Dies ist besonders präsent in der Dokumentation von Parameteränderungen. Die potentielle Einschränkung der Protokollierung von Benutzerzugriffen (im Sinne der Häufigkeit und Vielfalt der Ereignisse), basierend auf der Auslastung des Diagnosepuffers, schränkt die Möglichkeit der Dokumentation von Änderungen ebenfalls ein.

Bei dem Vergleich der Kommunikationsprotokolle ergaben sich die folgenden Unterschiede: Zum einen konnte keine Unterstützung von Integritätsprüfung der Kommunikationspartner bei der Verwendung des PROFIBUS-Protokoll festgestellt werden. Darüber hinaus ergeben sich Unterschiede in der Art der Festlegung verfügbarer Nachrichtentypen für die Kommunikation zwischen zwei Geräten auf der Anwendungsebene.

Kapitel 6

Bewertung der Informationslücken

In diesem Kapitel erfolgt die Bewertung der Unterschiede, die aus dem Vergleich ermittelt wurden. Zunächst wird die Methode definiert. Diese beschreibt das Ziel der Bewertung, die Herleitung der angewendeten Methode und die verwendeten Kriterien. Darauf folgt die Diskussion der Unterschiede.

6.1 Bewertungsmethode

Diese Bewertung soll veranschaulichen, ob es sich bei den gefundenen Unterschieden um einen oder mehrere Bereiche handelt, in denen Informationen nicht verfügbar sind („Informationslücke“), die für die proaktive Erkennung eines laufenden Angriffes und/oder reaktive, forensische Untersuchungen hilfreich sein könnten. Diese Aktivitäten stehen in direktem Zusammenhang mit der Funktion, die durch ein SIEM-System ausgeführt wird.

6.1.1 Entwicklung der Methode

Die Methodik basiert auf dem Vorgehen der forensischen Untersuchung eins Vorfalles durch ein Sicherheitsteam in einem Unternehmensnetzwerk.

Bei der Erkennung von Angriffsszenarien spielen erste Anzeichen, die potentiell auf einen Angriff hindeuten, eine tragende Rolle. Informationen, die zu solchen Anzeichen ausgewertet werden, werden als „Indicator of Compromise“ (IoC) bezeichnet. Ein IoC ist eine Information, die auf ein anomales Verhalten oder einen anomalen Zustand eines Netzwerkelementes schließen lässt. Um dieses Verhalten zu erkennen ergeben sich prinzipiell zwei Bereiche der Analyse: 1) Das Verhalten des Netzwerkelementes in sich, also der Zustand des Elements und Änderungen dieses Zustandes sowie 2) die Interaktion mit anderen Netzwerkelementen in Form des Austauschens

von Nachrichten über die Netzwerkverbindung (vgl. Kaspersky Paper).
Bewertung: Quelle Kaspersky Paper einfügen!

Diese Beobachtung wird durch das Vorgehen in Unternehmen durch den „Incident Response Process“gestützt. Dieser Prozess dient als Durchführungsplan eines Teams, das für die Aufklärung und Behebung von Indikatoren und Brüchen des Sicherheitszustandes eines Netzwerkes zuständig ist.

Der initiale Zustand dieses Plans ist die Überwachung der Netzwerkelemente und ihrer Kommunikation mit Hilfe von Detektions- und Analysesoftware (z.B. SIEM Systeme) und -hardware (z.B. „Intrusion Detection Systems“). Durch die Überwachung werden eine Vielzahl an Alarmen oder Sicherheitsereignissen produziert, die auf Fehler oder Anomalien im Netzwerk hinweisen. Werden diese Informationen aus diesen Ereignissen als potentiell kritisch eingestuft, wird ein IoC an das Team ausgegeben, das basierend auf Risiko und potentieller Schadensgröße weitere Untersuchungen einleitet. Bei diesen Untersuchungen wird der oder die IoC(s) als Ansatz genutzt. Ein IoC kann dabei auf Informationen im Payload oder Headern von Netzwerkpaketen als auch aus Logging Informationen eines Betriebssystems oder einer Überwachungssoftware auf einem Netzwerkelement beruhen. Basierend auf dem Ansatzpunkt können Informationen sowohl aus der Kommunikation als auch der Netzwerkelemente weitere Hinweise liefern, um den potentiellen Sicherheitseinbruch einzugrenzen und schließlich zu identifizieren.

Wie bereits in den Analysen beschränkt sich diese Arbeit konkret auf Angriffe mit Hilfe von Schadprogrammen (Malware). Ziel einer forensischen Untersuchung bzgl. einer *Malware* ist es, Mechanismen für Infektion und Verbreitung sowie Interaktionen mit System, Netzwerk und Angreifer zu identifzieren. In Bezug auf die betrachteten Elemente des Industrienetzwerkmodells und der verfügbaren Informationen ist interessant, welche Informationen als Indikatoren verwendet werden können, um auf die Existenz von Schadcodes zu schließen.

Dieses Vorgehen zeigt, wie Malware-Angriffe identifiziert und untersucht werden. Um also die Notwendigkeit bestimmter Informationen zu beurteilen, muss an konkreten Beispielen geprüft werden, ob und inwiefern die Dokumentation dieser Informationen vorhanden gewesen ist und welchen Beitrag die Informationen konkret für das Erreichen der forensischen Ziele leisten können. Eines der bekanntesten Beispiele für einen Angriff auf ein Fertigungssystem ist der Fall *Stuxnet*, welcher als Basis für die Bewertung verwendet wird. Das Beispiel von Stuxnet zeigt einen durchgängigen Infektionsweg von der Ausnutzung einer Schwachstelle, der Ausbrei-

tung über das Netzwerk und schließlich der Manipulation einer SPS. Aus diesem Grund werden anhand des Beispiels von Stuxnet die Unterschiede untersucht.

6.1.2 Bewertungkriterien

Basierend auf der Vorgehensweise der forensischen Untersuchung werden die folgenden Bewertungskriterien angewandt:

- Informationen über Ziele und Aktionen der Malware
- Bedeutung als Indikator

Bewertung: Bewertungskriterien kurz erläutern

6.2 Bewertung

Für die Bewertung wird zunächst kurz die Manipulation einer SPS durch Stuxnet geschildert.

