hochschule mannheim

Analyse der Netzwerkkommunikation in Industrie 4.0 Umgebungen und Erweiterung einer protoypischen Security Testumgebung zur Darstellung von Bedrohungsfaktoren

Philipp Minges

Bachelor-Thesis

zur Erlangung des akademischen Grades Bachelor of Science (B.Sc.)

Studiengang Informatik

Fakultät für Informatik

Hochschule Mannheim

15.07.2018

Betreuer

Prof. Sachar Paulus, Hochschule Mannheim

Prof. Dr. Maximilian Hauske, Hochschule Mannheim

Minges, Philipp:
Analyse der Netzwerkkommunikation in Industrie 4.0 Umgebungen und Erweiterung einer protoypischen Security Testumgebung zur Darstellung von Bedrohungsfaktoren / Philipp Minges. –
Bachelor-Thesis, Mannheim: Hochschule Mannheim, 2018. 99 Seiten.

Minges, Philipp:
TODO - Title EN / Philipp Minges. –
Bachelor Thesis, Mannheim: University of Applied Sciences Mannheim, 2018. 99 pages.

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Ich bin damit einverstanden, dass meine Arbeit veröffentlicht wird, d. h. dass die Arbeit elektronisch gespeichert, in andere Formate konvertiert, auf den Servern der Hochschule Mannheim öffentlich zugänglich gemacht und über das Internet verbreitet werden darf.

Mannheim, 15.07.2018

Philipp Minges

Abstract

Analyse der Netzwerkkommunikation in Industrie 4.0 Umgebungen und Erweiterung einer protoypischen Security Testumgebung zur Darstellung von Bedrohungsfaktoren

Nach der Einführung des Begriffs „Industrie 4.0“ im Jahr 2011 und dem gleichzeitigen Start der 4. industriellen Revolution werden Kommunikationsnetze in der Industrie immer mehr zur Automatisierung der Produktion von Gütern oder zum unternehmensinternen sowie -externen Datenaustausch genutzt. Um diese Echtzeitkommunikation oder auch Möglichkeiten der Fernwartung zu gewährleisten, werden immer mehr Anlagen mit Netzwerkzugängen ausgestattet. Die Kommunikation der Industrie 4.0 Netze und Systeme findet unternehmensübergreifend über einen unsicheren Kanal statt und kann somit ohne bereitgestellte Sicherheitsmaßnahmen genauso angegriffen werden, wie herkömmliche Heim- oder Büronetzwerke. Das Ziel dieser Arbeit ist es zum einen, die Netzwerkkommunikation zwischen Industrie 4.0 Komponenten anhand aktueller Standards zu analysieren, mögliche Angriffsvektoren darzustellen und deren Eintrittswahrscheinlichkeit sowie Schaden zu bewerten. Zum anderen wird ein vorhandenes Industrie 4.0 Security Testsystem anhand der gewonnenen Erkenntnisse im Bereich der Netzwerksicherheit zu Lehr- und Testzwecken prototypisch erweitert. TODO - neu

TODO - Title EN

TODO - Abstract EN

Inhaltsverzeichnis

1 Einleitung **1**

2 Grundlagen **5**
2.1 Historie 5
2.2 Automatisierungspyramide 8
2.3 Industrie 4.0 9
2.3.1 Internet of Things/Industrial Internet of Things 10
2.3.2 Referenzarchitekturen 10
2.3.3 Kommunikationsstrukturen 15
2.3.4 Protokolle 16
2.3.5 Anforderungen an die Netzwerkkommunikation 20
2.4 TCP/IP Referenzmodell 22
2.5 Security by Design 23
2.6 Testsystem 24

3 Analyse **27**
3.1 Bedrohungen 27
3.2 Integrationsansätze 28
3.2.1 Konsolidierung der Netzwerkkommunikation 29
3.2.2 Gatewaykommunikation 29
3.3 Netzzugangsschicht 30
3.3.1 physikalischer Zugang 31
3.3.2 Virtual Local Area Network (VLAN) 32
3.3.3 vertikale Integration bestehender Komponenten 32
3.4 Internetschicht 33
3.4.1 Address Resolution Protocol (ARP) 34
3.4.2 Quality of Service (QoS) 35
3.4.3 IPsec 36
3.5 Transportschicht 37
3.5.1 Transmission Control Protocol (TCP) 38
3.5.2 User Datagram Protocol (UDP) 43
3.6 Anwendungsschicht 43
3.6.1 Domain Name System (DNS) 44
3.6.2 Dynamic Host Configuration Protocol (DHCP) 51

3.6.3 Open Platform Communications Unified Architecture (OPC UA) 52
3.6.4 Constrained Application Protocol (CoAP) 57
3.7 Zwischenfazit 59

4 Anwendungsszenarien 61
4.1 OPC UA Kommunikation 61
4.2 Man in the Middle (MitM) 62
4.3 Manipulation von ungesichertem Netzwerkverkehr 62

5 Konzept 63
5.1 Verteilungssicht 64
5.1.1 Industrienetzwerk 66
5.1.2 Monitoring-Netzwerk 66
5.1.3 Containernetzwerk 67
5.2 Bausteinsicht 67
5.2.1 Router 68
5.2.2 DHCP Server/DNS Server 69
5.2.3 CoAP Client/CoAP Server 70
5.2.4 Rogue DHCP Server 71
5.2.5 CoAP Manipulationssystem 72
5.2.6 Docker Service 73
5.3 Laufzeitsicht 73
5.3.1 Routing 73
5.3.2 DHCP 73
5.3.3 DNS 74
5.3.4 Kommunikation CoAP Client und Server 75
5.3.5 Manipulation des Netzwerkverkehrs 77
5.4 Anpassungen 78

6 Umsetzung 81
6.1 Softwarewahl 82
6.2 Integration 83
6.2.1 Netzwerkverwaltung 83
6.2.2 CoAP Server 86
6.3 Implementierung 86
6.3.1 OPC UA Secure Channel 87
6.3.2 CoAP Monitoringsystem 87
6.3.3 CoAP Manipulationssystem 88
6.4 Quellcode 88
6.5 Dokumentation 89

7 Validierung 91

8 Fazit 95

9 Ausblick **97**
9.0.1 Defense in Depth 97

Abkürzungsverzeichnis **ix**

Tabellenverzeichnis **xiii**

Abbildungsverzeichnis **xv**

Quellcodeverzeichnis **xvii**

Literatur **xix**

Kapitel 1

Einleitung

Mit der heutigen, immer weiter fortschreitenden Vernetzung von Geräten aus Unternehmensinfrastrukturen und Heimnetzen über das Internet, erfährt die Industrie und deren Wertschöpfung einen strukturellen Wandel. Im Gegensatz zur Industrie 3.0, in der die Kommunikation der Geräte nur innerhalb einer Produktionsstätte oder eines Unternehmens stattgefunden hat, erstreckt sich die Kommunikation in Industrie 4.0 Umgebungen über die Unternehmensgrenzen hinweg. Es werden Konzepte zur Einbindung aller Komponenten eines Firmenprozesses, welcher z. B. Produktion, Service- Instandhaltungsaufgaben beinhaltet, realisiert. Diese Systeme kommunizieren miteinander und nutzen dafür immer häufiger eine Ethernet Netzwerkwerkstruktur. Dies setzt die Produktionsanlagen sowie die genutzten Softwaresysteme den gleichen potentiellen Gefahren durch Viren, Würmer oder Trojaner aus, wie reguläre Büro- oder Heim-PC.

Viele Kritische Infrastrukturen (KRITIS), wie Produktionsanlagen zur Energie- und Wasserversorgung nutzen automatisierte Prozesssteuerungssysteme, Industrie PC (IPC), speicherprogrammierbare Steuerungen (SPS) und Supervisory Control and Data Acquisition (SCADA) Systeme zur Steuerung der Abläufe in den Produktionsanlagen zwischen verteilten Systemen. Die ständige Verfügbarkeit und Überwachung dieser Dienste ist für eine funktionierende Infrastruktur essentiell. Systeme der KRITIS können nicht angehalten werden, um Sicherheitsupdates und einen anschließenden Systemneustart durchzuführen. Bei vielen dieser Prozesssteuerungssystemen wurde der Aspekt der IT-Sicherheit nicht berücksichtigt, da eine Vernetzung der Systeme im heutigen Ausmaß nicht vorgesehen war. Die Systeme bieten keine Möglichkeit der Verschlüsselung des Datenverkehrs oder der Authentifizierung der Benutzer.

Die Sicherheit der Produktionsanlagen und deren Netzwerkkommunikation spielt für ein Unternehmen im Industrie 4.0 Umfeld mit Hinblick auf Verfügbarkeit, Zuverlässigkeit und Authentizität eine essentielle Rolle. Sollte es durch Angriffe möglich sein, die Produktion zu sabotieren oder Anlagen und Systeme zu manipulieren, so können die Folgen schwerwiegend sein. Es kann zu Produktionsausfällen kommen und es können Vertragsstrafen drohen. Ein bekannter Angriff wurde im Jahr 2016 auf das Netz des deutschen Bundestages durchgeführt. Dort wurde ein Zusammenbruch der getroffenen Sicherheitsmaßnahmen erreicht. Es wurden über mehrere Monate unbemerkt sensible Daten entwendet. [TODO - Quelle]

TODO - Kleinere Losgrößen -> von Einzelmaschine zu Fabrik TODO - mehr -> leitfaden-it-security-i40.pdf - Einleitung TODO - Stuxnet,Duqu -> auf Produktionsanlagen zugegriffen

Die beschriebenen Probleme bei der Umsetzung einer sicheren Kommunikation im Industrie 4.0 Umfeld sowie die dargestellten, erfolgreich durchgeführten Angriffe auf bestehende Infrastrukturen bieten mir einen Anlass, den aktuellen Stand der IT-Sicherheit beim Datenaustausch in einer heterogenen Industrie 4.0 Umgebung zu analysieren und mögliche Risiken aufzuzeigen.

Um das erwünschte Ergebnis zu erhalten, muss im ersten Schritt eine Literaturanalyse durchgeführt werden. Mit Hilfe dieser werden die Grundlagen zur Analyse der Kommunikation geschaffen.

Anschließend wird die Sicherheitsanalyse der Netzwerkkommunikation in Industrie 4.0 Umgebungen durchgeführt. Diese beinhaltet die Analyse des Kommunikationsstacks der Netzwerkebene und der verwendeten Protokolle sowie Standards.

Zuletzt werden die Ergebnisse der Analyse durch eine prototypische Implementierung und Erweiterung eines vorhandenen Industrie 4.0 Security Testsystems dargestellt und validiert.

TODO ref. W.A. Halang 2016 und Bundesministerium für Wirtschaft und Energie 2016b und Schleupner 2016 und Lass Sander 2014

TODO - Um die Komplexität zu reduzieren, wird eine umfassende Modularisierung, eine breite Standardisierung und eine durchgängige Digitalisierung benötigt. Diese Anforderungen sind nicht neu, sie sind auch nicht revolutionär sondern die Folge einer permanenten Weiterentwicklung. Diese Evolution ist ein langjähriger Prozess, der schon lange begonnen hat und es existieren bereits Lösungen für viele der nachfolgend skizzierten Anforderungen, die unter anderem auch die zentralen

Grundbausteine für Industrie 4.0 sind. (ref. OPC UA - Wegbereiter der Industrie 4.0)

TODO - Umschreiben

a protocol used between components in the operation of an industrial facility at multipl e levels: from high-level enterprise management to low-level direct process control of a device. The use of OPC UA for enterprise management involves dealings with customers and suppliers. It may be an attractive target for industrial espionage or sabotage and may also be exposed to threats through untargeted malware, such as worms, circulating on public networks. Disruption of communications at the process control could result in financial losses, affect employee and public safety or cause environmental damage.

Kapitel 2

Grundlagen

2.1 Historie

Seit dem Beginn des Industriezeitalters um 1800, welches mit der Mechanisierung (Industrie 1.0) startete, befindet sich die Industrie in einem stetigen Wandel. Sie entwickelte sich um 1900 durch die Massenproduktion zur Industrie 2.0 und in den 1970er Jahren durch die Automatisierung zur Industrie 3.0. Die Einteilung der Industriezeitalter ist durch tiefgreifende Veränderungen im technologischen Fortschritt möglich, welche auch als industrielle Revolution bezeichnet werden. Aktuell befinden wir uns in der Phase der 4. industriellen Revolution.

Die 1. industrielle Revolution fand mit der Erfindung der Dampfmaschine statt. Sie ermöglichte es Eisenbahnen und Dampfschiffe sowie verschiedene Maschinen im Kohleabbau oder in Textilfabriken anzutreiben und trug massiv zur Industrialisierung und der Entstehung der Industrie 1.0 bei. Nach und nach wurden immer mehr Produktionsanlagen errichtet und somit Arbeitsplätze in Infrastruktur, Textilfabriken, Häuserbau, Kohleabbau und anderen Bereichen geschaffen.

Die Erforschung der Elektrizität im 19. Jahrhundert war der Auslöser der 2. industriellen Revolution. Nachdem ab 1830 die Gesetze der Elektrotechnik bekannt waren, fand die Elektrizität eine breite Anwendung in der Industrie und im Alltag. Im Jahr 1913 führte Henry Ford das Fließband in der Automobilbranche ein. Im Zuge dessen musste jeder Arbeiter nur noch einen Arbeitsschritt erledigen, welches einerseits die Produktion wesentlich beschleunigte und eine Massenproduktion ermöglichte und andererseits eine hohe Spezialisierung der einzelnen Arbeitskräfte für ihre bestimmte Aufgabe erforderte. Außerdem wurde es durch die Luftfahrt

möglich Produkte wie Autos, Kleidung und Lebensmittel über Kontinente hinweg immer schneller zu transportieren und zu handeln.

Die 3. industrielle Revolution fand in den 1970er Jahren statt. Sie ist durch eine sukzessive (Teil-) Automatisierung der Prozesse und durch den Einzug der IT in die Industrie- und Verbraucherwelt geprägt. In den 1940er Jahren wurden die ersten Rechenmaschinen und programmierbare Steuerungen in Unternehmen eingesetzt. In den 1970er Jahren zog der Computer auch in den Privatbereich ein, wurde zunehmend beliebter und schaffte einen neuen Industriezweig. Der Fertigungsprozess in Fabriken wurde mehr und mehr von Maschinen übernommen. Durch den zunehmenden Einsatz von IT in Unternehmen entstand immer mehr Kommunikation zwischen Menschen und Maschinen. Diese Kommunikation und die anfallenden Daten wurden jedoch nur unternehmensintern verarbeitet. Es gab nur wenige Schnittstellen nach außen.

Das Ende des 20. Jahrhunderts gilt als der Beginn der 4. industriellen Revolution. Das Kennzeichen dieser Phase ist die zunehmende Digitalisierung und der Einzug der Internet-Technologien in die Industrie. Mit ihr geht die technische Vernetzung physischer Gegenstände, dem Internet of Things (IoT), einher. Mehr und mehr Geräte oder Gegenstände besitzen die Möglichkeit aktiv über eine Netzwerkschnittstelle oder passiv mit Hilfe eines Bar- oder QR-Codes mit der digitalen Welt zu kommunizieren und somit eine fortschreitende Automatisierung und Individualisierung zu ermöglichen. Diese Entwicklung macht es möglich immer schneller Informationen auszutauschen, größere Datenmengen zu analysieren und diese zu verarbeiten. In der Industrie entstehen dadurch u. a. die folgenden Chancen:

- Die Kommunikationsinfrastruktur wird in Zukunft in Produktionssystemen so preiswert sein, dass sie sinnvoll für Konfiguration, Service, Diagnose, Bedienung und Wartung genutzt werden kann.
- Die Produktionssysteme werden mehr und mehr mit einem Netz verbunden, erhalten dort eine digitale Identität, werden somit such- und analysierbar und besitzen die Möglichkeit Daten über sich selbst zu veröffentlichen.
- Maschinen und Anlagen speichern ihre Zustände in ihrer digitalen Identität im Netz. Diese Zustände sind aktuell, aktualisierbar und zunehmend vollständig. Sind im Netzwerk viele solcher Identitäten vorhanden, können die Daten effizient abgerufen und ausgetauscht werden.

- Softwaredienste werden über das Netz verknüpft und können somit automatisiert individuelle Aufgaben durch die direkte Kommunikation der Systeme erledigen. Eine solche individuelle Wertschöpfung war bisher nur unwirtschaftlich oder gar nicht möglich.

Im Gegensatz zur Industrie 3.0 sollen Maschinen autonom, auch über Unternehmensgrenzen hinweg, miteinander kommunizieren können um gesamte Geschäftsprozesse zu übernehmen. Dies setzt eine Öffnung der Unternehmen nach außen voraus und wird in Abbildung 2.1 dargestellt.

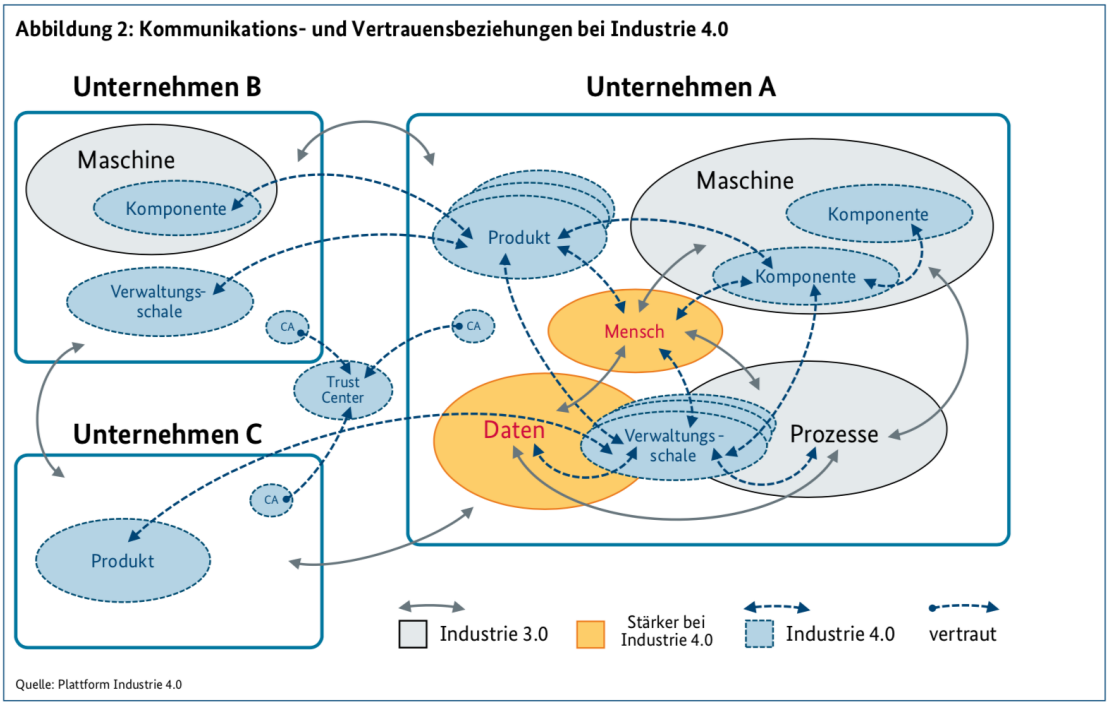


Abbildung 2.1: Kommunikationsbeziehungen in einer Industrie 4.0 Umgebung

2.2 Automatisierungspyramide

Die Automatisierungspyramide (Abbildung 2.2) stellt die beteiligten Systeme und Softwarekomponenten eines automatisierten Prozesses dar. In der Industrie 4.0 wird eine automatisierte und direkte Kommunikation zwischen allen Ebenen der Automatisierungspyramide angestrebt. Die beteiligten Systeme beginnen, ausgehend vom Kundenauftrag und der betriebswirtschaftlichen Planung der Produktion auf der Unternehmensebene beim Enterprise Resource Planning (ERP) System. Die Ergebnisse der Planung werden an das Manufacturing Execution System (MES) übergeben, welches die verschiedenen Fertigungs- oder Logistikaufträge generiert. Die Aufträge werden anschließend auf der Prozessleit- (SCADA), Steuerungs- (SPS) und Feldebene (Ein-/Ausgangssignale) mit Hilfe von Steuerungen und Sensoren bearbeitet. Während die oberen Schichten der Pyramide (ERP und MES) durch Standardkomponenten bzw. -software der IT realisiert werden, zählen die unteren Schichten (Prozessleit- bis Feldebene) zur Automatisierung, welche die Steuerung und Kontrolle der technischen Anlagen übernimmt. Sie sind durch spezielle Hard- und Softwarelösungen umgesetzt. Die Integration von Sicherheitsmaßnahmen bei der Kommunikation dieser Systeme stellt oft eine große Herausforderung dar, da besondere Anforderungen vorliegen oder wenig Ressourcen zur Verfügung stehen.