6.2.1 Stuxnet

Bei dem Stuxnet-Angriff handelte es sich um einen vielschichtigen Angriff mit dem Ziel, das Verhalten von ICSs (Industrial Control Systems) zu manipulieren. Zu diesem Zweck wird in mehreren Schritten das betreffende Netzwerk infiziert und die Malware verbreitet. Bei der Verbreitung wird nach einem Computer gesucht, der als Programmiergerät für eine SPS verwendet wird. Es wird u.a. gezielt nach einer installierten WinCC-Software gesucht. Wenn ein solchen Systems gefunden wird, kann über Kommandos an den Datenbankserver die Schadsoftware geladen werden. Diese infiziert schrittweise Dateien von bestimmten Step7-Projektdateien über Schnittstellen spezieller Einschubmethoden (Hook API). Die Step7-Projekte werden über eine integrierte Filter-Logik ausgewählt. Für die Beeinflussung der SPS wird die Datei so manipuliert, dass Stuxnet über die WinCC-Ladeschnittstelle für Projekte auf die SPS übertragen wird.

DIe WinCC/Step7-Software benutzt eine spezifische .dll-Datei, um die Kommunikation zwischen WinCC und der SPS zu ermöglichen. Für die Kommunikation ruft die Anwendung die Bibliothek „s7otbxdx.dll“ auf. Diese enthält bis zu 106 (Maximum) Funktionen für die Verwaltung der Kommunikation.

So wurde etwa die Funktion *s7blk_ read* verwendet, um eine Anfrage von WinCC an eine SPS weiterzuleiten. Stuxnet benennt diese Datei um und erstellt die eigene .dll-Datei mit gleichem Namen. Diese leitet die meisten Funktionsaufrufe an die Original-Datei weiter, fängt jedoch bestimmte Aufrufe ab. Zu diesen zählen u.a. die Funktionen für das Schreiben, das Lesen und die Aufzählung von Funktions-, (System-)Daten- und Organisationsbausteinen. Erwähnenswert ist an dieser Stelle, dass Stuxnet die Systemdatenbausteine prüft und unter bestimmten Bedingungen die Hersteller-ID (vergeben durch den PI-Dachverband (Profibus & Profinet International)) verändert wird. Das Abfangen dieser Funktionen erlaubt die Manipulation der Bausteindaten, die zu der SPS gesendet und von der SPS empfangen werden.

Im Folgenden manipuliert Stuxnet spezielle Organisationsbausteine nach bestimmten Mustern. Die Manipulation eines speziellen Datenbausteins erlaubt es dabei, Daten auf der SPS zu überwachen und das Verhalten der Malware (Muster) anzupassen. Diese Funktion wird benutzt, um das Kommunikationsverhalten der angeschlossenen Systeme nach Aktionen, die durch Stuxnet ausgelöst wurden, zu überwachen sowie die Manipulation der Pakete mit Hilfe von aufgezeichneten Daten zu verfeinern. Die eigentliche Manipulation im Falle von Stuxnet wird durch die Anpassung von Frequenzwerten durchgeführt, die an Frequenz-Konvertierungstreiber geschickt werden. Diese Treiber erlauben die Kommunikation mit Motoren, die in einer von der Frequenz-abhängigen Geschwindigkeit arbeiten.

Um diese Funktionalität zu erhalten, wird über die Manipulationsfähigkeit der Kommunikation und Bausteine der SPS zudem eine Art Rootkit durch die Vorverarbeitung von Antworten an das WinCC-Systems umgesetzt. Die Folge daraus ist, dass die Antworten, die an das WinCC weitergeleitet werden, keine korrekten Daten enthalten und damit die Veränderungen nicht unmittelbar erkannt werden können.

6.2.2 Betrachtung der Unterschiede

Die im Vergleich ermittelten Unterschiede sind im Kern die folgenden:

- S7-1200 SPS
  - Keine Dokumentation von Parameter- und Bausteinänderungen aus authentifizierter Quelle
  - Eingeschränkte Dokumentation der Benutzerzugriffe, u.a. bei bestimmten Einstellungen des Diagnosepuffers
- PROFIBUS-Kommunikation
  - Keine Informationen über die Integrität der Sender-/Empfänger-ID

– Eingeschränkte Nachrichteninformationen, u.a. basierend auf der GSD-Konfiguration

Fehlende Dokumentation von Parameter- und Bausteinänderungen aus authentifzierter Quelle

Der Fall Stuxnet zeigt, dass Parameter in Ein- und Ausgängen sowie Bausteine manipuliert werden können, um das Verhalten einer SPS zu verändern und Störungen am Fertigunssystem zu verursachen. Durch eine Dokumentation von Änderungen bzw. der Änderung eines Bausteines im Sinne von Änderungen bei Kommunikation mit einem Anwendersystem könnten Hinweise auf ungewollte bzw. nichtangeordnete Änderungen schneller und gezielter erschlossen werden. Desweiteren ermöglicht eine solche Dokumentation Rückschlüsse auf die Existenz und die Funktion der Malware. So ergibt eine Änderung an OB1 etwa, dass die Ausführung bestimmter Funktionsbausteine durchgeführt werden soll. Die Änderungen am Datenbaustein geben einen Hinweis auf die protokollierten Daten, woraus sich weitere Schlüsse auf Ziel und Möglichkeiten der Malware ergeben können. Die Dokumentation der Veränderung von bestimmten Speicherbereichen wie etwa der Prozessabbilder können Hinweise auf das Ziel der Manipulation des Fertigungsprozesses und die konkrete Methode geben. Zusätzlich gibt die bloße Dokumentation der Änderung bestimmter Bausteine einen Hinweis darauf, welche Bausteine anderer SPSen im Netzwerk betroffen sein könnten.

Insgesamt könnte durch eine regelmäßige Abfrage der protokollierten Änderung und dem Abgleich mit Aktionprotokollen etwa einer WinCC-Applikation und/oder durch Einsatz zusätzlicher Programmiergeräte die Integrität der Daten bestätigt werden. Jegliche Änderungen / Abweichungen vom Soll-Zustand ergeben einen Indikator für potentielle Manipulationen.

Geringe Dokumentation des Benutzerzugriffes

Das Stuxnet-Beispiel zeigt auch, wie wichtig der Schutz von Informationen auf den SPSen ist. Essentieller Bestandteil der Stuxnet Sequenzen ist das Verändern von Bausteinen, um gewisse Funktionen auszuführen oder Daten abzufragen. Die S7-1200 dokumentiert Zugriffsversuche auf die CPU. Nach erfolgter Recherche ist nicht bekannt, ob Ereignisse bei (in-)korrektem Zugriff auf geschützte Datenbausteine im Diagnosepuffer eingetragen werden. Basierend auf der abgedeckten Funktionalität des Systems wird jedoch die Hypothese aufgestellt, dass zumindest Fehlerereignisse dokumentiert werden. Alle darauf folgenden Benutzeraktionen bzgl. des Bausteins werden jedoch nicht dokumentiert, sofern keine funktionellen Fehlerereignisse ausgelöst werden. Die Dokumentation weiterer Informationen über die Interaktion des

Benutzers mit dem Gerät, liefert Hinweise auf den Zweck der Interaktion und damit ggf. Aktionen der Malware. Diese kann genutzt werden, um in Verbindung mit Informationen über Zugriffsverbindungen zu der SPS Hinweise auf Anomalien zu generien. Ein Beispiel für einen Hinweis auf eine Anomalie könnte etwa in Form von Anmeldungen außerhalb bestimmter Zeitintervalle oder undokumentierter Änderungsanweisungen bestehen. Greift etwa die Malware mit dem vorher beschaften Passwort oder anderweitig erfolgreich auf einen geschützten Baustein zu, kann über die Dokumentation der Aktionen ein Hinweis auf den Zweck und/oder die Funktion der Malware erhalten werden. Bzgl. der Durchführung von Stuxnet ist nach Wissen des Verfassers unklar, ob der Know-How-Schutz der S7-1200 die Manipulationen durch Stuxnet hätte einschränken oder verhindern können.

PROFIBUS Sender-/Empfänger-Verifikation

Eine bekannte, nicht verfügbare Information des PROFIBUS-Systems ist das Fehlen von Daten zur Verifikation einer Sender- bzw. Empfänger-ID. Im konkreten Fall von Stuxnet hätten diese Informationen keinen Einfluss gehabt, da der Datenaustausch zwischen WinCC und SPS unverschlüsselt stattfand und die Manipulation auf dem WinCC-System stattfand. Die Information, dass der Sender- bzw. der Empfänger-ID vertraut werden kann (etwa durch ein (Hardware-)Zertifikat), ermöglicht eine sichere Zuordnung dokumentierter Telegramme, sodass z.B. die Verschleierung der Identität durch Angriffstypen, die Spoofing-Mechanismen, etwa bei Man-In-The-Middle-Angriffen, ausgeschlossen bzw. als unwahrscheinlich eingestuft werden kann.

6.2.3 Zusammenfassung

Zusammenfassend ist festzuhalten, dass die gefundenen Unterschiede in der Informationverfügbarkeit bzw. die fehlenden Informationen bei einer Attacke wie im Beispiel von Stuxnet wichtige Informationen auf die Existenz und Aktionen der Malware bereitstellen könnten.

Hinzuzufügen ist, dass durch die vertikale (und teils horizontale) Integration der Systeme Komplexitäten bei der Entwicklung von Stuxnet bzgl. der Autonomie der Aktivitäten beseitigt werden könnten. Durch einen Insider oder durch Erlangen der Kontrolle über Systeme im Unternehmensnetzwerk könnte ein Angreifer weit genug vordringen, um diese Aktivitäten durch spätere Einschleusung von zusätzlichem Schadcode, etwa über einen C&C-Server, „nachzureichen“. Diese Hypothese soll darauf hinweisen, dass die Komplexität und der Erstellungsaufwand von Stuxnet-Funktionalität in zukünftigen Manipulationsversuchen leichter und damit

für eine größere Menge an Angreifern zugänglich sein könnte. Die konkrete Untersuchung dieses Sachverhaltes kann in folgenden Forschungsarbeiten durchgeführt werden.

Kapitel 7

Lösungsansatz zur Schließung der fokussierten Informationslücke

In diesem Kapitel wird die Informationlücke der Protokollierung von Änderungen an Datenbausteinen und Parametern untersucht und ein Ansatz für die Schließung dieser Informationslücke beschrieben.

7.1 Ansatzentwicklung und Szenarien

Das Ziel des Ansatzes ist es, eine Möglichkeit zu finden, Änderungen an E/A-Parametern sowie Bausteinen zu dokumentieren.

7.1.1 Annahmen und Szenarien

Zu diesem Zweck werden zunächst die folgenden Annahmen definiert:

- Die Protokollierung der Änderungen an Parametern und Bausteinen ist aktuell nicht möglich
- Für eine skalierbare Lösung ist eine dauerhafte Überwachung nach aktuellem Stand nicht möglich
- Für die Komprimittierung einer SPS ist die Infizierung eines Servers mit Kommunikationsschnittstelle zu der Malware notwendig

Diese Annahmen werden im Folgenden weiter ausgeführt und begründet.