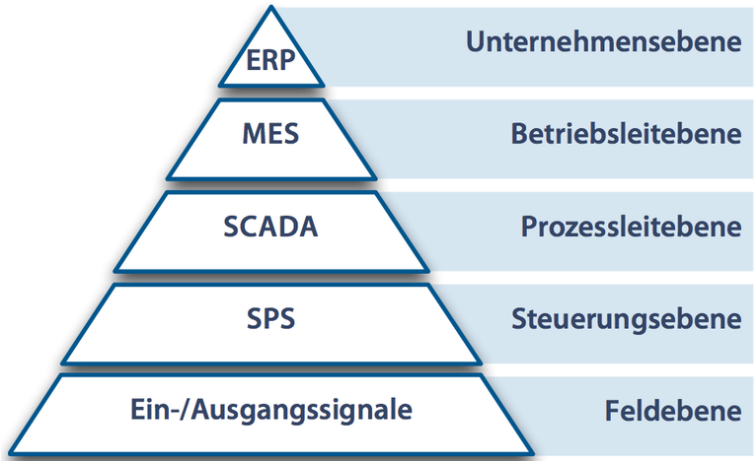


Abbildung 2.2: Automatisierungspyramide - TODO ref. Langmann,2004

2.3 Industrie 4.0

Der Begriff Industrie 4.0 wurde erstmals auf der Hannover Messe 2011 verwendet (Drath 2014) und soll das Ergebnis der 4. industriellen Revolution darstellen. Der Grundgedanke hinter Industrie 4.0 ist die flächendeckende Vernetzung von Informations- und Kommunikationstechnik zu einem Internet der Dinge, Dienste und Daten (Dieter Spath, Oliver Ganschar, Stefan Gerlach, Moritz Hämmerle, Tobias Krause, Sebastian Schlund 2013). Diese Vernetzung soll einen ständigen Informationsaustausch zwischen den Komponenten ermöglichen. Jede Komponente des IoT soll als Cyber-physisches System (CPS) arbeiten. Ein CPS besitzt neben seiner realen Identität eine digitale Identität, über welche es ständig mit anderen IoT-Geräten kommunizieren kann. Kunden- und Maschinendaten werden miteinander vernetzt (Plattform Industrie 4.0 2016). Dieser Prozess beschreibt auch einen Wandel in der Strukturierung und Organisation der Produktion in Unternehmen. Durch die fortschreitende Automatisierung wird die Umsetzung einer immer höheren Individualisierung bei geringerer produzierter Stückzahl rentabel. Abbildung 2.3 zeigt die Vernetzung der verschiedenen Industriesektoren und Komponenten über das IoT.

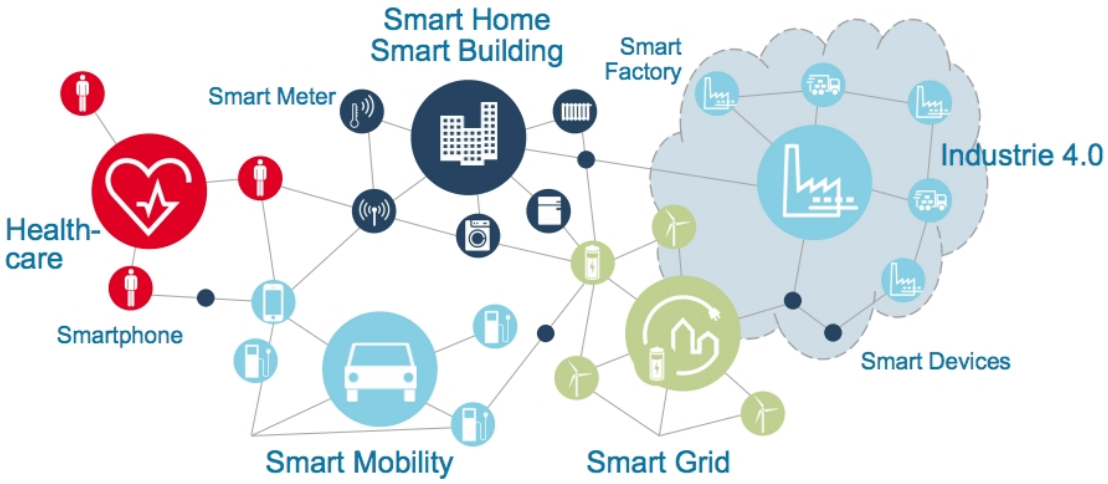


Abbildung 2.3: Das Internet der Dinge

2.3.1 Internet of Things/Industrial Internet of Things

Die fortschreitende Vernetzung der Komponenten spiegelt sich im IoT bzw. Industrial Internet of Things (IIoT) wieder. Das IoT beschreibt im Gegensatz zum IIoT ein verbraucherorientiertes Konzept für die Nutzung von digitalisierten und vernetzten Systemen. Hierbei werden die physischen Systeme virtuell abgebildet. Dies wird genutzt, um die Effektivität der Systeme zu verbessern und intelligente Services zu nutzen.

Das IoT ist ein wesentlicher Bestandteil der Industrie 4.0, welche Netzwerke aus Systemen, Daten und Dienstleistungen herstellt, in denen diese Komponenten miteinander kommunizieren. Im Verbraucherbereich und für die Kommunikation zwischen Mensch und Maschine findet das Protokoll Hypertext Transfer Protocol (HTTP) und dessen Representational State Transfer (REST) Programmierparadigma breite Anwendung.

Das IIoT beschreibt den Gebrauch von IoT-Technologien im industriellen Raum. Diese Systeme können besondere Anforderung an die Kommunikation im Netzwerk wie Skalierbarkeit, Ressourcenverbauch, Echtzeitkommunikation oder Sicherheit stellen. Des Weiteren findet in Industrie 4.0 Umgebungen Machine to Machine (M2M) Kommunikation statt. Um diesen Problemen entgegenzuwirken, wurden neue Protokolle zur Übermittlung von Daten im Netzwerk entwickelt. Hierbei erfahren vor allem die Protokolle Message Queue Telemetry Transport (MQTT) und CoAP ein hohes Maß an Beachtung. Diese Protokolle wurden für eine ressourcenschonende Kommunikation zwischen Maschinen entwickelt.

2.3.2 Referenzarchitekturen

Um eine flächendeckende Vernetzung der digitalen Komponenten zu ermöglichen, muss eine einheitliche Kommunikation geschaffen werden. Diese beschränkt sich nicht nur auf die Form der Nachrichten im Netzwerk und deren Gewährleistung der Sicherheit, sondern beinhaltet auch die Struktur und Bereitstellung von Informationen im Netzwerk. In der Folge wurden verschiedene Referenzarchitekturmodelle entwickelt, um Standards für die Kommunikation und Interaktion von Netzwerkkomponenten innerhalb einer Industrie 4.0 Umgebung zu definieren.

Referenzarchitekturmodell Industrie 4.0 (RAMI4.0)

Das RAMI4.0 wird in der DIN SPEC 91345 beschrieben und dient als Konzept zur strukturierten Umsetzung der grundlegenden Idee hinter dem Begriff Industrie 4.0. Die Aufgabe des Architekturmodells ist es, die Ziele einer Industrie 4.0 Umgebung, die vollständige Vernetzung der physischer Wertgegenstände, umzusetzen. Diese Gegenstände werden im Rahmen der RAMI4.0 als *Assets* bezeichnet. Jedes *Asset* besitzt seine eigene Verwaltungsschalte, welche als Schnittstelle zum Austausch von Informationen dient. Die Verwaltungsschale soll eine standardisierte Kommunikation und einfache Inbetriebnahme neuer Komponenten ermöglichen (Plattform Industrie 4.0 2016). Mit Hilfe der RAMI4.0 soll es möglich den Status eines *Assets* zu jedem Zeitpunkt im Lebenszyklus nachweisen zu können. Die RAMI4.0 ist in Abbildung 2.4 dargestellt und wird durch ein Modell aus sechs Schichten und drei Achsen dargestellt. (DIN SPEC 2016)

Auf der Architekturachse werden sechs Schichten beschrieben. Auf der untersten Schicht wird der Gegenstand der physischen Welt dargestellt. Alle zu ihm relevanten Information werden in den darüberliegenden Schichten gespeichert. Darüber stellt die *Integration* Schicht das Bindeglied zwischen der physischen und digitalen Welt bereit, indem sie die Eigenschaften des *Assets* für Computersysteme erreichbar macht (Bundesministerium für Wirtschaft und Energie 2016a). Die *Communication* Schicht beschreibt den Zugriff auf die Ressourcen und Funktionen der Komponente und stellt die Dienste einer Service Oriented Architecture (SOA) bereit. Auf der *Information* Schicht wird die Funktionalität des *Assets* gespeichert und die Datenintegrität gewährleistet. Die *Functional* Schicht beschreibt die Form, wie und mit welchen Parametern ein Funktionsaufruf stattfinden kann. In der *Business* Schicht werden die geschäftsrelevanten Daten gehalten. (DIN SPEC 2016)

Die Hierarchieachse zeigt die Anlagen, Maschinen sowie das Endprodukt, welche miteinander Vernetzt sind. Die in Abschnitt 2.2 beschriebene Automatisierungspyramide findet sich in der Hierarchieebene der RAMI4.0 wieder. Sie wurde dort auf der niedrigsten Ebene um das Produkt (*Product*) sowie auf der höchsten Ebene um die Stufe *Connected World* erweitert. Die *Connected World* beschreibt den Zusammenhang zwischen einem Asset oder einer Assetkombination und einem anderen Asset oder einer Assetkombination, also einem Fabrikverbund. (DIN SPEC 2016)

Der Produktlebenszyklus wird im Gegensatz zur Industrie 3.0 in das Netzwerk mit eingebunden. Der gesamte Prozess der Produktion, Wartung bis hin zur Verschrot-

tung wird digital erfasst. Somit könnten ständig Informationen über vorhandene *Assets* gesammelt und zur Optimierung des Werschöpfungsprozesses analysiert werden. (DIN SPEC 2016)

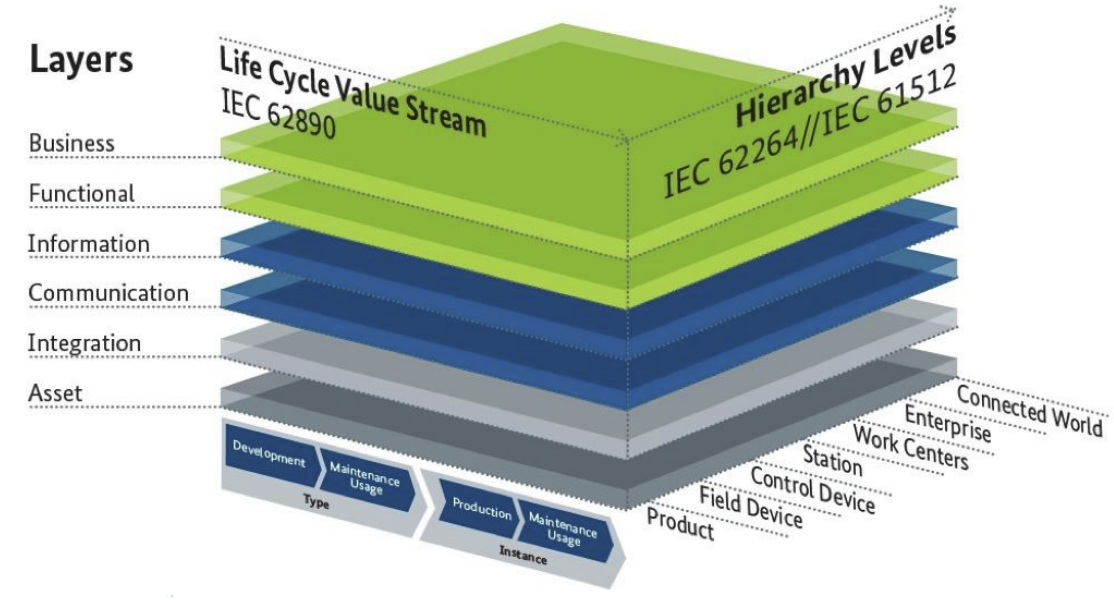


Abbildung 2.4: RAMI 4.0

IIRA

Das Industrial Internet Consortium (IIC) veröffentlichte im Jahr 2015 die Industrial Internet Reference Architetecture (IIRA). Die IIRA beschreibt eine standardbasierte, offene Referenzarchitektur für IIoT, welches auf dem Industrial Internet Architecture Framework (IIAF) basiert. Das IIAF unterstützt die Unternehmen bei der Entwicklung, Dokumentation, Kommunikation und Bereitstellung von Systemen im IIoT Bereich (Industrial Internet Consortium 2017a). Die Beschreibung der Architektur findet mit einem hohen Maß an Abstraktion statt, um das breite Feld der verschiedenen Industrielösungen abdecken zu können und standardisierte Vorgehensweisen zu ermöglichen. Das IIAF folgt der Vorgehensweise des ISO/IEC/IEEE Standard 42010:2011[1]. Hieraus werden die grundlegenden Architekturbeschreibungskonstrukte *Concern*, *Stakeholder* und *Viewpoint* übernommen. Die *Viewpoints* sind die grundlegenden Ebenden beim Aufbau der IIRA. Dabei werden vier *Viewpoints* für die Beschreibung festgelegt. (Dr.-Ing. Mike Heidrich, Dr. Jesse Jijun Lui 2016)

Der *Business Viewpoint* beinhaltet die betriebswirtschaftlichen *Concerns* bei der Umsetzung eines Industrial Internet Systems (IIS) sowie die entstehenden Rahmenbedingungen. Es werden die Systemeigenschafen definiert, welche an die Geschäftsziele gekoppelt sind. Die *Stakeholder* dieses *Viewpoints* bestehen aus Führungskräften, Produktmangern und Systemingenieuren.

Im *Usage Viewpoint* werden die *Concerns* bei der Nutzung eines IIS beschrieben. Dies beinhaltet die Beschreibung der Bedienabläufe.

Der *Functional Viewpoint* beschreibt die funktionalen Komponenten des IIS. Es werden Zusammenhänge, Struktur, Schnittstellen und Interaktionen mit Systemen im Netzwerk sowie der Außenwelt beschrieben.

Der *Implementation Viewpoint* beinhaltet die Technologien zur Umsetzung des IIS. Es werden die funktionalen Komponenten, deren Vernetzung, Kommunikationsschnittstellen sowie deren Produktlebenszyklen dargestellt. Diese *Concerns* sind wichtige Ansatzpunkte für Komponentendesigner, Systementwickler und Integratoren.

[1] ISO/IEC/IEEE Standard 42010:2011 - Systems and Software Engineering–Architecture Description

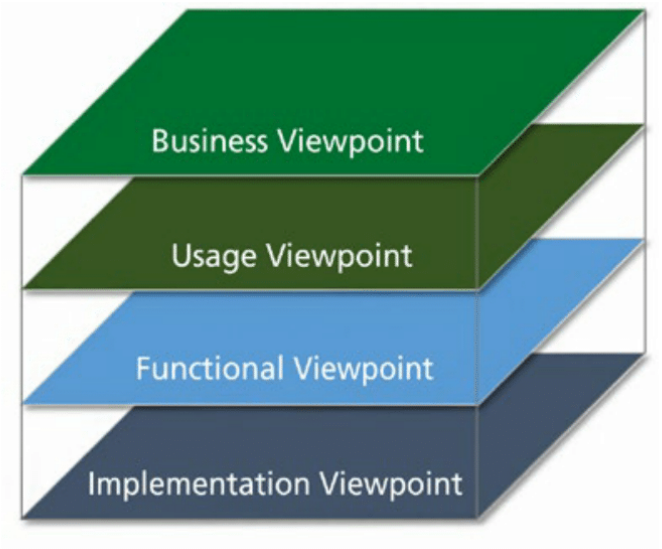


Abbildung 2.5: Die Grundebenen der IIRA

Die Anforderungen des Referenzarchitekturmodells beinhalten niedrige Latenzen und Schwankungen, einen hohen Durchsatz, Skalierbarkeit, Ausfallsicherheit, Datensicherheit und QoS. Die IIRA stellt einen softwaretechnischen Ansatz der Darstellung einer Referenzarchitektur bereit.

2.3.3 Kommunikationsstrukturen

TODO

End2End

Die Komponenten der Industrie 4.0 Umgebung kommunizieren über einen direkten Kanal miteinander. Dies setzt voraus, dass sich beide Teilnehmer in einem Netzwerk befinden, welches die benötigten Dienste wie z. B. Internet Protocol (IP) und DNS zur Kommunikation bereitstellt. Des weiteren müssen beide Systeme diese Dienste und Protokolle unterstützen.

Gateways

Um existierende Systeme, welche selbst nicht Industrie 4.0 konform kommunizieren oder zu wenig Rechenleistung besitzen, in die Industrie 4.0 Welt zu integrieren, werden Industrie 4.0 Gateways genutzt. Dabei ist jedoch zu beachten, dass die Systeme hinter den Gateways nicht als Industrie 4.0 Komponenten entwickelt wurden und somit auch keine oder nur wenige dieser Eigenschaften besitzen. Des Weiteren ist es möglich, dass die Kommunikation aus Leistungsgründen oder besonderer Anforderungen über optimierte, proprietäre Protokolle stattfindet. Die Gateways müssen auf die Systeme und deren Protokolle individuell konfiguriert werden, um die Funktionalitäten im Industrie 4.0 Netz bereitstellen zu können, und die Kommunikation zu schützen.

Publish-Subscribe

Das Publish-Subscribe Modell bietet die Möglichkeit Informationen an mehrere Teilnehmer zu verteilen. Hierbei melden sich die Empfänger beim Verteiler an und

wählen aus, über welche Nachrichtentypen sie informiert werden möchten. Diese Verteildienste nutzen zur besseren Skalierung und Reduzierung der Netzlast häufig Datagramme wie UDP. Durch die Nutzung von Datagrammen geht jedoch die Fehlertoleranz verloren. Somit muss entweder dafür gesorgt werden, dass eine sehr zuverlässige Netzwerkinfrastruktur vorhanden ist und hohe Bandbreitenreserven geschaffen werden, um die Dienstgüte (QoS) sicherzustellen oder dieses Modell nur für fehlertolerante Kommunikation wie z. B. Audio- und Video-Anwendungen oder Businessprozesse zu nutzen.

Kommunikation mit Netzwerk als Partner

Zeitkritische Automatisierungsanwendungen verlangen besondere Netzwerkeigenschaften. Sie können auf Latenz oder Jitter angewiesen sein. Um diese Eigenschaften sicherzustellen, ist es sinnvoll in diese Netze eine Industrie 4.0 Schnittstelle zu integrieren. Somit ist es den Teilnehmern möglich, über die Verwaltungsschale sicherzustellen, dass das Netzwerk die erforderlichen Anforderungen bereitstellt. (Plattform Industrie 4.0 2017)

2.3.4 Protokolle

Die Kommunikation in Industrie 4.0 Umgebungen findet nicht mehr über einzelne, vorgegebene Schnittstellen statt, sondern direkt von den Produktionssystemen, also den unteren Ebenen der Automatisierungspyramide. Um dies zu ermöglichen, ist es notwendig, eine einheitliche Kommunikation durch Normen und Standards herzustellen, um eine unternehmensübergreifende Kommunikation aller Komponenten zu ermöglichen. Durch die in der Industrie 4.0 benötigte M2M Kommunikation wurde die Entwicklung neuer Protokolle zum effizienten Informationsaustausch vorangetrieben, welche es ermöglichen sollen, eine Standardisierung bereitzustellen und somit eine herstellerübergreifende und plattformunabhängige Kommunikation zu ermöglichen. Hierbei haben sich bzgl. der Referenzarchitekturen RAMI4.0 und IIRA die Protokolle OPC UA und Data Distribution Services (DDS) etabliert.

OPC UA

OPC UA ist in der International Electrotechnical Commission (IEC) 62541 als offener Standard definiert und erstreckt sich über Communication- und Information Layer des RAMI4.0. Es vereint Daten- und Informationsdienste und stellt einen sicheren, zuverlässigen und plattformübergreifenden Informationsaustausch zwischen unterschiedlichen Geräten und Systemen der Industrie bereit. Die OPC UA ermöglicht die Kommunikation über die verschiedenen Schichten der Automatisierungspyramide von der Feldebene bis zur Unternehmensebene.

OPC UA wurde ursprünglich als Client-Server Architektur entwickelt. Um OPC UA besser in Systemen der unteren Ebenen der Automatisierungspyramide, wie Kleinsteuerungen, Sensoren und Low-End-Embedded-Systeme, einsetzen zu können, werden meist geringe Latenzen in den Netzwerken und ein geringer Overhead aufgrund von Ressourcenmangel sowie die Kommunikation mit mehreren Partnern benötigt. Diese Anforderungen wurden von dem im Jahr 2018 veröffentlichten 14. Teil der OPC UA Spezifikation *Publish Subscribe* adressiert. Das *Publish Subscribe* Modell wird mit Hilfe des Transportprotokolls UDP umgesetzt und ermöglicht den Multi- und Broadcast sowie die Möglichkeit der häufigen Übertragung von kleinen Datenmengen um *Logging* oder *Monitoring* durchzuführen, ohne das Netzwerk durch einen 3-Wege-Handshake bei jedem Verbindungsaufbau zusätzlich zu belasten (OPC Foundation 2018a).

OPC UA stellt ein Informationsmodell mit Hilfe einer SOA bereit, erfüllt die Anforderungen des RAMI4.0, etabliert sich zunehmend im Maschinen- und Anlagenbau und bietet einen vielversprechenden Ansatz für einen standardisierten Informationsaustausch über Unternehmensgrenzen hinweg (OPC Foundation 2014). Aufgrund dessen stellt es auch ein attraktives Ziel für Industriespionage und die Sabotage von Industrienetzen bereit (OPC Foundation 2018b).

OPC UA wird in 14 geschichteten Spezifikationen beschrieben, welche sich in die Bereiche *Core*, *Access Type* und *Utility* unterteilen lassen. Dabei stellen die Spezifikationen 1-7 sowie 14 die Kernfunktionalitäten des Architekturmodells dar. Sie beschreiben die Struktur des OPC Addressraums und der Dienste, die darauf operieren. Die Spezifikationen 8-11 wenden diese Kernfunktionalitäten auf spezifische Open Platform Communications (OPC COM) Spezifikationen, wie Data Access (DA), Alarms and Events (A&E) und Historical Data Access (HDA) an. Die

Teile 12 und 13 beinhalten Mechanismen zur Discovery von Systemen und beschreiben Möglichkeiten der Datenaggregation.

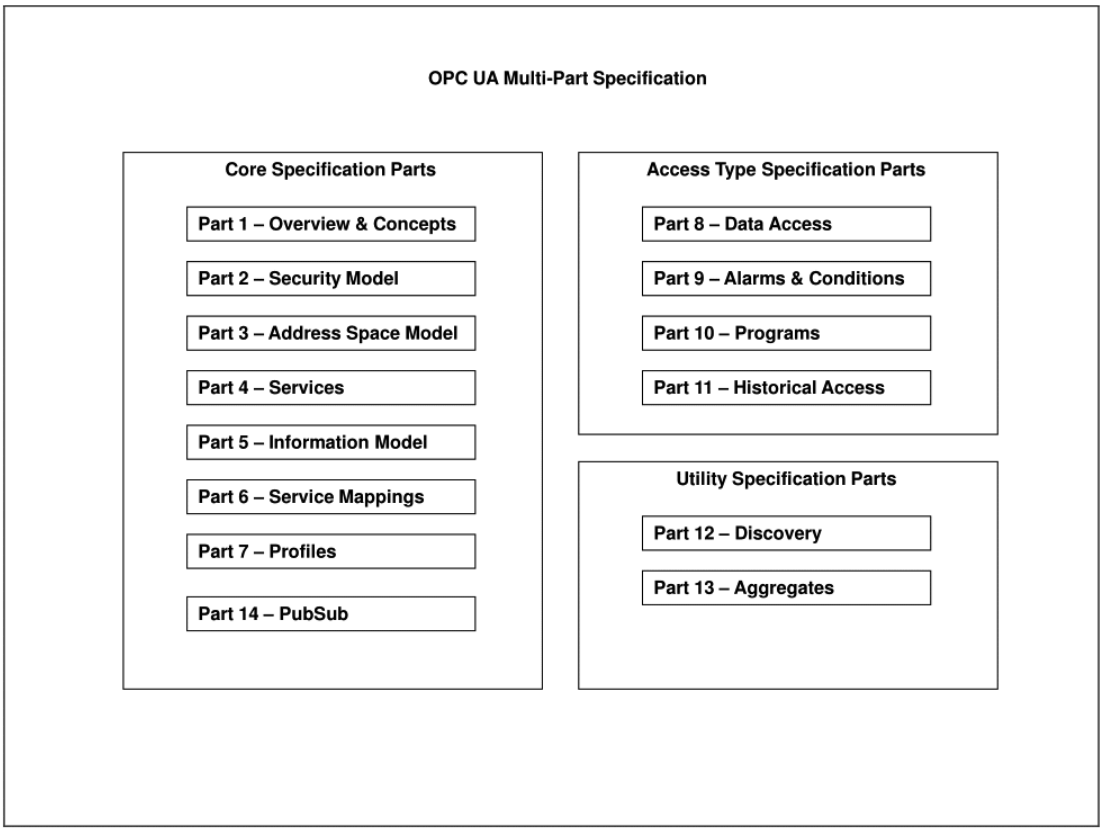


Abbildung 2.6: OPC UA Multi-Part Specification - OPC Foundation 2018a

Das Lesen- und Schreiben von Daten und die Kommunikation in Industrie 4.0 Umgebungen findet nach RAMI4.0 durch die Verwaltungsschale der Komponenten statt. Diese wird im OPC UA Stack durch den Adressraum beschrieben. Der Adressraum wird zur Speicherung von Knoten, deren Attribute und Referenzen zu anderen Knoten genutzt. Der Addressraum und das Informationsmodell von OPC UA werden in den Spezifikationen 3 OPC Foundation 2018c und 5 OPC Foundation 2018d definiert.

OPC UA ermöglicht die Kommunikation der Assets über ein Client-Server Pattern. Die Architektur setzt sich dabei aus einem OPC UA Client und einem OPC UA Server zusammen. Der OPC UA Server stellt verschiedene Funktionen bereit, auf welche der OPC UA Client mit Hilfe eines Request zugreifen kann. Des Weiteren ist es möglich durch einen Request des OPC UA Clients ein Element des Servers beobachten zu lassen, um bei Änderungen vom Server benachrichtigt zu werden. Um die Kommunikation zwischen OPC UA Servern zu gewährleisten, wird ein OPC UA Client in einen OPC UA Server integriert. In der Grafik Abbildung 2.7 wird das Client-Server Pattern der OPC UA Spezifikation schematisch dargestellt. Die linke Seite der Grafik beschreibt die Kommunikation zwischen einem Client und einem Server mit eingebettetem Client. In der rechten Seite der Grafik findet die Kommunikation zwischen dem eingebetteten Client und einem OPC UA Server statt.

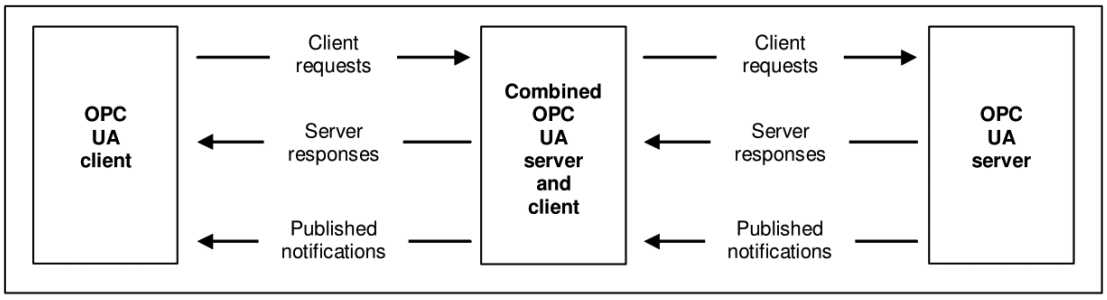


Abbildung 2.7: OPC UA Client-Server Architektur - OPC Foundation 2018a

Im Jahr 2018 wurde der Standard zusätzlich um eine Spezifikation für das Publish-Subscribe Modell erweitert (Hoppe 2018). Das Publish-Subscribe Modell ermöglicht die Nutzung von OPC UA in Wide Area Network (WAN) Umgebungen und die Verwendung von Protokollen wie MQTT und Advanced Message Queuing Protocol (AMQP), während die Ende zu Ende Sicherheit und die standardisierte Datenmodellierung erhalten bleiben. Im Publish-Subscribe Modell wird das fehlertolerante Datagramm UDP als Transportprotokoll verwendet, wodurch geringe Latenzen bei der Kommunikation ermöglicht werden.

DDS

DDS ist ein weiterer offener Standard der Open Management Group (OMG) und stellt eine Message oriented Middleware (MOM) zur Kommunikation in hochdynamischen verteilten Systemen dar. Er wurde für niedrige Latenzzeiten, einen hohen Datendurchsatz und eine skalierbare, belastbase und sichere Datenverteilung entwickelt, um die Kommunikation in Steuerungs- und Kontrollaufgaben zu realisieren. Der beschriebene Standard deckt alle Anforderungen der IIRA ab und hat sich bereits in industriellen Systemen etabliert. Gegenüber OPC UA beschreibt DDS eine dezentralisierte Architektur. Es bietet ein Konnektivitäts-Framework, welches ein Kommunikationsparadigma basierend auf einem Shared Data Model, einen Standard für die Definition domain-spezifischer Informationsmodelle, ein starkes Sicherheitsmodell, Discovery und reichhaltige APIs beinhaltet. Die Kommunikation findet direkt vom Publisher zum Subscriber statt. Dabei werden Latenzzeiten reduziert und durch die Nutzung von Broad- und Multicast die Netzlast beim Bereitstellen von Informationen an viele Empfänger gering gehalten. Es ist möglich die MOM DDS in eine OPC UA Architektur zu integrieren und mit dem Informationsmodell nutzen.

2.3.5 Anforderungen an die Netzwerkkommunikation

Aufgrund der unterschiedlichen Einsatzbereiche von Industrie 4.0 Systemen, unterscheiden sich auch dementsprechend deren Anforderungen. Um eine sichere Kommunikation in diesen Umgebungen bereitzustellen dienen die Grundprinzipien der sicheren Kommunikation. Die Referenzmodelle RAMI4.0 und IIRA beschreiben

ebenfalls drei Anforderungen an den Übertragungskanal: Sicherheit, Verfügbarkeit und QoS (Bundesministerium für Wirtschaft und Energie 2016a).

TODO - grundprizipien der Sicheren Kommunikation kurz dann, auf anforderungen von RAMI und IIRA eingehen.! TODO - in Bezug auf Industrie 4.0 ... oder Anforderungen an Industrie 4.0 Umgebungen basieren auf Grundprinzipien der sicheren Kommunikation Die Grundprinzipien der sicheren Kommunikation beschreiben die Schutzziele im Bereich der Informationssicherheit. Diese verdeutlichen den Anspruch an die Sicherheit an ein zu implementierendes System oder ein Netzwerk. Sie stellen einen vereinbarten Umfang gegen Bedrohungen dar, welcher von den Kommunikationspartnern gewährleistet wird und nachgewiesen werden kann. Diese klassischen Schutzziele sind auch für Industrie 4.0 Umgebungen zutreffend. Die weitreichende Vernetzung der Systeme in der Industrie 4.0 erfordert jedoch weitere Schutzziele, um einen rechtskonformen Umgang oder besondere Anforderungen sicherzustellen.

TODO - Hierunter fallen die Bereiche Netzsicherheit und Datensicherheit, Sichere Identitäten und funktionale Sicherheit. Netzsicherheit und Datensicherheit werden in der AG3 der Plattform Industrie 4.0 adressiert. Die UAG Netzkommunikation arbeitet bzgl. dieser Punkte mit der AG3 zusammen. Zum Thema „Security und funktionale Sicherheit“ arbeitet die AG3 mit dem DKE-TBINK AK IT Security und Security by Design zusammen. Hinsichtlich funktionaler Sicherheit gibt es Anforderungen von Seiten IEC 61784-3. Diese müssen bei der Definition neuer Systeme berücksichtigt werden. - TODO

- Vertraulichkeit/Zugriffsschutz
- (Daten)-Integrität/Änderungsschutz
- Authentizität/Fälschungsschutz
- Verfügbarkeit

Sicherheit

- Netzsicherheit und Datensicherheit
- Sichere Identitäten
- funktionale Sicherheit

Verfügbarkeit

Die ständige Verfügbarkeit von Daten und Diensten spielt in der Industrie 4.0 eine bedeutende Rolle, um den Datenaustausch zwischen zwei Kommunikationspartnern im Netz jederzeit zu ermöglichen. Als Verfügbarkeit wird die Wahrscheinlichkeit bezeichnet, dass ein System innerhalb eines bestimmten Zeitraumes erreichbar ist. Ein System gilt als verfügbar, wenn es erreichbar ist und die für es vorgesehenen Aufgaben erledigen kann.

Die Verfügbarkeit eines Systems wird in Verfügbarkeitsklassen gegliedert. Diese beschreiben Verfügbarkeitswahrscheinlichkeiten von 99% (Verfügbarkeitsklasse 2) bis 99,9999% (Verfügbarkeitsklasse 6). Eine exakte Definition, wann ein System hochverfügbar ist, gibt es nicht - TODO ref. Im Allgemeinen wird ab Verfügbarkeitsklasse 3 (99,99%) von Hochverfügbarkeit gesprochen.

TODO - Verfügbarkeit gewährleisten durch...

QoS

TODO - QoS

2.4 TCP/IP Referenzmodell

TODO - Das TCP/IP Referenzmodell stellt die Basis für moderne Kommunikationsnetze dar. Unternehmensübergreifende Kommunikation in Industrie 4.0 Umgebungen findet im wesentlichen über IP-Netze statt. Diese basieren auf dem TCP/IP Referenzmodell, welches ein Schichtenmodell ist und die vier Schichten der Internetprotokollfamilie beschreibt. Sie setzen sich aus Application-, Transport-, Internet- und Link-Layer zusammen. Die Schichten des TCP/IP Referenzmodells überlagern sich mit den Schichten des ISO/OSI Referenzmodells.

Application Layer

Die Anwendungsschicht ist für die Übertragung der Nutzdaten zwischen verschiedenen Anwendungen zuständig. Dabei kann es sich um entfernte Anwendungen

handeln. Diese sollen sich für den Benutzer verhalten, als würden sie lokal ausgeführt werden.

TODO - Prozess- und Businesslogik

Transport Layer

Die Transportschicht sorgt für die Kommunikation zwischen Prozessen. Die Transportschicht nutzt Ports um verschiedene Dienste zu adressieren. Sie beeinflusst, ob es sich um eine zuverlässige Verbindung (TCP) oder nicht (UDP) handelt.

TODO - End2End Security

Internet Layer

Die Internetschicht wird genutzt, um Daten von einem Teilnehmer im Netzwerk zum anderen zu übertragen. Die Endpunkte im Netzwerk werden durch IP Adressen beschrieben.

Link Layer

Der Bitübertragungsschicht beschreibt die Topologie des Netzwerks. Sie stellt die physikalische Verbindung der Netzwerkteilnehmer zur Verfügung.

TODO - Bild Internetprotokollfamilie TODO - Mit Bild nur kurz erklären und referenzieren, Überschriften entfernen.

2.5 Security by Design

In der Vergangenheit wurden Sicherheitsmechanismen üblicherweise nachträglich und reaktiv in die Entwicklung von Komponenten mit einbezogen. Industrie 4.0 Umgebungen erfordern umfassende Maßnahmen, um die in Unterabschnitt 2.3.5 beschriebenen Schutzziele zu erfüllen und eine sichere Kommunikation zu gewährleisten. Dies gilt vor allem für Maschinenbau- und Fertigungsunternehmen, welche häufig proprietäre Individualsoftware zur Steuerung der Maschinen einsetzen

(DTAG 2016). Aus der Notwendigkeit, Sicherheitsaspekte bereits in die Softwareentwicklung mit einzubeziehen und einen Schutz der Kommunikation zu gewährleisten, hat sich der Begriff *Securiy by Design* entwickelt.