Keine Änderungsprotokollierung auf der SPS

Für diese Hypothese sprechen mehrere Faktoren. Zum einen sind die Werte von Prozessdaten und der zugehörigen Speicherplätze sehr volatil. Es wird angenommen, dass sich aus der Zykluszeit der Ausführung eines Rezeptes, der zugehörigen Änderung der Sensorwerte und der daraus resultierenden Anpassung der Ausgangswerte der verschiedenen angeschlossenen Feldgeräte eine hohe Wertefluktuation ergibt. Abhängig von der Größe des Ladespeichers und der Verfügbarkeit von zusätzlichem Speicherplatz durch eine Speicherkarte ist zudem die Kapazität des Ladespeichers und damit die Anzahl der möglichen Einträge begrenzt. Ein weiteres Argument ist, dass die Protokollierung von Prozessdaten auf einer SPS durch den Programmierer des Programmes eingepflegt werden müssen. Abhängig von der Zykluszeit und der notwendigen Operationen kann die Anzahl an möglichen Schreiboperationen stark begrenzt sein. Laut dem Systemhandbuch der im Modell verwendeten SPS S7-1200 kann bspw. nur die Erstellung einer Datenprotokolldatei einige Zeit in Anspruch nehmen. Desweiteren ist die Limitierung einer Datenprotokolldatei auf 255 Einträge eine weitere Begrenzung.

Keine dauerhafte Überwachung bei skalierbarer Lösung

Diese Hypothese wird aufgestellt basierend auf der Anzahl der zu überwachenden Geräte an einem Feldbus, z.B. PROFIBUS. Durch den gemeinsamen Datenbus steht für ein potentielles Gerät, welches Datenanfragen an SPS-Teilnehmer schickt, nur ein begrenzter Zeitrahmen zur Verfügung. Die Erhöhung des Zeitrahmens könnte die Prozessdurchführung einschränken bzw. nicht die dafür notwendigen Reaktionszeiten erreichen, die für die Prozessdurchführung notwendig sind.

Infizierung eines Servers für SPS-Manipulation notwendig

Diese Hypothese leitet sich aus der Vorgehensweise des Stuxnet-Angriffes ab. Die Manipulation der Kommunikationsschnittstelle kann bspw. durch reinen Schadcode auf der SPS nur schwierig verhindert oder verborgen werden. Zu diesem Zweck müssten Anwenderprogramm und Firmware derart manipuliert werden, dass die Korrektur der eigentlichen Werte, wie sie bei Stuxnet durch die Rootkit-Funktionalität durchgeführt werden, direkt auf der SPS mit der zur Verfügung stehenden, unter Umständen sehr geringen, Rechenleistung durchgeführt werden. Die Infizierung der SPS ist zudem ohne eine Infizierung eines Servers mit einer Verbindung zu der betroffenen SPS nur durch einen Insider möglich, dessen Aktivitäten ohne spezielle Kenntnisse, etwa durch das WinCC-SCADA-System, protokolliert werden.

Szenarien

Aus diesen Hypothesen ergeben sich die folgenden potentielle Angriffsszenarien für die Manipulation des SPS-Verhaltens:

- Manuelle Manipulation von Parametern durch einen Insider
  - über ein Hardware-Programmiergerät (erfordert physikalischen Zugriff)
  - über einen angeschlossenen Server mit installierter Programmiersoftware (etwa WinCC)
- Manipulation per Malware
  - Autonome Malware wie am Beispiel Stuxnet
  - Manipulierbare Malware, die etwa eine eigene Kommunikationsverbindung mit Hilfe der Serverschnittstellen aufbauen kann (erfordert Zugriff des Angreifers entweder direkt oder indirekt über eine Hintertür)

Auf der Grundlage der beschriebenen Hypothesen und Szenarien lassen sich zudem weitere Schlussfolgerungen für Anforderungen an einen möglichen Lösungsansatz generieren.

Basierend auf der weiteren Hypothese, dass eine dauerhafte Überwachung nicht stattfinden kann, lässt sich die Schlussfolgerung ziehen, dass die Erhebung und Prüfung der SPS-Daten zu bestimmten Zeitpunkten, etwa durch ein Ereignis, ausgelöst werden muss. Dieses Ereignis kann unterschiedlichen Szenarien entstammen. In Bezug auf das Stuxnet-Beispiel könnten sich z.B. Hinweise bei einem Abgleich der Prozessdaten der gesendeten/empfangenen Nachrichten ergeben. In diesem Falle wird ein Unterschied bei gesendeten Frequenzdaten festgestellt, die beim Aufruf der Write-Methode durch Stuxnet manipuliert werden. Ein weiterer Hinweis kann aus einem Abgleich der Anzahl der durch einen Benutzer angeordneten und der an die SPS gesendeten Anfragen resultieren. Hier wird im Zuge des Abgleiches der Anzahl der Nachrichten eine Differenz entstehen durch die regelmäßigen Anfragen der Stuxnet-Malware bzgl. des manipulierten Datenbausteines.

Eine weitere Schlussfolgerung ergibt sich aus der limitierten Protokollierungskapazität. Durch diese Limitierungen ist es notwendig, die Protokollierung der Nachrichten etwa durch ein dediziertes Element wie einen weiteren Server als DP-V1-Master der Klasse 2 durchführen zu lassen.

7.2 Beschreibung des Lösungsansatzes

Der folgende Lösungsansatz wird basierend auf den vorher geschilderten Annahmen und Schlussfolgerungen erstellt. Eine Übersicht über Datenfluss und Elemente des Ansatzes sind in der folgenden Abbildung zu sehen.

Lösungsansatz: Bild für Ansatzkonzept einfügen

Für den Lösungsansatz wird angenommen, dass eine Möglichkeit besteht, die Kommunikation zwischen SCADA-Server und SPS über ein Switch zu leiten oder über eine andere Möglichkeit den Nachrichtenverkehr durch ein externes Gerät zu überwachen. Aus den beschriebenen Hypothesen und Schlussfolgerungen ergibt sich die Notwendigkeit, dass eine unabhängige Datenquelle vorhanden sein muss, auf deren Basis unabhängig Informationen über den Datenverkehr erheben lassen. Diese Notwendigkeit besteht, da die Integrität der Kommunikationsdaten auf dem SCADA-Server bei einer Infizierung nicht gewährleistet werden kann.