Die Methoden und Ziele der Angreifer stehen unter einem ständigen Wandel. Somit ist es nicht möglich, eine Sicherheitsimplementierung zu entwickeln und diese wiederholt einzusetzen. Vielmehr ist es notwendig, die Sicherheit durch *Security by Design* so weit als möglich proaktiv herzustellen und gleichzeitig im Schadensfall flexibel zu reagieren, um das Schadensausmaß zu begrenzen. Es sind Maßnahmen zur Prävention, Detektion und Reaktion erforderlich (Platform Industrie 4.0 2015).

Das Konzept *Security by Design* wird von RAMI4.0 und IIRA sowie von den darin genutzten Protokollen OPC UA und DDS verfolgt. Die Absicherung der Kommunikation im Netzwerk gehört zu den Kernbestandteilen der Referenzarchitekturen (Industrial Internet Consortium 2017b und OPC Foundation 2018b).

2.6 Testsystem

TODO - kürzer

Die aus der Analyse hervorgehenden möglichen Schwachstellen und Bedrohungen im Bereich der Netzwerksicherheit in Industrie 4.0 Umgebungen und deren Auswirkungen sollen anhand eines vorhandenen, prototypischen Industrie 4.0 Testsystems (Weber 2018) veranschaulicht werden. Das vorhandene System setzt die drei Schichten der Software-Architektur (Verteilungs-, Baustein- und Laufzeitschicht) nach Starke / Hruschka um. Die Netzwerkkommunikation wird über das Protokoll OPC UA realisiert, welches die Anforderungen der Industrie 4.0 und RAMI4.0 umsetzt.

TODO - Architektur Das vorhandene System ist, aufgrund der vorgesehenen Einsatzgebiete Lehre, Integrations- und Sicherheitstests, als virtuelle Maschine (VM) umgesetzt worden. Dies ermöglicht es die Testinfrastruktur vom restlichen Netz zu kapseln. Das Betriebssystem der VM stellt eine Firewall bereit, welche unerwünschten Netzwerktraffic von oder zu dem System verhindert. Um eine gute Erweiterbarkeit der Testumgebung und Modularisierung der Komponenten zu erreichen, werden die einzelnen Industrie 4.0 Komponenten mit Hilfe der Containerlösung Docker isoliert ausgeführt, verwaltet und deren Netzwerkkommunikation sichergestellt. Durch den zusätzlichen Einsatz des Deploymentsystems Kubernetes

wird ein verteiltes Ausführen des Systems ermöglicht und somit eine gute Skalierbarkeit erreicht.

TODO - Komponenten: Repository, Discovery Server, Verwaltungsinterface, Scheduler

Kapitel 3

Analyse

Im folgenden Kapitel wird die Analyse der Netzwerksicherheit in Industrie 4.0 Umgebungen durchgeführt. Zuerst wird eine Beschreibung der Bedrohungen von Industrie 4.0 Systemen durchgeführt. Anschließend werden, aufgrund der bestehenden Infrastruktur und der Heterogenität der Netzwerklandschaft der Industrie, verschiedene Integrationsansätze für einen standardisierten Datenaustausch beschrieben. Die dabei etablierten Techniken und Protokolle des IoT und IIoT sowie neue M2M Kommunikationswege der Industrie 4.0 werden nach den Schichten des TCP/IP Referenzmodells untersucht, um eine strukturierte Vorgehensweise zu ermöglichen und ein ganzheitliches Bild der Netzwerkkommunikation zu erhalten. Dabei werden beispielhaft Mis-Use-Cases der etablierten Technologien beschrieben und deren Auswirkung auf die Kommunikation im Netzwerk dargestellt.

3.1 Bedrohungen

Die vierte industrielle Revolution, das IIoT und dessen Vielzahl an aktiven und passiven Elementen stellen in ihrer Komplexität eine große Herausforderung für die IT-Sicherheit dar. Einerseits muss die Sicherheit der laufenden Software, der Infrastruktur, Anwendungs- und Rechnersysteme gewährleistet werden, andererseits muss die Betriebssicherheit der Geräte und Anlagen, welche mit dem Internet verbunden sind sichergestellt werden. Das Management der IT-Sicherheit in Industrie 4.0 Netzen geht über Unternehmensgrenzen hinweg, da Netze und Systeme für Kunden, Lieferanden und Partner bereitgestellt werden (DTAG 2016). Somit hat sich auch die Bedrohungslage der Netze geändert. Das Bundesamt für Sicherheit

in der Informationstechnik (BSI) beschreibt die Top 10 Bedrohungen und deren Folgen für Industrial Control System (ICS) in Bundesamt für Sicherheit in der Informationstechnik 2016.

1. Social Engineering und Phishing
2. Einschleusen von Schadsoftware über Wechseldatenträger und externe Hardware
3. Infektion mit Schadsoftware über Internet und Intranet
4. Einbruch über Fernwartungszugänge
5. Menschliches Fehlverhalten und Sabotage
6. Internet-verbundene Steuerungskomponenten
7. Technisches Fehlverhalten und höhere Gewalt
8. Kompromittierung von Extranet und Cloud Komponenten
9. Denial of Service (DoS) und Distributed Denial of Service (DDoS)
10. Kompromittierung von Smartphones im Produktionsumfeld

Die Auswirkungen dieser Bedrohungen sollen im weiteren Verlauf der Analyse beispielhaft mit Bezug auf die in Unterabschnitt 2.3.5 genannten Schutzziele und der aktuellen Industriestandards dargestellt werden. Für eine sichere Kommunikation ist es notwendig einen größtmöglichen Schutz gegen diese Bedrohungen bereitzustellen. Die Sicherheit eines Gesamtsystems kann nicht nur an einer einzigen Stelle im Netzwerk hergestellt und gewährleistet werden. Es muss auf allen Ebenen des Netzwerkstacks für Sicherheit gesorgt werden (Plattform Industrie 4.0 2017). Dafür müssen die Netzwerkinfrastruktur und die eigentliche Kommunikation im Netzwerk gesichert werden. Dies geschieht durch die Abschottung von Systemen, die Einschränkung von Zugangsberechtigungen, die Härtung der Sicherheit der genutzten Komponenten sowie den Einsatz von geeigneten Netzwerkprotokollen und Verschlüsselungsverfahren.

3.2 Integrationsansätze

Die Grundlage der Industrie 4.0 Kommunikation ist ein standardisierter Datenaustausch über alle Schichten der Automatisierungspyramide (Abschnitt 2.2) hinweg.

Dafür müssen bestehende Systeme in die Industrie 4.0 Kommunikation integriert werden. Dies führt häufig zu Problemen, da diese Systeme proprietäre Protokolle nutzen, besondere Anforderungen besitzen oder gar keine digitalen Schnittstelle bereitstellen. Es bestehen für kommunikative Systeme grundsätzlich zwei Ansätze zur Integration in die Netze der Industrie 4.0.

3.2.1 Konsolidierung der Netzwerkkommunikation

Bestehende Systeme können, wenn möglich, erweitert oder ersetzt werden, um den in Unterabschnitt 2.3.5 beschriebenen Anforderungen gerecht zu werden. Dies ist mit einem hohen technischen und betriebswirtschaftlichen Aufwand verbunden. Die Konsolidierung der Netzwerkkommunikation muss als stetiger Prozess verstanden werden. Dabei stellt die digitale Kommunikation und Vernetzung der Systeme bei der Integration neuer Komponenten oder dem Austausch bestehender Systeme eine zentrale Rolle. Die Referenzarchitekturmodelle RAMI4.0 (Abschnitt 2.3.2) und IIRA (Abschnitt 2.3.2) stellen die Grundlage zur Konzeption neuer Industrie 4.0 Netze bereit.

3.2.2 Gatewaykommunikation

Eine Alternative zur Umstellung der bestehenden Systeme stellt die Kommunikation über Gateways dar. Hierbei gibt es mehrere Softwarelösungen, welche unterschiedliche Ziele verfolgen. Es werden Systeme zur Anlagenoptimierung (SePiA.Pro[1]), der Bereitstellung einer offenen, branchenübergreifenden Plattform mit diversen Smart Services wie Datenanalyse und Flottenmanagement (Siemens Mindsphere[2] und DeviceInsight[3]) und dem herstellerübergreifenden Gerätemanagement (AXOOM[4]) entwickelt (**acatec2016**). Neben der Sammlung, Verwaltung und Bereitstellung der Daten, bieten sie Schnittstellen für vorhandene Systeme, um diese in Industrie 4.0 Netze zu integrieren. Die Einsatzmöglichkeiten dieser Softwarelösungen sind von den vorhandenen Schnittstellen der Anlagen abhängig und benötigen eine individuelle Konfiguration um den unterschiedlichen Anforderungen der Industrielandschaft gerecht zu werden. Diese werden in Absprache mit dem

[1] TODO - Link
[2] TODO - Link
[3] TODO - Link
[4] TODO - Link

Hersteller erarbeitet oder, wenn möglich, über Plug-Ins bereitgestellt und besitzen somit eine gewisse Herstellerabhängigkeit.

Die beschriebenen Softwarelösungen und deren Implementierungen können Softwarefehler besitzen und Schwachstellen bereitstellen. Die weitere Betrachtung dieser Systeme ist im Rahmen der Thesis aufgrund ihrer Proprietarität nicht weiter möglich.

Das in Abschnitt 2.6 beschriebene Testsystem implementiert die Form der Gatewaykommunikation im Industrie 4.0 Netz mit Hilfe des Protokolls OPC UA. Das Anwendungsszenario des Buchdrucks ermöglicht die Kommunikation eines Industrie 4.0 Netzes auf Basis von OPC UA mit einer nicht Industrie 4.0 kompatiblen Druckerkomponente. Der OPC UA Server agiert als Gateway für seine Netzwerkkomponente. Bei der selbstständigen Implementierung dieser Funktionalitäten ist nach dem Prinzip *Security by Design* (Abschnitt 2.5) vorzugehen, um nicht das Netzwerk durch die Integration der neuen Komponente zu gefährden. Das genutzte Protokoll OPC UA basiert auf dem abstrakten OPC UA Connection Protocol (UACP). Dieses bietet mehrere konkrete Implementierungen der Nachrichtenübermittlung und dessen Sicherheitsprofile, um den unterschiedlichen Anforderungen der Industrie 4.0 Netze und deren Komponenten gerecht zu werden. Das Protokoll OPC UA wird in Unterabschnitt 3.6.3 untersucht.

3.3 Netzzugangsschicht

Die Netzzugangsschicht stellt die erste Instanz der Kommunikation im Netzwerk dar. Sie beinhaltet das Übertragungsmedium sowie die Topologie, in welcher die Kommunikation stattfindet. Das Übertragungsmedium bestimmt die Form der Signalübertragung. In Industrie 4.0 Netzen können neben der klassischen Kabelverbindung auch andere (instabile) Kanäle wie Mobilfunk oder Satelliten in Frage kommen. Um die Kommunikation über alle Medien sicher und zuverlässig zu gestalten, müssen auf technischer Ebene Protokolle genutzt werden, welche es ermöglichen die gegebenen Schutzziele zu realisieren und die Integrität der Daten bei der Übertragung über große Entfernungen zu gewährleisten. Dominante Technologien dieser Schicht sind Institute of Electrical and Electronics Engineers (IEEE) 802.3

(Ethernet)[5], IEEE 802.11 (Wireless LAN)[6] und IEEE 802.15.4[7] (Plattform Industrie 4.0 2017).

Im weiteren Verlauf dieser Arbeit wird sich, aufgrund der weiten Verbreitung in Industrienetzen, auf kabelgebundene, Ethernet-basierte Netze als Grundlage der Signalübertragung beschränkt. Eine Analyse weiterer Übertragungsmedien wie Funk, Licht oder Infrarot und deren Protokolle wird nicht durchgeführt.

3.3.1 physikalischer Zugang

Die Netzzugangsschicht beinhaltet als einzige Schicht des TCP/IP Referenzmodells nicht nur die verwendeten Protokolle zur Signalübertragung, sondern auch die physikalischen Gegebenheiten des Übertragungsmediums. Die Sicherheit dieser Netzwerkschicht beinhaltet somit nicht nur die verwendeten Techniken, sondern auch die physische Sicherheit der Systeme. Sie wird durch den Zugang zur Hardware dargestellt und besitzt eine große Bedeutung, um unbefugte Eingriffe in das Netzwerk zu verhindern. Die Sicherheit dieser Systeme wird durch die physikalische Abschottung mit Hilfe von abschließbaren Serverschränken, genereller Zugangskontrolle sowie der Abschaltung von Ports an Netzwerkkomponenten oder Endsystemen gewährleistet. (Plattform Industrie 4.0 2017)

Die Infektion von Systemen stellt nicht nur eine Bedrohung für die Kommunikation im Netzwerk dar, sondern für alle vernetzen Prozesse im Unternehmen. Die Schadsoftware kann direkt über die Netzwerkkomponenten oder auch durch Zugriff auf externe Schnittstellen der Clients wie Universal Serial Bus (USB) Ports oder andere Wechselmedien im Netzwerk verbreitet werden und Einfluss auf die im Netzwerk vorhandenen IIoT Systeme nehmen. Da der physikalische Zugang zu den Clients nicht durch Zugangs- oder Zutrittskontrolle verhindert werden kann, muss die Sicherheit vor diesen Eingriffen durch Authentifizierung und Autorisierung mit Hilfe von Access Control List (ACL)[8] oder Verzeichnisdiensten wie Samba[9] oder Active Directory[10] auf der Anwendungsschicht gewährleistet werden.

[5] Link zu IEEE 802.3
[6] Link zu IEEE 802.11
[7] Link zu IEEE 802.15.4
[8] TODO - ACLs
[9] TODO - Samba
[10] TODO - AD

3.3.2 VLAN

Eine weitere Sicherheitsmaßnahme zur Prävention von Manipulation des Netzwerks durch physikalischen Eingriff, stellt die logische Trennung der Netze durch die Verwendung von VLAN dar.VLANs arbeiten auf der Netzzugangsschicht des TCP/IP Referenzmodells. Die Netzwerksegmente werden mit einem *TAG* versehen. Das physikalische Netz wird in logische VLAN Teilnetze gegliedert. Die Technologie des VLAN unterteilt sich in die Ausprägungen statisches- und dynamisches-VLAN. Das statische VLAN wird am Switch konfiguriert und ordnet einen Port einem VLAN zu. Beim dynamischen VLAN wird die Zuordnung des VLANs anhand von Inhalten im eintreffenden Netzwerksegment getroffen. Das statische VLAN bietet im Gegensatz zum dynamischen VLAN eine höhere Sicherheit gegenüber Manipulation, da die Zuordnung des Netzwerks über einen statischen Port stattfindet und nicht über Software manipuliert werden kann. Jedoch erfordert es einen erhöhten Administrationsaufwand, bei Änderungen im Netzwerk entweder die Konfiguration des Switch angepasst oder die physikalische Verkabelung geändert werden muss.

Da zwischen den Schichten im TCP/IP Referenzmodell keine Kommunikation stattfindet, werden VLANs auch genutzt, um QoS für Dienste der höheren Schichten bereitzustellen. Im Netzwerk könnten Ressourcen für das VLAN reserviert werden und somit die Sicherheit der Kommunikation gewährleistet werden.

3.3.3 vertikale Integration bestehender Komponenten

Bisher werden in der Industrie, je nach Anwendungsfall, verschiedene Umsetzungen von Netzwerktopologien, wie Punkt-zu-Punkt-, Bus-, Stern- oder auch Hybridegenutzt. Jede dieser Netzstrukturen bietet Vor- und Nachteile bzgl. Durchsatz, Administrationsaufwand und Skalierbarkeit (Burke 2013). Um die Grundidee der Industrie 4.0, die unternehmensübergreifende, intelligente Vernetzung von Produktionsressourcen umzusetzen, ist jedoch eine einheitliche, vertikale Kommunikation über alle Ebenen der Automatisierungspyramide notwendig. Industrie 4.0 Netze kommunizieren über TCP/IP Verbindungen und basieren auf dem *Ethernet*[11] Protokoll. Die Integration bestehender Komponenten findet wie in Unterabschnitt 3.2.2 beschrieben über Gateways statt, welche in beide Netze integriert werden und so-

[11] TODO - Link zu IEEE Ethernet

mit die Kommunikation zwischen dem bestehenden Netzwerk und dem Industrie 4.0 *End2End* Netzwerk bereitstellen.

Die Systeme der Unternehmens- und Betriebsleitebene werden durch komplexe ERP und MES Systeme beschrieben und in Standardkomponenten und Software umgesetzt, welche nach dem TCP/IP Referenzmodell kommunizieren. Die unteren Ebenen der Automatisierungspyramide (Steuerungs- und Feldebene) werden durch spezielle Hard- und Softwarelösungen dargestellt. Die vorhandenen Ressourcen dieser Systeme sind begrenzt und deren Kommunikation ist u. a. für spezielle Anwendungsfälle wie harte Echtzeitkommunikation mit Verzögerungen <1ms ausgelegt. Die Integration einer Kommunikationsschnittstelle für die digitale Vernetzung ist in der Praxis nur mit erheblichem Aufwand oder gar nicht möglich. Somit muss die Integration der bestehenden Komponenten auf der Prozessleitebene der Automatisierungspyramide stattfinden. Die Systeme besitzen die benötigten Ressourcen und müssen als Schnittstelle zwischen der (meist proprietären) Kommunikation der Steuerungs- und Feldebene und den oberen Schichten der Automatisierungspyramide dienen. Die Prozessleitebene stellt das Bindeglied zwischen den Industrieanlagen und der einheitlichen Kommunikation in Industrie 4.0 Umgebungen dar.

3.4 Internetschicht

Auf der Internetschicht findet die Vermittlung der Datenpakete zwischen den Teilnehmern im Netzwerk statt. Auf dieser Schicht hat sich IP zum Standard für Netzwerkübergreifende Rechnerkommunikation durchgesetzt. Dies gilt auch für die immer komplexer werdenden Industrienetzwerke und die Industrie 4.0. (Christoph Meinel 2011)

Zu den Aufgaben der Internetschicht gehört das Bereitstellen von Adressen, das Routing, die Fragmentierung von Datenpaketen zur Übertragung im Netzwerk sowie die Sicherstellung der Dienstgüte. Um Routing und Adressvergabe in IP-Netzen zu realisieren, werden die Dienste DNS (Unterabschnitt 3.6.1) und DHCP (Unterabschnitt 3.6.2) genutzt. Das IP Adress Management (IPAM) und die Zuordnung der physikalischen Hardware zur logischen IP-Adresse erfolgt mit Hilfe des ARP. Da die Kommunikation in einem IP-Netz ohne diese Dienste und Protokolle nicht

möglich ist, stellen sie einen wichtigen Bestandteil im Netzwerk dar und müssen vor Sabotage geschützt werden.

3.4.1 ARP

ARP dient der Zuordnung einer physikalischen Hardwareadresse einer Netzwerkschnittstelle zu einer logischen IP Adresse. Diese Zuordnung wird mit Hilfe einer Tabelle, des ARP-Cache, ermöglicht. Jeder Client im Netzwerk verwaltet einen ARP-Cache. Abbildung 3.1 stellt das Format eines ARP Pakets dar. Während des ARP Request wird ein ARP Paket gesendet, welches die MAC- und IP-Adresse des Absenders sowie die IP-Adresse des Empfängers enthält. Der Request wird über die Broadcast MAC-Adresse des Netzes an alle Teilnehmer gesendet. Empfängt ein Teilnehmer das Paket mit seiner IP Adresse, sendet er einen ARP Reply mit seiner MAC-Adresse zum Absender. Dieser trägt die MAC-Adresse in seinem ARP Cache ein.

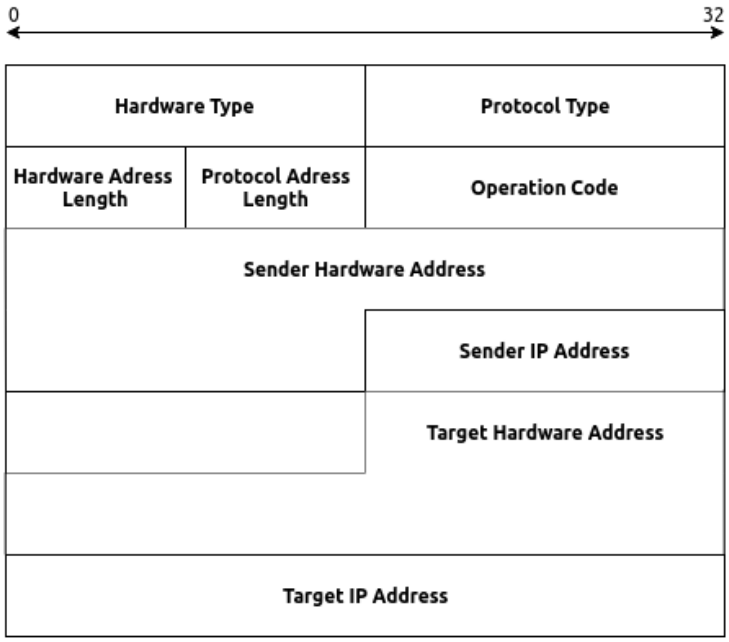


Abbildung 3.1: ARP Paketformat

Die simple Architektur des ARP ermöglicht es, die Kommunikation im Netzwerk zu manipulieren und einen MitM Angriff im Netzwerk durchzuführen. Da im Protokoll keine Mechanismen zur Überprüfung der empfangenen Daten wie Checksum o. Ä. vorhanden sind und die Pakete über die Broadcast Adresse im Netzwerk verteilt werden, können die Pakete von jedem Teilnehmer im Netzwerk eingesehen und darauf geantwortet werden. Die Angriffsform des *ARP Spoofing* nutzt diese Schwachstelle im Protokoll aus. Der Angreifer versendet gefälschte ARP Pakete im Netzwerk. Diese beinhalten die MAC-Adresse der Netzwerkschnittstelle des Angreifers als Zuordnung zu den IP Adressen im Netzwerk. Der angegriffene Host sendet die Netzwerkpakete zur im Paket angegebenen IP Adresse zukünftig über den Host des Angreifers. Dies lässt den Angreifer die Pakete zwar empfangen, stellt aber die Funktionalität des Netzwerks ein. Um dies zu verhindern, muss der Angreifer die Pakete nun mit Hilfe seiner ARP Tabelle zum eigentlichen Empfänger weiterleiten.

Moderne Instrusion Detection System (IDS) können *ARP Spoofing* anhand von Mustererkennung identifizieren und Maßnahmen zur Sperre dieser Netzwerkpakete einleiten. Aufgrund der geringen Gültigkeitsdauer des ARP Cache ist eine effiziente Erkennung und Vermeidung eines Angriffs in der Praxis nur schwer umzusetzen. Eine weitere Schutzmaßname gegen diese Form des Eingriffs in das Netzwerk kann durch das Arbeiten mit statischen Tabellen erreicht werden. Dies kann jedoch aufgrund des hohen Administrationsaufwands nur in kleinen Netzwerkinfrastrukturen zum Einsatz kommen.

3.4.2 QoS

Eine Industrie 4.0 Netzwerkinfrastruktur kann aufgrund der unterschiedlichen Anforderungen an die Systeme auf verschiedenste Weisen ausgeprägt sein. Die Heterogenität der Komponenten im Netzwerk und deren Anforderungen an die Kommunikation auf der vertikalen Ebene der Automatisierungspyramide Abschnitt 2.2 stellen eine Herausforderung für die Sicherheit der Datenübertragung dar und können die Umsetzung eines Netzwerks beeinflussen. Industrie 4.0 Umgebungen können sich über weite Distanzen (Metropolitan Area Network (MAN), WAN und Global Area Network (GAN)) erstrecken und sind somit auch von physikalischen Gegebenheiten wie Latenz und Jitter betroffen. Diese Erscheinungen müssen berücksichtigt

werden, um eine fehler- und verlustfreie, sichere Kommunikation zu gewährleisten (Torscht 2014).

Für die Beurteilung und Bereitstellung der Dienstgüte in IP-Netzen müssen die Übertragungsgüte der Netzzugangsschicht sowie die übertragungstechnischen Parameter der Internetschicht (IP-Ebene) betrachtet werden. In IP-Netzen wird der Einfluss auf die QoS in den folgenden Parametern beschrieben:

- Latenzzeit: Dauer der Paketübertragung
- Jitter: Abweichung der Latenzzeit von ihrem Mittelwert
- Paketverlustrate: Wahrscheinlichkeit des Verlusts von IP-Paketen während der Übertragung
- Durchsatz: gemittelte Datenmenge pro Zeiteinheit

All diese Faktoren haben in einem paketorientierten Netzwerk, in welchem die Datenpakete nach dem *Best-Effort-Prinzip* versendet werden, auf die fehlerfreie Kommunikation aufgrund der durch *Ethernet* und IP bereitgestellten Fehler- und Flusskontrolle wenig Einfluss. Sie spielen jedoch bei zeitkritischen Anwendungen der Industrie 4.0 eine wichtige Rolle. Auf den niedrigeren Schichten des TCP/IP Referenzmodells ist es nicht möglich zwischen verschiedenen Datenpaketen der höheren Schichten zu unterscheiden. Um dieses Problem zu lösen werden auf Dienste mit besonderer Güte in VLANs aufgenommen und somit deren Pakete bereits auf der Netzzugangsschicht kenntlich gemacht (Unterabschnitt 3.3.2), um die Dienstqualität sicherzustellen. Des Weiteren müssen, um QoS in einem Netzwerk anzuwenden, diese Mechanismen auf der gesamten Übertragungsstrecke implementiert werden. Der Transport von Daten unterschiedlicher Priorität in Netzwerken wird in IEEE 802.1p und IEEE 802.1Q[12] beschrieben.

3.4.3 IPsec

Die Internet Engineering Task Force (IETF) beschreibt im Request for Comments (RFC) 4301[13] die Architektur von Internet Protocol Security (IPsec). IPsec ermöglicht es die Schutzziele Vertraulichkeit, Authentizität und Integrität bereits auf der Internetschicht des TCP/IP Referenzmodells zu umzusetzen. Um alle Schutzzie-

[12] IEEE Std 802.1Q - IEEE Standard for Local and metropolitan area networks–Bridges and Bridged Networks
[13] IETF RFC 4301 - Security Architecture for the Internet Protocol

le umzusetzen, wird das Protokoll IP um die Bestandteile Authentication Header (AH), Encapsulating Security Payload (ESP) und IKE[14] erweitert.

IPsec wurde in der Industrie zur Bereitstellung von dauerhaften Site-to-Site Virtual Private Network (VPN) Verbindungen genutzt. Aufgrund der immer weiteren Öffnung der Unternehmen und der direkten Kommunikation der Komponenten miteinander finden dauerhafte diese Verbindungsformen in Industrie 4.0 Umgebungen jedoch immer seltener Anwendung. Seit der Verbreitung der Verschlüsselung der Anwendungsdaten über Transport Layer Security (TLS) werden für die Bereitstellung von getunneltem Netzwerkverkehr aufgrund der besseren Handhabbarkeit und der einfacheren Konfiguration für Administrator und Anwender bevorzugt Secure Sockets Layer (SSL)-VPN Lösungen verwendet.

Aufgrund der geringen Relevanz der Technik in Industrie 4.0 Netzwerken wird die Kommunikation über das Protokoll IPsec im weiteren Verlauf der Thesis nicht weiter untersucht.

3.5 Transportschicht

Während auf der Netzwerkschicht allein das Protokoll IP die Basis für die Vernetzung und Adressierung von Industrie 4.0 Systemen darstellt, wird das Protokoll der Transportschicht durch die Anforderungen an das Netzwerk bestimmt. Die Kommunikation in der Industrie 4.0 erfolgt über ein IP Netzwerk, welches zum Datentransport das Protokoll TCP für *End2End* (Abschnitt 2.3.3) Kommunikation nutzt (Plattform Industrie 4.0 2017). Wie in Unterabschnitt 2.3.3 beschrieben, bieten sich in der Praxis bei besonderen Anforderungen wie der Verteilung von Informationen im Netzwerk oder zeitkritischen Automatisierungsanwendungen jedoch auch andere Strukturen für die Kommunikation wie *Publish-Subscribe* (Abschnitt 2.3.3) in Verbindung mit dem Datagramm UDP an.

Das Protokoll TCP sowie das Datagramm UDP sind für die Übertragung der Segmente im Netzwerk sowie das Multi-/Demultiplexing verantwortlich. Dabei spielt der Inhalt des zu übertragenden *payload* keine Rolle. Die Analyse der Sicherheit im Netzwerk auf dieser Schicht des TCP/IP Referenzmodells beschränkt sich ausschließlich auf die Form der Datenübertragung sowie der dafür genutzten Netzlast.

[14] IKE - Protokoll zur Verwaltung der Security Association

3.5.1 TCP

Das Protokoll TCP[15] verfolgt das Prinzip eines *guaranteed delivery* und stellt eine zuverlässige Datenübertragung zwischen zwei *Hosts* (*Unicast*) bereit. Hierzu werden verschiedene Mechanismen zur Segmentierung der Daten, dem Verbindungsmanagement sowie der Fehler- und Flusskontrolle bereitgestellt. Diese Mechanismen, welche durch den Aufbau des des TCP Headers und die Nutzung von Timeouts und Algorithmen realisiert werden, sind für den Erfolg des Protokolls für zuverlässige, paketorientierte *End2End* Kommunikation verantwortlich.

Ein wichtiger Bestandteil des TCP Verbindungsmanagements und der Fehlerkontrolle stellt der 3-Wege-Handshake beim Verbindungsaufbau sowie -abbau dar. Er wird mittels der *Sequence-* und *Acknowledgementnumber* sowie den zugehörigen *Flags* des TCP Headers (synchronise (SYN), synchronise-acknowledge (SYN-ACK), acknowledge (ACK), final (FIN)) realisiert. Während des Verbindungsaufbaus werden die Adresse des Clients sowie der Status der Verbindung im Speicher gehalten. Die folgende Abbildung zeigt schematisch den Ablauf eines TCP Verbindungsaufbaus zwischen Client und Server. Der Client sendet zuerst eine Paket mit SYN-Flag zum Server. Dieser bestätigt das eingetroffene Paket des Clients mit dem SYN-ACK Flag, inkrementiert die Sequenznummer x in seinem *acknowledgment number* Segment[16] und erzeugt eine neue Sequenznummer für das Antwortpaket. Der Verbindungsaufbau wird durch die Bestätigung des SYN-ACK Pakets durch den Client und den Empfang des ACK Pakets vom Server abgeschlossen. Die initiale Sequenznummer x des SYN Pakets sowie die initiale Sequenznummer y des SYN-ACK Pakets können von den Beteiligten frei bestimmt werden. Der 3-Wege-Handshake wird in Abbildung 3.2 an einem Verbindungsaufbau von Client zu Server dargestellt.

[15] Network Working Group 1981
[16] text

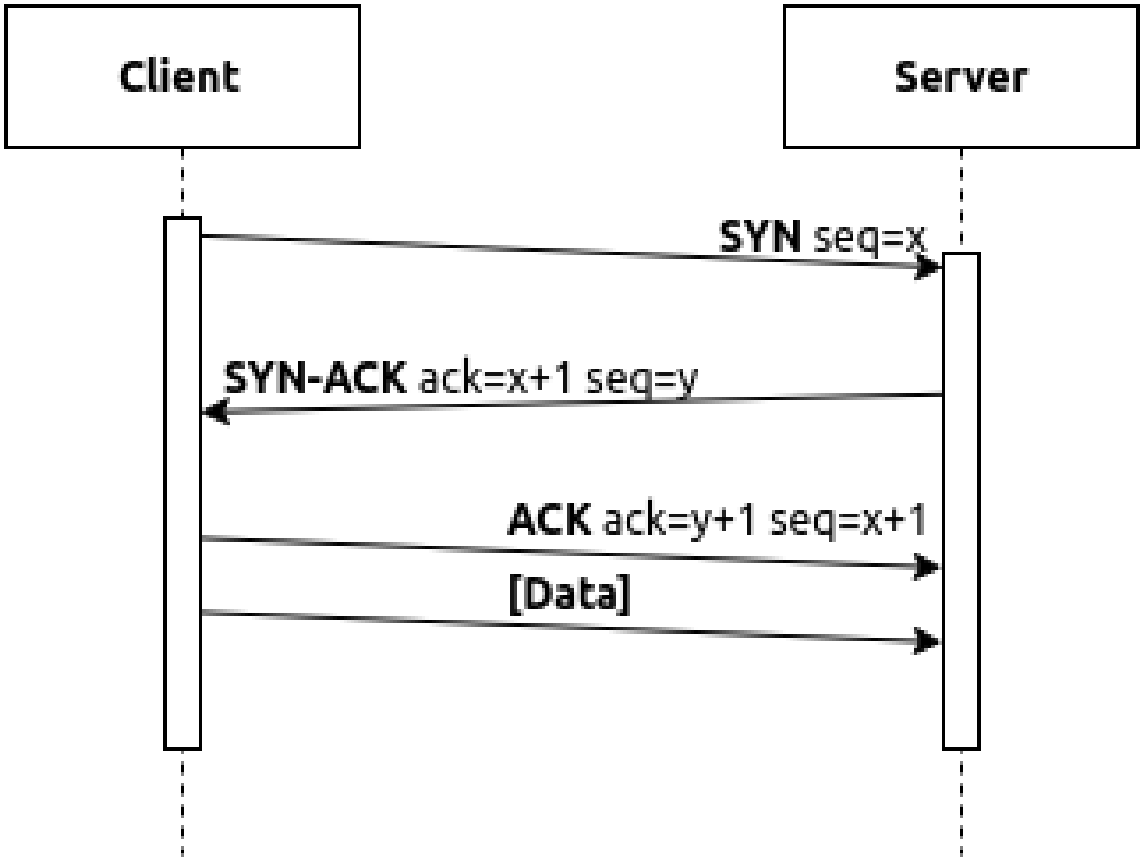


Abbildung 3.2: TCP Verbindungsaufbau

SYN-Flood

Der Mechanismus des 3-Wege-Handshakes kann durch einen SYN-Flood Angriff ausgenutzt werden und somit die Netz- und Systemlast manipuliert werden. Der SYN-Flood stellt eine Form des DoS bzw. DDoS Angriffs dar. Dabei werden zu einem gezielten TCP Dienst große Menge an Paketen mit gesetztem SYN *Flag* gesendet, um einen Verbindungsaufbau zum Server vorzutäuschen. Nach Erhalt des Pakets sendet der Server dem Client ein SYN-ACK Paket um den Verbindungsaufbau zu initiieren und wartet auf Bestätigung. Diese Bestätigung wird vom Angreifer unterschlagen. Somit bleiben auf dem angegriffenen System Ressourcen dieser halb offenen Verbindung bis zum erreichen eines Timouts belegt. Ein verteilter Angriff auf ein System kann dessen Ressourcen schnell komplett beanspruchen und somit zur Ablehnung jeglicher weiterer Verbindungen führen.

Die Kritikalität dieses Angriffs liegt in der Unausgewogenheit der benötigten Ressourcen zwischen Angreifer und Opfer. Es benötigt nur wenig Rechenaufwand und Bandbreite um ein 20 Byte großen TCP-SYN Header mit entsprechender *payload* zu erzeugen und zu versenden, jedoch viele Ressourcen um sich durch eine Echtzeitanalyse der Pakete durch eine Firewall oder SYN-Cookies vor diesen Angriffen zu schützen. Diese werden aufgrund ihrer Ressourcenbelastung aus wirtschaftlichen Gründen meist nur minimal oder gar nicht umgesetzt.

Der Mechanismus der SYN-Cookies wird auf dem Server implementiert. Hierbei werden die Informationen Zeitstempel, IP Adresse und Port von Client und Server in die initiale Sequenznummer des vom Server gesendeten SYN-ACK Pakets kodiert. Diese müssten normalerweise in einer Tabelle im Speicher gehalten werden. Ein überlaufen der Tabelle ist somit unmöglich, da sie nicht vorhanden ist. Jedoch benötigt jede Kodierung und Dekodierung Systemressourcen. Ein ausreichend großer Angriff auf das System kann somit trotzdem die gesamten Systemressourcen beanspruchen und das Ziel eines DDoS Angriffs, der Negierung eines Dienstes, erfüllen.

Dedizierte Firewalls können mit Hilfe von IDS die Pakete beim Eintreffen im Netzwerk analysieren, Angriffe erkennen und Verbindungen dieser Quelladressen blockieren.

Sockstress

Eine weitere Angriffsform, welche den 3-Wege-Handshake des TCP Protokolls als Mis-Use-Case nutzt, wurde im Jahr 2008 von den Sicherheitsforschern Jack C. Louis und Robert E. Lee von Outpost24[17] entdeckt und mit dem Namen *sockstress* bezeichnet. Der Angriff stellt eine einfache Form eines DoS Angriffs dar. Das Ziel dieses Angriffs ist, ähnlich wie beim SYN-Flood, eine Negierung eines Dienstes oder des gesamten Systems mit Hilfe asymmetrischer Ressourcenauslastung bei Angreifer und Opfer zu erzielen.

Im Gegensatz zum SYN-Flood stellt *sockstress* eine vollständige Verbindung zum Server über den 3-Wege-Handshake her. In der einfachsten Form des Angriffs wird das *Receive Window* Flag des TCP Headers im ersten TCP Segment, welches vom Client zum Server nach dem Verbindungsaufbau übertragen wird, auf 0 gesetzt. Dies bedeutet, dass der Client dem Server mitteilt, dass er im Moment keine weiteren Daten empfangen kann. Der Server wird, durch den abgeschlossen Verbindungsaufbau, gezwungen die Verbindung im Speicher zu halten und den Client periodisch zu prüfen, ob dieser Daten empfangen kann. Dies belegt Systemressourcen und kann genutzt werden, um einen Dienst oder ein System zum Ablehnen aller Verbindungen oder zum Absturz zu bewegen. Der Ablauf des Angriffs wird in Abbildung 3.3 in Anlehnung an ein Unified Modeling Language (UML)-Sequenzdiagramm schematisch dargestellt. Die Nutzung von SYN-Cookies hat bietet keinen Schutz gegen diese Form des Angriffs, da die Verbindung vom Client zum Server vollständig aufgebaut wird. Industrieanlagen müssen mit Hilfe externer DDoS Serviceanbieter wie Akamai[18] oder Cloudflare[19], Firewalls und IDS Systeme oder spezieller Appliances, welche den Netzwerkverkehr Netzwerk-, Transport- und Anwendungsschicht überwachen, geschützt werden.