Mit Hilfe dieses „Netzwerkmonitors“ kann der Datenaustausch protokolliert werden und an einen weiteren Server weitergegeben werden, der etwa als Zwischenstation für ein SIEM-System oder eine andere Sicherheitsanwendung verwendet werden kann. Dieser Server wird im Folgenden als „Agenten-Server“ bezeichnet. Die Aufgabe dieses Servers ist es, Datenprotokolle der SCADA-Anwendung und Datenprotokolle des Netzwerkmonitors zu vergleichen und Unterschiede in den Datensätzen festzustellen. Ist eine solche Differenz festgestellt, wird basierend auf den Details des Unterschiedes (etwa Unterschied zwischen den Prozessdaten oder Unterschiede in der Nachrichtenanzahl) die Aufgabe der Prüfung der entsprechenden SPS an ein unabhängiges Element weitergeleitet. Dies wird durch einen weiteren Server oder Industrie-PC mit einer Kommunikationsschnittstelle zu der entsprechenden SPS umgesetzt. Dieses Element wird im folgenden als „Prüfer“ bezeichnet. Die Aufgabe dieses Elementes ist es, einen bestimmten Datensatz aus der SPS auszulesen und diesen an den Agenten-Server zu senden. Der Agenten-Server vergleicht den Datensatz mit den zugehörigen Werten des SCADA-Systems. Diese zusätzliche Prüfung ist notwendig, um sicherzugehen, dass der übertragene Wert des Netzwerkmonitors dem auf der SPS gespeicherten Wert entspricht und ein Unterschied tatsächlich vorliegt. Wird ein Unterschied festgestellt, ergibt sich daraus die Information, dass eine Änderung der SPS-Daten ohne Dokumentation im SCADA-System vorgenommen wurde.

7.2.1 Limitierungen des Ansatzes

Der beschriebene Ansatz ermöglicht eine grundlegende Prüfung der SPS-Daten auf Änderungen, jedoch ist die Problematik der Datenmanipulation auf dem System vor der Übertragung der Daten an das Prüfer-System nicht auszuschließen. Eine weitere Problematik könnte im Bezug auf die Funktionsweise der SPS und die Dauer der Übertragung auftreten. So kann etwa bei dem Vergleich der Datensätze basierend auf dem Zeitstempel ein Unterschied der Daten bestehen. Aus diesem Grund funktioniert der beschriebene Ansatz nur annäherungsweise.

7.3 Demonstration des Lösungsansatzes

Für die Demonstration des beschriebenen Ansatzes wurde eine abstrakte Simulation erstellt. Da der Erwerb der notwendigen Software-Lizenzen außerhalb des finanziell sinnvollen Rahmens liegt, wurden für die Demonstration notwendige Charakteristiken der Elemente in Form von Java-basierten Klassen umgesetzt.

Dafür wurden die folgenden Software-Pakete erstellt:

- Kommunikationspakete: Dieses Paket definiert notwendige Kommunikationsroutinen und Nachrichtenformate (Orientierung an PROFINET-Formaten).
- SPS: Dieses Pakete enthält repräsentativ anfragbare Klassen (Bausteine und E/A-Speicher), eine „Firmware“ für die Ausführung von der Bausteinklassen sowie eine Verwaltung zu Kommunikationszwecken.
- SCADA_Server: Dieses Paket nutzt die definierten Kommunikationsroutinen für die Kommunikation mit der SPS und dem Agenten-Server. Zusätzlich werden die Daten der SPS während der Anwendung gespeichert.
- Netzwerkmonitor: Diese Komponente dokumentiert die Kommunikation zwischen SCADA-Server und SPS.
- Prüfer: Diese Komponente enthält eine Logik, die abhängig von der empfangen Nachricht des Agentenservers eine bestimmte Komponente der SPS anfragt.
- Agenten-Server: Verwaltung des Anfrageprozesses, ausgelöst durch den Netzwerkmonitor. Enthält Vergleichsroutinen für den Vergleich der empfangenen Daten.

Das folgende Schaubild zeigt die Struktur der Demonstration:

Lösungsansatz: Schaubild des Demonstrationsnetzwerkes einfügen

Die versendeten Kommunikationspakete haben die folgende Struktur:

Lösungsansatz: Bild des Nachrichtenformates einfügen

Der Prozessablauf des SPS-Paketes simuliert in einem einfachen Beispiel den Prozess einer SPS. Dabei wird in einer definierten Zykluszeit ein definiertes „Anwenderprogramm“ ausgeführt, welches die Werte im E/A-Array durch simple mathematische Operationen modifiziert. Empfängt die SPS eine Nachricht des WinCC-Servers werden die Anpassungen der Werte vorgenommen. Das Modell orientiert sich dabei an der echten Adressierung einer SPS. So werden E/A-Werte in verschiedenen Arrays gespeichert, die durch einen Zahlenwert (Slot-Number) adressiert werden können.

Der SCADA-Server sendet in regelmäßigen Abständen Änderungen für E/A-Daten an die SPS. Die Modifikation der Daten wird dabei durch eine mathematische Funktion zwischen den Write-Nachrichten verändert. Ein Zufallszahlengenerator berechnet dabei die Chance einer Negation des entsprechenden Wertes. Diese Modifikation stellt den Manipulationsversuch einer Malware da.

Da die Simulation einer weiteren Komponente in Form eines Switches weiteren Aufwand erfordert hätte, wird bei der Anwendung der Kommunikationsroutinen die Nachricht jeweils an die SPS bzw. den SCADA-Server und jeweils auch an den Nachrichtenmonitor gesendet. Dieser prüft den enthalten Wert auf das Vorzeichen. Bei einem negativen Wert wird eine Nachricht an den Agentenserver versendet, die die Werte der verdächtigen Nachricht enthält.

Der Agentenserver lauscht auf dem Socket. Sobald eine Nachricht des Netzwerkmonitors eingeht, wird eine Nachricht mit der entsprechenden Speicherstelle an den Prüfer gesendet. Dieser fragt bei der SPS den Wert der entsprechenden Quelle ab und leitet diesen Wert an den Agenten-Server zurück.

Der Agentenserver fordert den aktuellen Wert der Speicherstelle von dem SCADA-System an und vergleicht die Werte.

LösungsansatzDemo: Falls noch Zeit ist, Wireshark Bild einfügen

Der vollständige Quellcode ist auf der DVD im Einband der Masterarbeit hinterlegt.