[17] http://www.outpost24.com
[18] Link - https://www.akamai.com
[19] Link - https://www.cloudflare.com

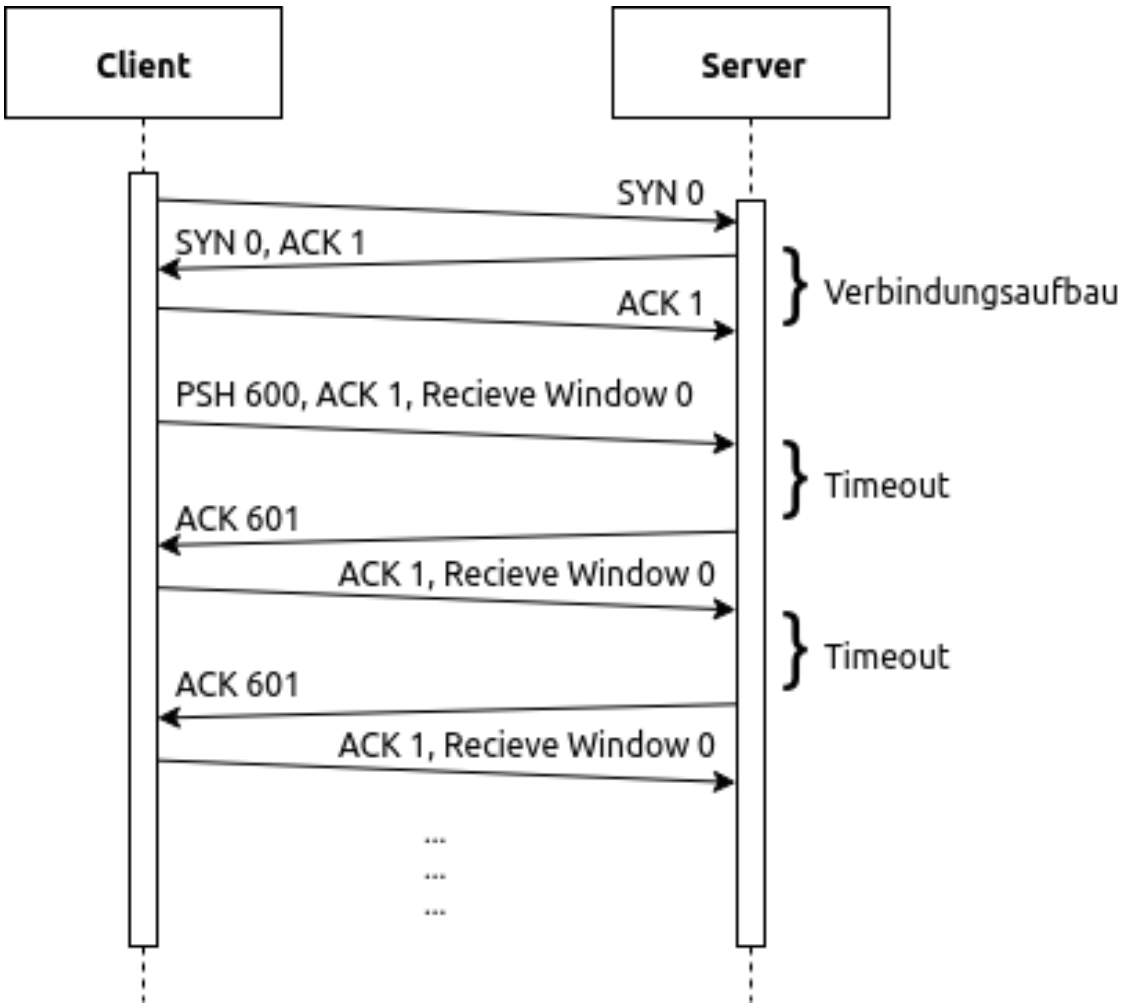


Abbildung 3.3: Sockstress Sequenzdiagramm

3.5.2 UDP

Das von der IETF im RFC 768[20] definierte Datagramm UDP ist ein verbindungsloses[21], nicht-zuverlässiges[22] Übertragungsprotokoll. Der UDP Header ist im Gegensatz zum TCP Header (20 Bytes) nur 8 Byte lang und bietet somit einen sehr geringen *Overhead*[23] beim Versenden von IP Paketen. UDP wurde als Alternative zu TCP entwickelt, um die Kommunikation mit niedrigeren Latenzen für Dienste wie Simple Network Management Protocol (SNMP) oder DNSoder Voice over IP (VoIP) zu ermöglichen (Olsen 2003). Es wird auf den für die Latenz kritischen 3-Wege-Handshake verzichtet und das *Fire-and-Forget*[24] Prinzip angewandt, wobei keine Verbindung zwischen zwei Kommunikationspartnern hergestellt wird, sondern die Pakete ohne Flusskontrolle von Sender zu Empfänger gesendet werden.

UDP findet in vielen Industrienetzen Einsatz als Transportprotokoll. Es bietet sich, vor allem durch seine Simplizität und den geringen Overhead im Netzwerk für die Informationsverteilung mit niedrigen Latenzzeiten an. Durch die Broad- und Multicast Funktionalitäten des UDP ist es möglich über das in Abschnitt 2.3.3 beschriebene *Publish Subscribe* Muster zu kommunizieren und somit die Netzlast bei einer großen Anzahl von Empfängern gering zu halten.

UDP führt keine Validierung der Absenderadresse im Paketheader durch (Olsen 2003). Dies ermöglicht die Anwendung von IP Spoofing. IP Spoofing kann genutzt werden, um DoS bzw. DDoS Angriffe auf ein System durchzuführen. Eine Untersuchung des Netzwerkdienstes DNS, welcher auf der Nutzung des Transportprotokolls UDP basiert, wird in Unterabschnitt 3.6.1 beschrieben und durchgeführt.

3.6 Anwendungsschicht

Die Netzwerkkommunikation der Anwendungsschicht in Industrie 4.0 Umgebungen basiert auf dem IP der Internetschicht. Um die Integration und Verwaltung der Netzwerkteilnehmer zu erleichtern, werden IP basierende Dienste wie DNS für die Namensauflösung sowie DHCP für die Adressvergabe und das Routing genutzt. Die Anwendungsschicht des TCP/IP Referenzmodells wird durch eine Viel-

[20] Link - https://tools.ietf.org/html/rfc768
[21] verbindungslos - TODO
[22] nicht zuverlässig - TODO
[23] Overhead - TODO
[24] Fire-and-Forget - TODO

zahl von Protokollen beschrieben. Bestehende Lösungen des IoT nutzen Protokolle wie HTTP, Extensible Messaging and Presence Protocol (XMPP) oder Simple Mail Transfer Protocol (SMTP) zur Kommunikation über das Netzwerk. In der M2M Kommunikation des IIoT haben sich die Protokolle und Standards OPC UA, DDS, MQTT und CoAP für unterschiedliche Anforderungen an die Netze und deren Teilnehmer hervorgetan.

3.6.1 DNS

DNS wird von der IETF in den RFC 1034[25], 1035[26], 2181[27] und 2782[28] beschrieben und verwaltet. Es stellt einen hierarchischen Verzeichnisdienst für IP-Netze zur Verfügung.

Eine der Hauptaufgaben des DNS ist der *forward lookup*. Hierbei werden Domain- bzw. Hostnamen in IP-Adressen übersetzt. Das Zusammenspiel eines hierarchischen Verzeichnisdienstes und der Namensauflösung bietet Angriffsfläche zum Eingriff auf die Kommunikation im Netzwerk. Im folgenden werden bekannte Angriffsformen auf den DNS Dienst und deren Auswirkungen auf das Netzwerk beschrieben.

DNS Spoofing

Die Angriffsmethode des DNS Spoofing verfolgt, ähnlich wie *Cache Poisoning*[29], das Ziel gefälschte Resource Record (RR) in den DNS Cache des Opfers einzuschleusen. Während das *Cache Poisoning* aus einer Softwareschwachstelle hervorging, bei der zusätzliche, gefälschte DNS Einträge zu korrekten DNS Antworten hinzugefügt wurden und somit der Cache eines Nameservers kompromittiert wurde, befindet sich der Angriffsvektor beim DNS Spoofing in der Fälschung von DNS Antworten. Die Header der Netzwerkpakete werden mit Hilfe von *IP Spoofing*[30] so manipuliert, dass sie vorgeblich vom *authorativen* Nameserver stammen.

[25] Domain Names – Concepts and Facilities
[26] Domain Names – Implementation and Specification
[27] Clarifications to the DNS Specification
[28] A DNS RR for specifying the location of services (DNS SRV)
[29] TODO - Cache Poisoning
[30] TODO - IP Spoofing bezeichnet das Versenden von IP Paketen mit gefälschter Absender IP

Um DNS Spoofing erfolgreich durchzuführen muss die gefälschte DNS Response des Angreifers vor der Antwort des zuständigen Nameservers beim angegriffenen DNS Resolver eintreffen. Sobald der physikalische Zugang zum Netzwerk gewährleistet ist, können die Latenzzeiten der gefälschten Pakete im Netzwerk sehr gering gehalten werden. Ist dies nicht möglich, kann mit Hilfe eines DoS bzw. DDoS Angriffs auf den zuständigen Nameserver, dessen Antwortzeit beeinflusst werden. Des weiteren muss die ID im DNS Header mit der des Request übereinstimmen. Dies wird in Abbildung 3.4 dargestellt und kann mit Hilfe des Netzwerkanalysetools Wireshark[31] in einer beliebigen Netzwerkumgebung mit zuständigem DNS Server nachgewiesen werden. Auf der linken Seite ist ein DNS Request eines Hosts im Netzwerk und dessen DNS Header mit ID zu erkennen. Auf der rechten Seite ist die Antwort des im Netzwerk vorhandenen DNS Nameservers zu sehen. Request und Response müssen die gleiche ID besitzen, um als gültig betrachtet zu werden.

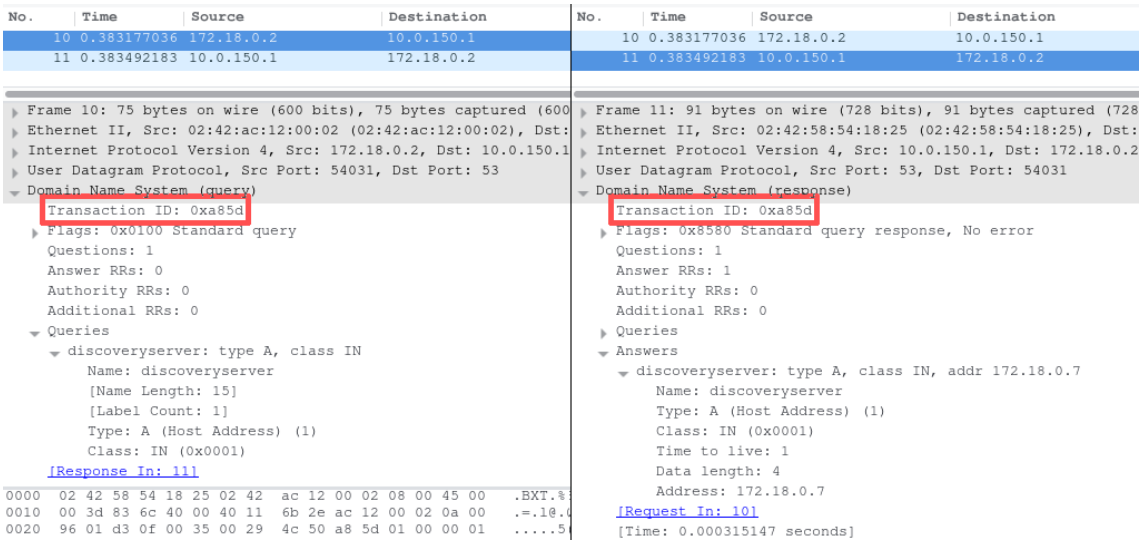


Abbildung 3.4: Wireshark - ID im DNS Header

[31] TODO Link zu Wireshark

DNS Amplification

Eine Form eines DDoS Angriffs (**??**) ist über DNS möglich und wird DNS Amplification genannt. Bei der DNS Amplification werden DNS Anfragen an offene Nameserver gesendet und mit Hilfe von IP Spoofing als Quell-IP die Adresse des Angreifers genutzt. Somit treffen die DNS Antworten beim anzugreifenden System ein und belasten dieses durch erhöhten Rechenaufwand sowie dessen Netzwerk durch Traffic. Ein weiterer Seiteneffekt dieses Angriffs ist eine hohe Last der Nameserver, welches durch das rekursive Verhalten der DNS Namensauflösung hervorgerufen wird. DNS Amplification beschreibt eine Form des Destributed-Reflected-Denial-of-Service (DRDoS).

Mit der Erweiterung des DNS in der IETF RFC 2617[32] wurde es notwendig, die Größe der DNS Antworten von 512 Byte auf einen dynamischen Puffer bis über 4000 Bytes zu erhöhen, um zusätzliche Informationen und Flags wie **??** über das DNS übertragen zu können. Dies wird sich vom Angreifer zunutze gemacht, da an den Nameserver Requests mit einer Paketgröße von 60 Bytes gesendet werden können, welche eine Antwort mit 4000 Bytes und mehr provozieren und somit einen Base Amplification Factor (BAF)[33] von ca. 66 im Netz haben (Ledermüller 2009). Dieser wird bei DNS Amplification durch die Paketgröße der Anfrage sowie der Antwort dargestellt.

Abbildung 3.5 stellt die Funktionsweise eines mit Hilfe von DNS Amplification durchgeführten DDoS Angriffs dar. Der Angreifer (links) sendet zum offenen Nameservern gefälschte DNS Anfragen mit Quelladresse des Opfers (rechts). Der Nameserver erfragen beim *authorativen* Nameserver die Zone, dieser stellt die erfragten RR bereit, anschließend sendet der Nameserver dem Opfer die Antworten zu.

[32] Link - https://www.ietf.org/rfc/rfc2671.txt
[33] BAF - Verhältnis von Eingangs- zum Ausgangssignal

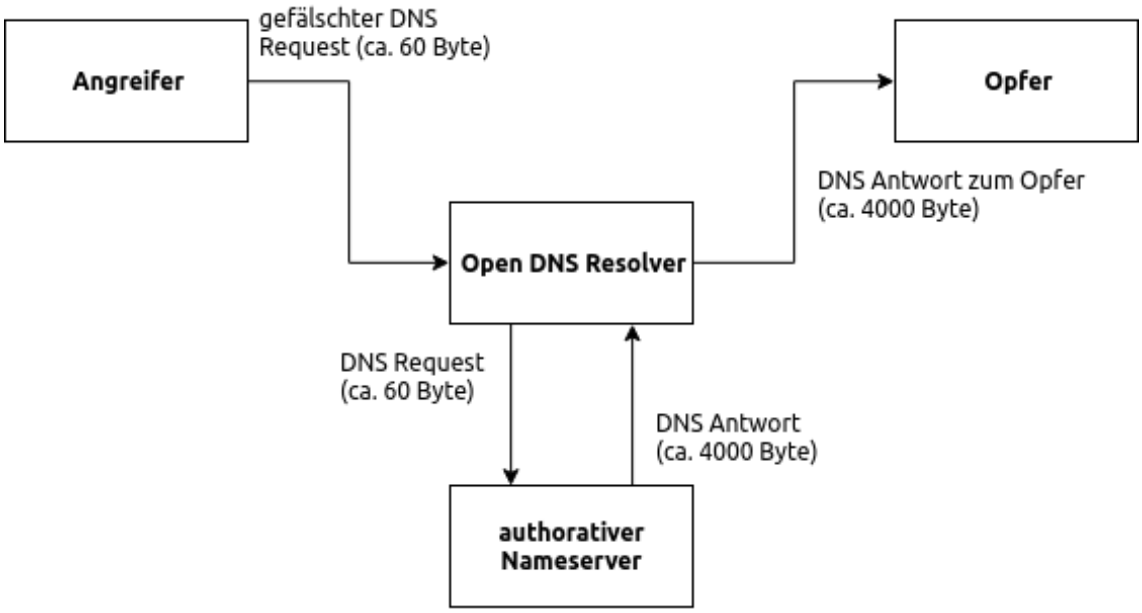


Abbildung 3.5: Schematisches Beispiel: DNS Amplification

Diese Form des Angriffs kann aus dem internen Netz sowie von extern auf öffentlich zugängliche Systeme durchgeführt werden. DoS Attacken stellen besonders für Industrie 4.0 Netzwerke, deren komplexe Kommunikation und Anforderungen eine hohe Bedrohung dar. Durch den erheblichen BAF können diese Angriffe mit wenig Bandbreite beim Angreifer durchgeführt werden und gleichzeitig das Netzwerk des Opfers voll auslasten. Wie in Abbildung 3.6 dargestellt, kann durch die Abfrage der Zone *isc.org* eine 3385 Byte große Antwort vom Nameserver provoziert werden. Dies führt zu einem BAF von ca. 56. Abbildung 3.7 beschreibt die Netzlast während eines DNS Amplification Angriffs im Quell- und Zielnetz. Auf der X-Achse wird die Anzahl der gesendeten Pakete pro Sekunde dargestellt, die Y-Achse zeigt die Netzlast in Gigabit pro Sekunde. Es ist eine lineare Steigerung der Netzlast in beiden Netzen zu erkennen, entscheidend ist jedoch der BAF. Beim Versandt von 10000 Paketen pro Sekunde müssen im Quellnetz nur ca. 4,8 Megabit/s an Daten transferiert werden, im Zielnetzwerk wird mit 287 Megabit/s eine wesentlich höhere Last erzeugt. Es ist möglich, mit einer vergleichsweise geringen Bandbreite im Quellnetz ein Netzwerk mit hoher Bandbreite im Zielnetz auszulasten.

```
; <<>> DiG 9.13.0 <<>> -t any isc.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 45468
;; flags: qr rd ra; QUERY: 1, ANSWER: 32, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1280
;; QUESTION SECTION:
;isc.org.                       IN      ANY

;; ANSWER SECTION:
isc.org.                7200    IN      SPF     "v=spf1 a mx ip4:204.1
isc.org.                7200    IN      CAA     0 issue "letsencrypt.o
isc.org.                60      IN      A       149.20.64.69
isc.org.                7200    IN      MX      20 mx.ams1.isc.org.
isc.org.                7200    IN      TXT     "google-site-verificat
isc.org.                7200    IN      CAA     0 iodef "mailto:hostma
isc.org.                7200    IN      NS      ams.sns-pb.isc.org.
isc.org.                7200    IN      NS      ord.sns-pb.isc.org.
isc.org.                7200    IN      CAA     0 issue "comodoca.com"
isc.org.                7200    IN      MX      10 mx.pao1.isc.org.
isc.org.                7200    IN      SOA     ns-int.isc.org. hostma
isc.org.                7200    IN      DNSKEY  257 3 5 BEAAAAOhHQDBrh
isc.org.                7200    IN      CAA     0 issue "Digicert.com"
isc.org.                7200    IN      DNSKEY  256 3 5 AwEAAcdkaRUlsR
isc.org.                7200    IN      NS      ns.isc.afilias-nst.inf
isc.org.                7200    IN      TXT     "v=spf1 a mx ip4:204.1
isc.org.                3600    IN      NSEC    _adsp._domainkey.isc.o
isc.org.                7200    IN      NAPTR   20 0 "S" "SIP+D2U" ""
isc.org.                7200    IN      NS      sfba.sns-pb.isc.org.
isc.org.                60      IN      AAAA    2001:4f8:0:2::69
isc.org.                7200    IN      RRSIG   SOA 5 2 7200 201807112
isc.org.                7200    IN      RRSIG   NS 5 2 7200 2018071123
isc.org.                60      IN      RRSIG   A 5 2 60 2018071123340
isc.org.                7200    IN      RRSIG   MX 5 2 7200 2018071123
isc.org.                7200    IN      RRSIG   TXT 5 2 7200 201807112
isc.org.                60      IN      RRSIG   AAAA 5 2 60 2018071123
isc.org.                7200    IN      RRSIG   NAPTR 5 2 7200 2018071
isc.org.                3600    IN      RRSIG   NSEC 5 2 3600 20180711
isc.org.                7200    IN      RRSIG   DNSKEY 5 2 7200 201807
isc.org.                7200    IN      RRSIG   DNSKEY 5 2 7200 201807
isc.org.                7200    IN      RRSIG   SPF 5 2 7200 201807112
isc.org.                7200    IN      RRSIG   CAA 5 2 7200 201807112

;; Query time: 596 msec
;; SERVER: 10.0.150.1#53(10.0.150.1)
;; WHEN: Mo Jun 18 18:04:45 CEST 2018
;; MSG SIZE  rcvd: 3385
```

Abbildung 3.6: DNS Amplification am Beispiel von isc.org

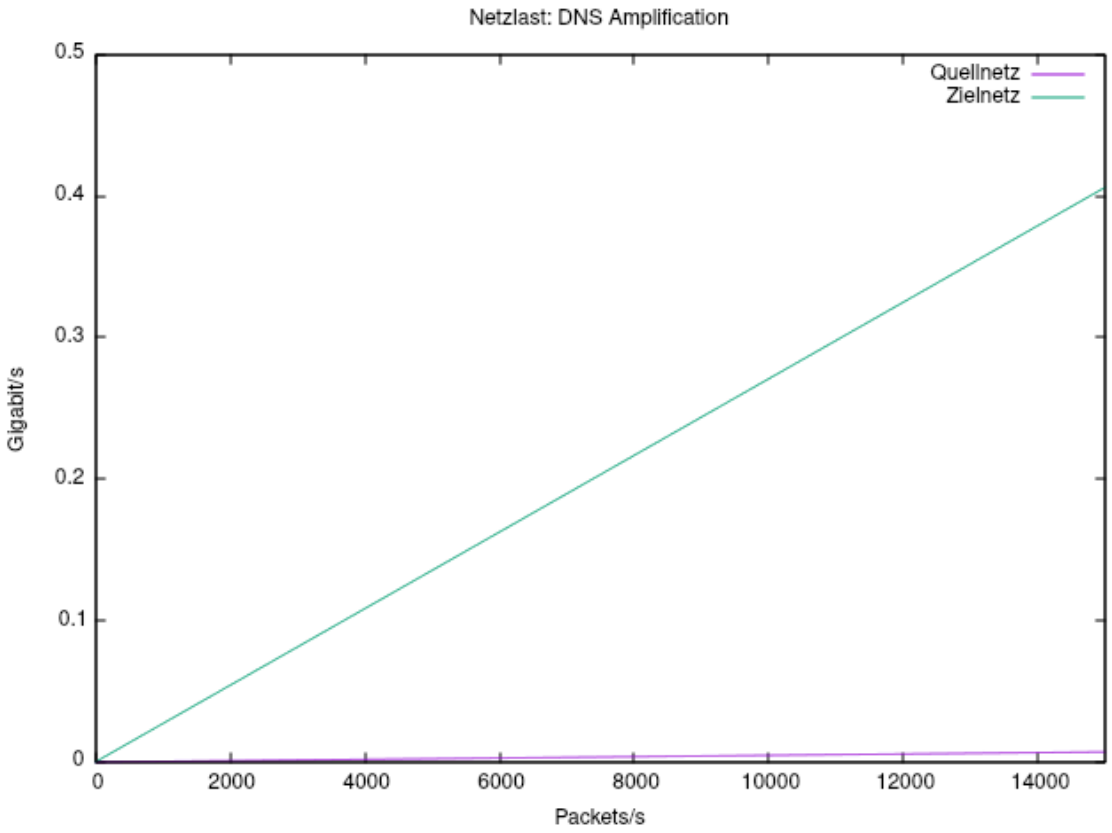


Abbildung 3.7: Netzlast bei DNS Amplification

Der DNS bietet weitere Angriffsmöglichkeiten wie DNS Fast Fluxing oder DNS Information Leakage. Diese Angriffe dienen zum *Phishing* von Daten oder der Spionage der Netzwerkstruktur. Sie nehmen initial keinen Einfluss auf den Netzwerkverkehr und dienen der Vorbereitung von Folgeangriffen und dem Sammeln von Informationen. Die Analyse der Sicherheit der Netzwerkkommunikation beschränkt sich auf Angriffe, welche direkten Einfluss in die Kommunikation im Netzwerk haben.

Das DNS kann um zusätzliche Sicherheitsmechanismen wie Transaction Signature (TSIG) und Domain Name System Security Extensions (DNSSEC) erweitert werden, um einen Schutz vor Eingriffen in die Namenauflösung zu gewährleisten. Diese bieten die Möglichkeit die Kommunikation zwischen Nameservern und Resolvern zu sichern, die Authentizität sowie die Validität der Zonen sicherzustellen. Bei TSIG werden symmetrische Schlüssel während der Übertragung der Domainzonen genutzt, DNSSEC benötigt die Verwendung von Extended DNS[34] und erweitert die Zonen der Domains um zusätzliche RR. Diese beinhalten den öffentlichen Schlüssel eines asymmetrischen Schlüsselpaars. Der private Schlüssel liegt beim *authorativen* Nameserver der Zone. Durch die Signierung der Zonen ist deren Authentizität geschützt. Alle im Internet vorhandenen Root-Nameserver[35] nutzen die DNS Erweiterung DNSSEC. In internen Netzen werden diese Sicherheitsmaßnahmen aufgrund von zusätzlichem Aufwand meist nicht umgesetzt (Ledermüller 2009).

3.6.2 DHCP

DHCP ist von der IETF im RFC 2131[36] definiert. Es stellt ein Framework zur Bereitstellung von Host-Konfigurationsparamertern in einem TCP/IP Netzwerk dar. Dazu gehören die IP Adresse, Netzmaske, Gateway sowie zuständiger DNS des Clients. Da ein neuer Client im Netzwerk keine Informationen über die vorhandenen Clients und dessen Topologie besitzt, muss er, um die Konfigurationsparameter für das Netzwerk zu erhalten (DHCP Discover), über einen Broadcast im Netzwerk nach Adressangeboten fragen. Der DHCP Discover findet über das Transportprotokoll UDP statt.

[34] TODO - Extended DNS
[35] TODO - Root-Nameserver
[36] Link - https://www.ietf.org/rfc/rfc2131.txt

Der Mangel an Informationen bei der initialen Verbindung eines Clients im Netzwerk, kann von einem Angreifer genutzt werden, um die Netzwerkkonfiguration zu manipulieren. Da der Broadcast des Clients an das gesamte Netzwerk versandt wird, ist es dem Angreifer möglich selbst auf diese Anfrage zu antworten. Hierzu wird die Technik des Spoofing in Verbindung mit ARP Poisoning oder einem zusätzlichen DHCP Server (Rogue DHCP), welcher vom Angreifer kontrolliert wird, im Netzwerk genutzt. In diesem Fall muss es dem Angreifer nur gelingen schneller auf den DHCP Discover bzw. DHCP Request des Clients zu antworten als der zuständige DHCP Server. Somit kann die Netzwerkkonfiguration des Clients manipuliert werden.

Ein Angriff auf das DHCP in Netzwerken kann weitreichende folgen für die Netzwerksicherheit mit sich bringen. Durch die Änderung der Client-Adressen und Subnetze kann es zu IP Adresskonflikten im Netzwerk kommen und die Kommunikation beschränkt bzw. stillgelegt werden. Größere Auswirkungen auf die Netzwerksicherheit stellt das Umlenken des Datenverkehrs durch Manipulation der DNS- bzw. Gateway-Parameter dar. Hierbei kann der gesamte Netzwerkverkehr eines Clients umgelenkt werden, um einen MitM Angriff durchzuführen und den Netzwerkverkehr auszulesen.

Da das IP Hilfsprotokoll DHCP keine Sicherheitsmaßnahmen zur Verhinderung dieser Angriffe mit sich bringt, ist es notwendig sich vor diesen Bedrohungen schon auf den unteren Schichten des TCP/IP Referenzmodells zu schützen. Eine in der Industrie weit verbreitete Technik zum verhindern von Angriffen auf das DHCP wird bereits in der Netzzugangsschicht umgesetzt. Netzwerkkomponenten wie Router und Swichtes werden mit Hilfe von DHCP Snooping konfiguriert, welches es ermöglicht DHCP Nachrichten zu überwachen und diese nur von vertrauenswürdigen Ports in das Netzwerk weiterzuleiten. Somit werden DHCP Pakete, welche von einem im Netzwerk eingeschleusten Rogue DHCP Server verteilt werden direkt verworfen und nehmen keinen Einfluss auf das bestehende Netzwerk.

3.6.3 OPC UA

Die OPC UA beschreibt ein mehrschichtiges, generisches Sicherheitskonzept, welches *Transport Layer*, *Communication Layer* und *Application Layer* umfasst. Die Kommunikation des Protokolls OPC UA wird während des Nachrichtenaustauschs über eine Unified Architecture (UA) *Secure Conversation* durchgeführt. Diese wird

vom Netzwerkstack, dem in der Transportschicht genuntzten Protokoll, dem *Secure Channel* sowie der *Session* der Anwendungsschicht (Abbildung 3.8) bestimmt. (OPC Foundation 2018b)

In der *Session* werden die Sicherheitsbestandteile Benutzerauthentifizierung und Benutzerauthorisierung umgesetzt sowie die Nutzdaten übertragen (OPC Foundation 2018b). Die auftretenden Daten werden zur Übertragung an den *Communication Layer* übergeben. Die Sicherheit der *Session* basiert somit auf den Gegebenheiten des *Secure Channel* im *Communication Layer*. Die Form der Nachrichten im *Secure Channel* wird durch die *Security Policy* im gewählten *Message Security Mode* bestimmt.

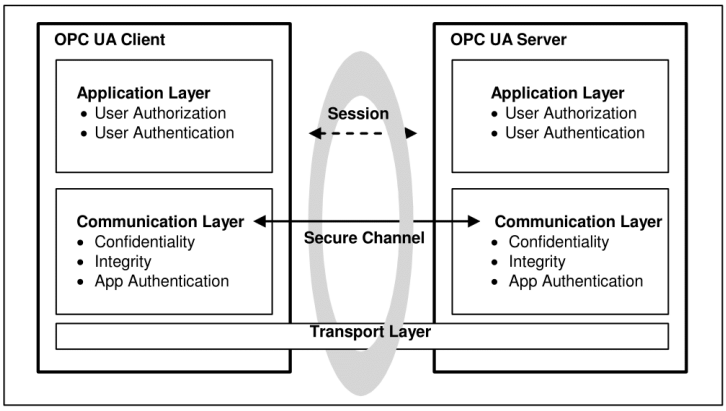


Abbildung 3.8: OPC UA Security Architecture

Aufgrund der verschiedenen Anforderungen der Industrie an ihre Systeme werden von OPC UA mehrere *Security Policies* im *Secure Channel* zur Informationsübertragung zur Verfügung gestellt. Dies ist notwendig, da die Verwendung von Verschlüsselungsalgorithmen und Kodierungsverfahren Ressourcen benötigt und Latenzen verursacht, welche unter Umständen nicht vorhanden sind oder nicht geleistet werden können. Eine Fehlkonfiguration des Protokolls kann jedoch Auswirkungen auf den Betrieb des Netzwerks und der Komponenten haben sowie die Integrität der Daten gefährden. Die Bezeichnungen der geöffneten Verbindungen können irreführend sein, da bei der Verwendung der *Security Policy* „none“ zwar ein *Secure Channel* hergestellt wird, welcher jedoch keine Sicherheitsprofile bereitstellt und somit die unverschlüsselte Kommunikation der Komponenten zulässt (OPC Foundation 2018e).

Auf der Transportschicht beschreibt die Spezifikation von OPC UA mit dem UACP ein abstraktes Protokoll zur Herstellung einer Vollduplexverbindung in einer Client-Server Architektur. Implementierungen dieses Protokolls können über jede Middleware, welche den Austausch von Nachrichten im Vollduplexverfahren über TCP/IP und *Websockets* unterstützt, durchgeführt werden. Somit ist das von OPC UA spezifizierte Protokoll für die Zukunft flexibel. Die Spezifikation des abstrakten Protokolls UACP wird im 5. Teil der OPC UA Spezifikation[37] (OPC Foundation 2018d) beschrieben und beinhaltet die Form der Nachricht, den Verbindungsaufbau, die Kommunikation und die Fehlerbehandlung.

Die *Transport Layer Security* stellt die Sicherheit auf Nachritenebene her und erfüllt die Schutzziele Authentifizierung, Integrität und Vertraulichkeit. Um dies zu ermöglichen, werden verschiedene Transport- sowie Anwendungsprotokolle im Verbund genutzt. Die Sicherheitsmechanismen der Anwendungsschicht dienen der Bereitstellung der Authorisierung und Authentifizierung der Benutzer.

In Abbildung 3.9 werden die von OPC UA bereitgestellten Protokollstacks beschrieben. Diese bestehen aus einer Verbindung von TCP auf Transportebene und dem UA Binary Protokoll zur Kodierung der Nachrichten, einem Webservice über HTTP bzw. Hypertext Transfer Protocol Secure (HTTPS) und Simple Object Access Protocol (SOAP) mit Extensible Markup Language (XML) Encoding oder einer hybriden Form aus beiden. Alle Kommunikationsformen beinhalten mit UA Secure Conversation, TLS und Webservice (WS) Sercure Conversation ein Sicherheitsmodell auf Transport- oder Nachrichtebene. Die Nutzung von Webtechnologien sowie

[37] OPC Unified Architecture Specification Part 6: Mappings

eines binären Protokolls ermöglicht eine hohe Kompatibilität und flexible Anwendungsmöglichkeiten des Protokolls in Umgebungen mit unterschiedlichen Anforderungen.

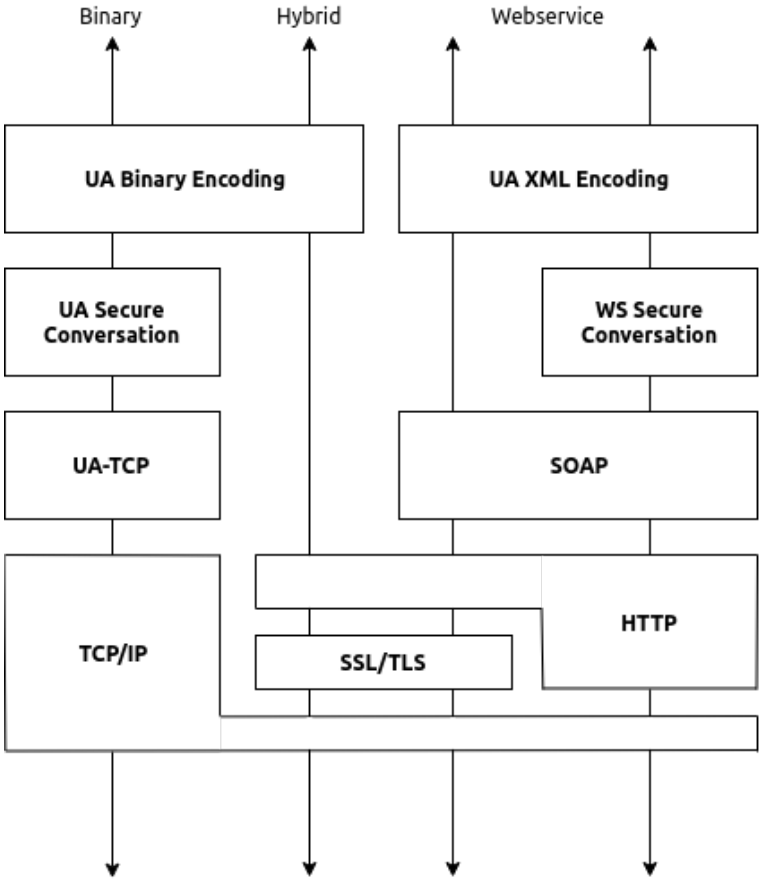


Abbildung 3.9: OPC UA Kommunikationswege

Die beschriebenen Protokollstacks (Abbildung 3.9) bilden mit ihren Sicherheitsmechanismen die Grundlage für eine sichere Datenübertragung. Die Form der Datenübertragung über SOAP/HTTP wurde ab Version 1.03 der Spezifikation als veraltet angesehen, da es in der Industrie nicht umgesetzt wurde (OPC Foundation 2018d) und wird im weiteren Verlauf der Thesis nicht beschrieben. Die Protokolle UA Binary über TCP und die Hybridform aus den Protokollen UA Binary und HTTPS Webservice werden produktiv genutzt und im folgenden näher erläutert.

UA Binary über TCP

Das UA Binary Protokoll über TCP wird für Kommunikation mit optimierter Geschwindigkeit und Durchsatz genutzt. Es besitzt den geringsten Overhead sowie Ressourcenverbrauch, da kein zusätzlicher Parser für HTTP oder XML genutzt werden muss und somit die Systemlast gering gehalten werden kann. Die sichere Kommunikation wird erst auf Nachrichtenebene durch die UA Secure Conversation hergestellt. Für die Transportebene gelten durch das genutzte Protkoll TCP weiterhin die Bedrohungen von Unterabschnitt 3.5.1.

UA Binary über HTTPS

Die Hybridform der Kommunikation über einen HTTPS Webservice mit Hilfe der UA Binary Protokolls vereint die Vorteile des Ressourcenschonenden UA Binary Protokolls über TCP und die weitreichende Kompatibilität eines Webservice. Die Sicherheit der Netzwerkkommunikation wird auf der Transportebene mit Hilfe von TLS hergestellt.