7.4 Zusammenfassung

In diesem Kapitel wurden Hypothesen und Schlussfolgerungen bzgl. eines möglichen Ansatzes für die Protokollierung von Datenänderung auf einer SPS ausgeführt. Basierend auf diesen Kriterien wurden ein Lösungsansatz und eine demonstrativen Umsetzung des Ansatzes beschrieben.

Kapitel 8

Fazit

Die Analyse der Netzwerke und die Durchführung des Vergleiches haben Unterschiede in den verfügbaren Informationsbereichen aufgezeigt. Die Informationsmengen wurden nach Systemen und Kommunikationsprotokollen unterteilt und basierend auf ihrer Zugehörigkeit zu definierten Kategorien verglichen. Der Vergleich offenbarte Unterschiede in der Detailtiefe der verfügbaren Informationen, u.a. im Bereich der Dokumentation von Parameter- und Bausteinänderungen der untersuchten SPS. Dies konnte durch eine Analyse des Stuxnet Angriffes als signifikante Informationen für die Erkennung und Eingrenzung eines Malware-Angriffes mit Charakteristiken des Stuxnet-Angriffes bzgl. der Manipulation von SPSen eingestuft werden. Ein Lösungsansatz, der eine Prüfung von Änderungen bestimmter Daten auf der Grundlage von Verdachtsereignissen darstellt, wurde beschrieben und demonstriert.

Zusammenfassend konnte die ursprüngliche These für die gewählte Modellbasis belegt werden. Für eine allgemeinere Aussage müssten allerdings weitere Analysen bzgl. anderer Systemkombination und Hersteller vorgenommen werden. Erfolgversprechend könnte dies etwa für eine Analyse weiterer SPSen der Firma Siemens oder konkurrierender Produkte (z.B. von Rockwell Automation) bzgl. der gefundenen Unterschiede sein.

Die Ausführung des Vergleiches hat Schwierigkeiten im Bereich der Herstellung von Beziehungen zwischen den verfügbaren Daten und der Zuordnung zu den Kategorien gezeigt. Mögliche Folgearbeiten könnten es sich zur Aufgabe machen, einen genaueren Blick auf einzelne Kategorien zu werfen, weitere Unterschiede in den Details zu finden und diese auf Signifikanz bzgl. dokumentierter Angriffe auf industrielle Fertigungssysteme zu prüfen. Eine weitere interessante Folgefrage stellt sich auch im Bezug der Signifikanz der gefundenen Unterschiede in Bezug auf weitere bekannte Angriffsszenarien und potentielle Erkenntnisse und weitere Lösungsansätze für die Schließung der Informationslücken.

Literaturverzeichnis

[1] admin. Was ist der unterschied zwischen scada und hmi? "http://www.indusoft.com/blog/2013/04/19/unterschied-zwischen-scada-und-hmi", 2013. Last visited on 20. May. 2018.

[2] Admin. Centos / redhat : Beginners guide to log file administration. "https://www.thegeekdiary.com/centos-redhat-beginners-guide-to-log-file-administration/", N/A. Last visited on 05. June. 2018.

[3] Siemens AG. Simatic hmi wincc v7.2 getting started. "https://cache.industry.siemens.com/dl/files/596/73505596/att_77660/v1/GettingStarted_en-US.pdf", 2013. Last visited on 05. June. 2018.

[4] Siemens AG. Wincc v7.3 - working with wincc. "https://cache.industry.siemens.com/dl/files/925/102754925/att_62020/v1/WinCC_Working_with_WinCC_en-US_en-US.pdf", 2014. Last visited on 05. June. 2018.

[5] Siemens AG. Simatic s7-1200 automatisierungssystem betriebshandbuch. "ï\T1\guillemotright\T1\textquestiondownhttps://support.industry.siemens.com/cs/document/109741593/simatic-s7-s7-1200-automatisierungssystem?dti=0&dl=de&lc=en-WW", 2016. Last visited on 05. June. 2018.

[6] Siemens AG. Die simatic s7-1200 baugruppen im Überblick. "https://www.siemens.com/global/de/home/produkte/automatisierung/systeme/industrie/sps/s7-1200.html", 2018. Last visited on 05. June. 2018.

[7] Siemens AG. Simatic wincc v7 - produktinformation. "https://mall.industry.siemens.com/mall/de/WW/Catalog/Products/10042373?tree=CatalogTree", 2018. Last visited on 0ß5. June. 2018.

[8] Sadequl Hussain Amy Echeverri. Linux logging basics. "https://www.loggly.com/ultimate-guide/linux-logging-basics/", N/A. Last visited on 05. June. 2018.

[9] Apache. Apache-kernfunktionen - loglevel-direktive. "https://httpd.apache.org/docs/2.4/mod/core.html#loglevel", 2018. Last visited on 05. June. 2018.

[10] Verschiedene Autoren. Logdateien. "https://wiki.ubuntuusers.de/Logdateien/", 2018. Last visited on 05. June. 2018.

[11] S. Bhatt, P. K. Manadhata, and L. Zomlot. The operational role of security information and event management systems. *IEEE Security Privacy*, 12(5): 35–41, Sept 2014. ISSN 1540-7993.

[12] Tony Campbell. *Protection of Systems*, pages 155–177. Apress, Berkeley, CA, 2016. URL https://doi.org/10.1007/978-1-4842-1685-9_10.

[13] Christoph. Howto: linux logfiles. "http://www.linux-community.de/fragen/howto-linux-logfiles/", 2003. Last visited on 05. June. 2018.

[14] Conrad Constantine. What kind of logs do you need for an effective siem implementation? "https://www.alienvault.com/blogs/security-essentials/what-kind-of-logs-for-effective-siem-implementation", 2014. last visited on 20. Mai. 2018.

[15] Craig Guyer Ed Macauley. Sql server audit action groups and actions. "https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-action-groups-and-actions?view=sql-server-2017", 2016. Last visited on 05. June. 2018.

[16] Jeff Edwards. 7 siem and security analytics vendors to watch in 2017. "https://solutionsreview.com/security-information-event-management/7-siem-and-security-analytics-vendors-to-watch-in-2017", 2016. Last visited on 20. May. 2018.

[17] PROFIBUS Nutzerorganisation e.V. (PNO). Profinet systembeschreibung. "https://de.profibus.com/downloads/profinet-technology-and-application-system-description/", 2014. Last visited on 05. June. 2018.

[18] Feldbusse.de. Profibus. "http://www.feldbusse.de/Profibus/profibus.shtml", 2015. Last visited on 05. June. 2018.

[19] Feldbusse.de. Profibus - buszugriffsprotokoll (fdl). "http://www.feldbusse.de/Profibus/Buszugriffsprotokoll.shtml", 2015. Last visited on 05. June. 2018.

[20] Feldbusse.de. Profibus-dp. "http://www.feldbusse.de/Profibus/Profibus-DP.shtml", 2015. Last visited on 05. June. 2018.

[21] Feldbusse.de. Profibus-dpv1. "http://www.feldbusse.de/Profibus/DPV1.shtml", 2015. Last visited on 05. June. 2018.

[22] Feldbusse.de. Industrial ethernet - status und ausblick. "http://www.feldbusse.de/trends/status-ethernet.shtml", N/A. Last visited on 20. May. 2018.

[23] Feldbusse.de. Vergleich der industrial-ethernet-systeme. "http://www.feldbusse.de/Vergleich/vergleich_ethernet.shtml", N/A. Last visited on 20. May. 2018.

[24] Max Felser. Profibus handbuch - ms1 verbindung. "https://www.profibus.felser.ch/ms1_verbindung.html", 2017. Last visited on 24. June. 2018.

[25] Max Felser. Profibus handbuch - stationsübergreifende dienste. "https://www.profibus.felser.ch/stationsubergreifende_dienste.html", 2017. Last visited on 24. June. 2018.

[26] Max Felser. Profibus handbuch - dienste für die datenübertragung. "https://www.profibus.felser.ch/datenubertragung_1.html", 2017. Last visited on 24. June. 2018.

[27] Max Felser. Profibus handbuch - telegrammformate. "https://www.profibus.felser.ch/telegrammformate.html", 2017. Last visited on 24. June. 2018.

[28] Max Felser. Profibus handbuch - initialisierung eines dp-slave. "https://www.profibus.felser.ch/initialisierung_eines_dp-slave.html", 2017. Last visited on 24. June. 2018.

[29] Max Felser. Profinet handbuch - protokolle und dienste. "https://www.profinet.felser.ch/protokolleunddienste.html", 2018. Last visited on 24. July. 2018.

[30] Bundesamt für Sicherheit in der Informationstechnik. Industrial control system security - top 10 bedrohungen und gegenmaßnahmen 2016. "https://www.allianz-fuer-cybersicherheit.de/ACS/DE/_/downloads/BSI-CS_005.pdf?__blob=publicationFile", 2016. Last visited on 20. May. 2018.

[31] Bundesministerium für Wirtschaft und Energie. Was ist industrie 4.0? "https://www.plattform-i40.de/I40/Navigation/DE/Industrie40/WasIndustrie40/was-ist-industrie-40.html", 2018. Last visited on 20. May. 2018.

[32] Will Gibb. Openioc series: Investigating with indicators of compromise (iocs) – part i. "https://www.fireeye.com/blog/threat-research/2013/12/openioc-series-investigating-indicators-compromise-iocs.html", 2013. Last visited on 05. June. 2018.

[33] Jürgen Gutekunst. *Schnittstellen, Bussysteme und Netze*, pages 687–744. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017. URL https://doi.org/10.1007/978-3-662-54214-9_15.

[34] Craig Guyer, Ed Macauley, Sudeep Kumar, Bruce Hamilton, Saisang Cai, Rick Byham, and Gene Milener. Sql server audit (database engine). "https://docs.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-database-engine?view=sql-server-2017", 2016. Last visited on 05. June. 2018.

[35] Alexander Hanel. An intro to creating anti-virus signatures. "http://hooked-on-mnemonics.blogspot.de/2011/01/intro-to-creating-anti-virus-signatures.html", 2011. Last visited on 05. June. 2018.

[36] Rainer Hönle. *Speicherprogrammierbare Steuerungen*, pages 745–792. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017. ISBN 978-3-662-54214-9. URL https://doi.org/10.1007/978-3-662-54214-9_16.

[37] IPC2U. Hmi. "https://ipc2u.de/catalog/automatisierungstechnik/hmi/", N/A. Last visited on 20. May. 2018.

[38] ITWissen.info. Scada (supervisory control and data acquisition). "https://www.itwissen.info/SCADA-supervisory-control-and-data-acquisition-SCADA-Protokoll.html", N/A. Last visited on 20. May. 2018.

[39] D. Jayathilake. Towards structured log analysis. In *2012 Ninth International Conference on Computer Science and Software Engineering (JCSSE)*, pages 259–264, May 2012.

[40] Wootae Jeong. *Sensors and Sensor Networks*, pages 333–348. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-540-78831-7. URL https://doi.org/10.1007/978-3-540-78831-7_20.

[41] Kerneltalks.com. 11 log files you should see on your linux system. "https://kerneltalks.com/troubleshooting/11-log-files-you-should-see-on-your-linux-system/", 2017. Last visited on 05. June. 2018.

[42] Heinrich Kersten, Gerhard Klett, Jürgen Reuter, and Klaus-Werner Schröder. *Risikomanagement*, pages 39–62. 09 2016. ISBN 978-3-658-14693-1.

[43] Elektronik Kompendium. Iso/osi-7-schichtenmodell. "https://www.elektronik-kompendium.de/sites/kom/0301201.htm", N/A. Last visited on 20. May. 2018.

[44] M. Krotofil and D. Gollmann. Industrial control systems security: What is happening? In *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, pages 670–675, July 2013.

[45] Marc M. Lankhorst. *Introduction to Enterprise Architecture*, pages 1–10. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017. ISBN 978-3-662-53933-0. doi: 10.1007/978-3-662-53933-0_1. URL https://doi.org/10.1007/978-3-662-53933-0_1.

[46] Stephen Lawton. A guide to security information and event management. "http://www.tomsitpro.com/articles/siem-solutions-guide,2-864-2.html", 2015. Last visited on 20. May. 2018.

[47] David LeBlanc. Overview of microsoft sql server 2012. "https://www.microsoftpressstore.com/articles/article.aspx?p=2201648", 2013. Last visited on 05. Junes. 2018.

[48] David LeBlanc. Overview of microsoft sql server 2012 - database engine. "https://www.microsoftpressstore.com/articles/article.aspx?p=2201648&seqNum=2", 2013. Last visited on 05. June. 2018.

[49] David LeBlanc. Overview of microsoft sql server 2012 - security subsystem. "https://www.microsoftpressstore.com/articles/article.aspx?p=2201648&seqNum=4", 2013. Last visited on 05. June. 2018.

[50] Petra Linke. *Grundlagen zur Automatisierung*, pages 1–28. Springer Fachmedien Wiesbaden, Wiesbaden, 2017. ISBN 978-3-658-17582-5. URL https://doi.org/10.1007/978-3-658-17582-5_1.

[51] Marcel. 12 critical linux log files you must be monitoring. "https://www.eurovps.com/blog/important-linux-log-files-you-must-be-monitoring/", 2018. Last visited on 05. June. 2018.

[52] Raffael Marty. Event processing – normalization. "http://raffy.ch/blog/2007/08/25/event-processing-normalization", 2007. Last visited on 20. May. 2018.

[53] McAffee. Mcafee siem event aggregation. "https://community.mcafee.com/nysyc36988/attachments/nysyc36988/business-documents/457/1/SIEM-Aggregation-AW.pdf", 2016. Last visited on 20. May. 2018.

[54] Microsoft. Description of security events in windows 7 and in windows server 2008 r2. "https://support.microsoft.com/en-us/help/977519/description-of-security-events-in-windows-7-and-in-windows-server-2008", 2011. Last visited on 05. June. 2018.

[55] Microsoft. Event properties. "https://technet.microsoft.com/de-de/library/cc765981(v=ws.11).aspx", 2013. Last visited on 05. June. 2018.

[56] Microsoft. Event logs. "https://technet.microsoft.com/de-de/library/cc722404(v=ws.11).aspx", 2013. Last visited on 05. June. 2018.

[57] Microsoft. Windows filtering platform. "https://msdn.microsoft.com/de-de/library/windows/desktop/aa366510(v=vs.85).aspx", 2018. Last visited on 05. June. 2018.

[58] Microsoft. Remote desktop protocol. "https://msdn.microsoft.com/en-us/library/aa383015.aspx", 2018. Last visited on 05. June. 2018.

[59] Microsoft. Remote desktop protocol: Basic connectivity and graphics remoting. "https://msdn.microsoft.com/en-us/library/cc240445.aspx", 2018. Last visited on 05. June. 2018.

[60] Microsoft. [ms-ssso]: Sql server system overview. "https://msdn.microsoft.com/en-us/library/ff420673(v=sql.105).aspx", 2018. Last visited on 05. June. 2018.

[61] O'Reily. Ssh: The secure shell - inside ssh-2. "https://docstore.mik.ua/orelly/networking_2ndEd/ssh/ch03_05.htm", 2002. Last visited on 05. June. 2018.

[62] Alberto Partida and Diego Andina. *Vulnerabilities, Threats and Risks in IT*, pages 1–21. Springer Netherlands, Dordrecht, 2010. ISBN 978-90-481-8882-6. URL https://doi.org/10.1007/978-90-481-8882-6_1.

[63] Carlos E. Pereira and Peter Neumann. *Industrial Communication Protocols*, pages 981–999. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-540-78831-7. URL https://doi.org/10.1007/978-3-540-78831-7_56.

[64] John Rinaldi. Profinet i&m. "https://www.rtaautomation.com/blog/profinet-im/", 2016. Last visited on 04. July. 2018.

[65] Karen Scarfone. Comparing the best siem systems on the market. "http://searchsecurity.techtarget.com/feature/Comparing-the-best-SIEM-systems-on-the-market", 2015. Last visited on 20. May. 2018.

[66] Karen Scarfone. Comparung the best siem systems on the market. "http://searchsecurity.techtarget.com/feature/Comparing-the-best-SIEM-systems-on-the-market", 2015. Last visited on 20. May. 2018.

[67] The Barking Seal. Understanding dns: Essential knowledge for all it professionals. "https://www.appliedtrust.com/resources/infrastructure/understanding-dns-essential-knowledge-for-all-it-professionals", 2009. Last visited on 20. May. 2018.

[68] Siemens. Basic hmi. "http://w3.siemens.com/mcms/human-machine-interface/de/bediengeraete/basic-hmi/Seiten/Default.aspx", N/A. Last visited on 20. May. 2018.

[69] SoftSelect. Definition mes - manufacturing execution system. "http://www.softselect.de/business-software-glossar/mes-manufacturing-execution-system", N/A. Last visited on 20. May. 2018.

[70] Mark Stingley. Infrastructure security architecture for effective security monitoring. "https://www.sans.org/reading-room/whitepapers/bestprac/infrastructure-security-architecture-effective-security-monitoring-36512", 2015. Last visited on 20. May. 2018.

[71] techopedia. Attack vector. "https://www.techopedia.com/definition/15793/attack-vector", N/A. Last visited on 20. May. 2018.

[72] Alex Teixeira. Get over siem event normalization. "https://medium.com/ateixei/get-over-siem-event-normalization-595fc36559b4", 2017. Last visited on 20. May. 2018.

Abbildungsverzeichnis

Abkürzungsverzeichnis

VM Virtuelle Maschine

KDE K Desktop Environment

SQL Structured Query Language

Bash Bourne-again shell

JDK Java Development Kit

SPS Speicherprogrammierbare Steuerung