In der aktuellen Version des OPC UA Protokolls wird die Transportsicherheit mit TLS 1.2 und der Cipher Suite *TLS RSA WITH AES 256 CBC SHA256* bereitgestellt (OPC Foundation 2018e). Hierbei ist zu beachten, dass die Rechenleistung der Systeme wächst und somit Verschlüsselungsalgorithmen mit der Zeit unsicher werden. Eine TLS Verbindung mit schwacher Cipher Suite stellt keine sichere Verbindung bereit. Des Weiteren wurden in den Implementierungen von TLS bereits schwerwiegende Fehler verursacht, welche die Transportsicherheit wie z. B. im Falle von *Heartbleed*[38] einschränken können.

[38] TODO - Link zu Heartbleed

Eine weitere Bedrohung stellt das genrelle Verfahren der Ausstellung von Zertifikaten bereit. Diese Zertifikate werden von Dienstleistern ausgestellt, welche als vertrauenswürdig eingestuft und als Root-Certificate Authority (CA) bezeichnet werden. Die Herstellung der Vertrauenswürdigkeit eines Ausstellers liegt im Ermessen des Softwareherstellers und dessen Aufnahme in die Liste vertrauenswürdiger CA.

Die Verwaltung der Zertifikate im Netzwerk kann, um ein erhöhtes Maß an Sicherheit zu gewährleisten, durch Bereitstellung einer CA, Registration Authority (RA) und Validation Authority (VA) im Netzwerk selbst Umgesetzt werden. Diese Bestandteile beschreiben eine Public-Key Infrastructure (PKI). Sie dient der Erzeugung, Verteilung, Überprüfung, Verwaltung und Speicherung der öffentlichen sowie privaten, asymmetrischen Schlüssel und Zertifikate. Die intern erstellten und genutzten Zertifikate können erst für die Kommunikation mit externen Partnern genutzt werden, wenn dieser die CA des Unternehmens als vertrauenswürdig einstuft. Geschieht dies nicht, ist die Herstellung einer vertrauenswürdigen Verbindung nur über ein Zertifikat eines akkreditierten Zertifizierungsdienstanbieters möglich.

3.6.4 CoAP

Um den in Unterabschnitt 2.3.1 beschriebenen Einsatzmöglichkeiten in ressourcenbeschränkten Umgebungen sowie der M2M Kommunikation gerecht zu werden, stellt das Protokoll CoAP, welches im Rahmen des IETF im RFC 7252[39] standadisiert wird, eine einfache Paketstruktur für einen geringen Overhead bei der Kommunikation im Netzwerk bereit. Der Header des Protokolls CoAP besteht aus 32 Bit. Dieser wird von einem *Token*, weiteren *Options* Parametern sowie dem *Payload* gefolgt. CoAP basiert auf dem Transportprotokoll UDP. Der minimale Header sowie die Nutzung von UDP ermöglicht es, auch bei häufiger Kommunikation der Komponenten im Netzwerk die Netzlast gering zu halten. Das Nachrichtenformat des CoAP Protokolls wird in Abbildung 3.10 dargestellt.

[39] Link - https://tools.ietf.org/html/rfc7252

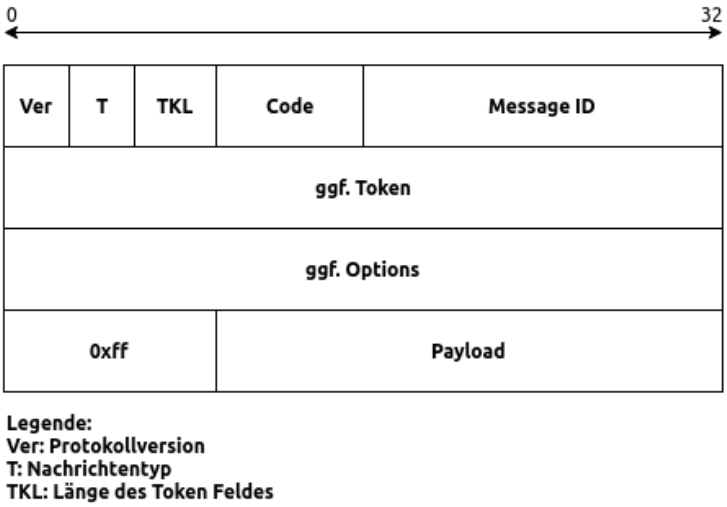


Abbildung 3.10: CoAP Message Format

Die Nutzung des Protokolls UDP macht es weiterhin möglich Multicast Nachrichten im Netzwerk zu verbreiten. Dies ermöglicht das Auffinden von Ressourcen im Netzwerk und ist für M2M Kommunikation in Industrie 4.0 Umgebungen von großer Bedeutung. CoAP Server werden mit Hilfe ihrer Uniform Resource Identifier (URI) referenziert. Werden neue Server im Netzwerk integriert, so ist es möglich, den Server mit Hilfe eines Broad- oder Multicast im Netz oder Teilnetzen bekannt zu machen. (Trapickin 2013)

UDP verhindert jedoch, dass eine Verschlüsselung der Nutzdaten über TLS bereitgestellt werden kann. TLS benötigt eine zuverlässige Übertagung der Pakete[40], da es auf einer sequenziellen Integritätsprüfung[41] der Daten beruht. Diese Anforderungen werden vom unzuverlässigen Datargramm UDP nicht erfüllt. Eine Sicherung der Datenübertragung muss mit Hilfe von Datagram Transport Layer Security (DTLS)[42] hergestellt werden.

3.7 Zwischenfazit

Das Ziel einer weitreichenden Vernetzung aller Komponenten in Industrie 4.0 Umgebungen erfordert ein hohes Maß an Kommunikation zwischen den Komponenten. Neue Industrie 4.0 Netze basieren auf etablierten Technologien wie TCP, UDP und IP und deren Dienste und Übernehmen deren Eigenschaften sowie Vor- und Nachteile der unteren Ebenen des TCP/IP Referenzmodells. Durch die Vereinheitlichung der Netzwerkkommunikation über den IP Stack und der Vernetzung von Industriekomponenten mit Business- und Anwendungsprozessen, spielt die Bereitstellung einer sicheren Kommunikation im Netzwerk und der Schutz der Produktionssysteme vor unbefugten Zugriffen eine zentrale Rolle.

Der Fokus der Weiterentwicklung der Kommunikation im Netzwerk liegt auf der Etablierung neuer Anwendungsprotokolle zur effizienten Nachrichtenübermittlung. Darunter zählt die umfangreichen Industrielösungen OPC UA sowie ressourcenschonende Protokolle für integrierte Lösungen wie CoAP. OPC UA stellt ein zukunftsorientiertes, generisches Protokoll zur Kommunikation im Netzwerk bereit. Durch die hohe Flexibilität, welche durch die weitreichenden Anforderungen der Industrie erforderlich ist, erfordert die Umsetzung einer Infrastruktur mit diesem

[40] TODO - in
[41] TODO - was ist das
[42] TODO - was ist DTLS

Protokoll ein hohes Maß an administrativem Aufwand und kann bei Fehlkonfiguration eine Schwachstelle im Netzwerk darstellen.

Um den in **??** genannten Schutzzielen gerecht zu werden und einen ausreichenden Schutz gegen die in Abschnitt 3.1 genannten Bedrohungen bereitzustellen, muss eine sichere Datenübertragung durch das Zusammenwirken mehrerer Schichten im Netzwerkstack hergestellt werden.

Im folgenden Teil der Thesis werden die in der Analyse gewonnenen Erkenntnisse und deren Auswirkungen auf das Netzwerk und dessen Kommunikation durch die Darstellung verschiedener Bedrohungsszenarien am Testsystem (Weber 2018) untersucht.

Kapitel 4

Anwendungsszenarien

Im folgenden Kapitel wird die Netzwerksicherheit verschiedener Bestandteile des TCP/IP Referenzmodells anhand der in Kapitel 3 durchgeführten Analyse der Netzwerkkommunikation überprüft. Dies wird mit Hilfe des Testsystems (Weber 2018) ermöglicht und soll die Auswirkungen von Bedrohungen der genutzten Technologien darstellen. Die beschriebenen Anwendungsszenarien wurden gewählt, um die Nutzung eines Schichtenmodells im Netzwerk hervorzuheben. Die Einführung eines neuen Protokolls auf einer Ebene des Referenzmodells hat keine Auswirkungen auf die Funktionsweise der anderen, genutzten Schichten. Jede Schicht im Netzwerk kann durch Sabotage die Manipulation des Netzwerk ermöglichen.

Die Umsetzung der Anwendungsszenarien SYN-Flood, ARP Spoofing und DNS Amplifictation war nicht möglich, da die Manipulation des Netzwerks und dessen Kommunikation im gegebenen Testsystem aufgrund der genutzten Software nur begrenzt ausführbar war und somit die Bereitstellung einer Netzwerkinfrastruktur wesentlicher Bestandteil der Thesis darstellte. Dies wird in Abschnitt 5.4 näher erläutert.

Nach Abwägung der Faktoren zeitlicher Aufwand, Umsetzbarkeit und Darstellungsmöglichkeit und Mehrwert für das System, wurde sich für die folgenden Anwendungsszenarien entschieden.

4.1 OPC UA Kommunikation

Die Kommunikation des OPC UA Protokolls findet laut Spezifikation, wie in Unterabschnitt 3.6.3 beschrieben, immer im *Secure Channel* statt. Der *Secure Channel*

stellt verschiedene *Security Policies* für unterschiedliche Anwendungsfälle zur Verfügung. Administratoren müssen abwägen in wie Fern Ressourcen und Latenz für die Absicherung der Kommunikation im Netzwerk gewährleistet werden können. Die Kommunikation zwischen OPC UA Komponenten und die verschiedenen Sicherheitsprofile können im vorhandenen Testsystem (Weber 2018) untersucht werden. Hierfür wird das Netzwerkanalysetool Wireshark[1] genutzt, um den Netzwerkverkehr zwischen den Komponenten abzuhören. In Verbindung mit dem OPC UA *Secure Channel* wird das System erweitert, um verschiedene Sicherheitsprofile für die Kommunikation bereitzustellen und somit die Auswirkungen einer Fehlkonfiguration darstellen zu können.

4.2 MitM

Im vorhandenen System soll die Darstellung eines MitM Angriffs ermöglicht werden. Dieser wird mit Hilfe eines Rogue DHCP Servers durchgeführt. Es soll ermöglicht werden, die Netzwerkkonfiguration einer Komponente so zu Manipulieren, um die Kommunikation mithören zu können. Dies beinhaltet die Bereitstellung einer Netzwerkinfrastruktur inklusive DHCP und DNS.

4.3 Manipulation von ungesichertem Netzwerkverkehr

Im dritten Anwendungsszenario sollen die durch den durchgeführten MitM Angriff gewonnenen Informationen genutzt werden, um die Funktionalität eines weiteren Systems im Netzwerk zu stören und somit direkt Einfluss auf einen Prozess in einem Industrienetzwerk nehmen. Aufgrund der in Unterabschnitt 2.3.1 beschriebenen Faktoren und der immer weiteren Vernetzung ressourcenschwacher Komponenten soll das Anwendungsszenario anhand einer minimalen IIoT Komponente am Protokoll CoAP durchgeführt werden.

[1] Link: https://www.wireshark.org/

Kapitel 5

Konzept

Im folgenden Kapitel wird das entwickelte Konzept zur Erweiterung des Testsystems (Weber 2018) beschrieben. Das Konzept wurde entwickelt, um anhand der in Kapitel 3 gewonnenen Erkenntnisse, verschiedene Angriffsformen in einer Industrie 4.0 Netzwerkumgebungen und deren Einfluss auf das System und die Netzwerkkommunikation demonstrieren zu können. Da die entstehende Integration und Implementierung des Konzepts für Test- und Lehrzwecke genutzt werden soll, in Zukunft um weitere Funktionalitäten erweitert werden soll und flexibel Einsetzbar sein soll, stehen die Aspekte Portabilität, Skalierbarkeit und Erweiterbarkeit bei der Entwicklung des Konzepts im Vordergrund.

Das Konzept beinhaltet die Darstellung der Verteilungs-, Baustein- und Laufzeitsicht. Dabei wird die entwickelte Netzwerkinfrastruktur, die einzelnen Komponenten sowie die Gesamtfunktionalität des Systems beschrieben.

Um eine ganzheitliche Infrastruktur mit den grundlegenden Netzwerkdiensten bereitstellen zu können, kann die Kommunikation der Systeme nicht auf der Abstraktionsebene des Docker Containernetzwerks stattfinden. Das IPAM im Netzwerk sowie die gewünschten Dienste DHCP und DNS können dort nicht ausreichend konfiguriert werden. Es mussten im Laufe der Entwicklung, aufgrund des gegebenen Systems und der genutzten Software, mehrere Konzepte zur Umsetzung verschiedener Anwendungsszenarien verworfen werden. Diese werden in **??** kurz beschrieben. Die Softwarebeschränkungen und deren Auswirkungen auf die Umsetzung des Konzepts sowie der Integration werden in Abschnitt 5.4 erläutert.

5.1 Verteilungssicht

Die Skalierbarkeit und Erweiterbarkeit des Testsystems wird auf der Verteilungssicht durch die Nutzung einer VM und deren Docker Container realisiert. Um die Heterogenität des Systems gering zu halten, jedoch ein vollwertiges Netzwerk abbilden zu können, werden der vorhandenen Grundstruktur des Testsystems (Weber 2018) VM zwei weitere Maschinen hinzugefügt.

Die VMs simulieren, im Gegensatz zur Containervirtualisierung ein vollständiges System sowie dessen Hardware. Dies ermöglicht eine weitgehende Kontrolle über die Netzwerkschnittstellen und die Konfiguration gekapselter Netzwerkinfrastrukturen. Diese Vorteile werden genutzt, um die Netzwerkkommunikation mit Hilfe von virtuellen, internen Netzwerken auf der Abstraktionsebene der VMs durchzuführen.

Das Konzept der Netzwerkinfrastruktur ist in Abbildung 5.1 in Anlehnung an ein UML-Verteilungsdiagramm dargestellt. Zur Umsetzung der in Kapitel 4 beschrieben Szenarien, ist es notwendig den Netzwerkdienst DHCP bereitzustellen. Dieser wird, zusammen mit dem Dienst DNS im Netzwerk „i40-network“ bereitgestellt. Das zusätzliche Netzwerk „i40-monitoring“ wird genutzt, um die Netzwerkkommunikation über ein Gateway umzusetzen und somit einen MitM Angriff durchführen zu können. Die Kommunikation zwischen den Netzwerken wird über Router realisiert.

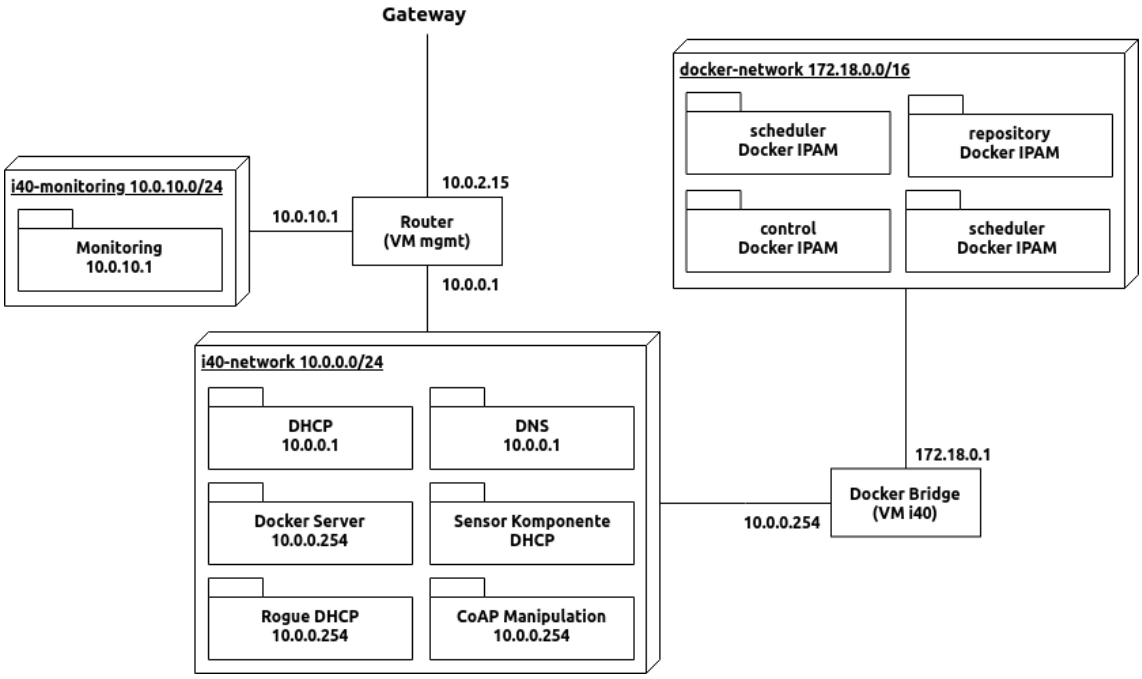


Abbildung 5.1: Netzwerkinfrastruktur

Die Kapselung und Sicherheit des Testsystems ist durch die Erweiterung um zusätzliche VMs und Dienste weiterhin gegeben. Die Kommunikation mit dem Hostsystem und externen Netzen findet ausschließlich über die Netzwerkschnittstelle des Routers statt und dient der Installation und Konfiguration der Komponenten. Diese kann im Betrieb deaktiviert werden, um eine vollständige Isolation der Umgebung zu erzielen.

Durch die Konfigurationsmöglichkeiten der Hardware und des Netzwerks ist eine weitere Kapselung der Netzwerke mit Hilfe von VLAN auf der Netzzugangssicht umsetzbar. Dies wird im Rahmen der Thesis jedoch aus zeitlichen Gründen nicht weiter bearbeitet.

5.1.1 Industrienetzwerk

Das Industrienetzwerk „i40-network“ stellt den zentralen Bestandteil der Architektur dar. Dieses Netzwerk wird von den Diensten DHCP und DNS verwaltet und beinhaltet das Industrie 4.0 Produktionssystem und dessen Docker Sub-Architektur sowie die Sensor Komponente, welche Daten zum CoAP Monitoring liefert. Der Netzwerkverkehr sowie die Verwaltung des Containernetzwerks wird vom Docker Dienst und dessen Netzwerkbrücke übernommen. Die Kommunikation mit dem Monitoringnetzwerk sowie mit externen Netzen wird mit Hilfe von IP Forwarding und Network Address Translation (NAT) hergestellt. Der Rogue DHCP Server befindet sich ebenfalls in diesem Netz und dient der Manipulation des für das Netzwerk *authorativen* DHCP Servers. Die Umsetzung dieses Netzwerks, der beinhalteten Dienste und der vollständigen Kontrolle über diese ist die Basis für die Durchführung der Anwendungsszenarien.

5.1.2 Monitoring-Netzwerk

Das Netzwerk „i40-monitoring“ simuliert ein zusätzliches Netzwerk, welches Dienste zur Analyse und Überwachung der Komponenten im Netzwerk bereitstellen soll. Im beschriebenen Konzept besitzt das Monitoring-Netzwerk ausschließlich eine Komponente zur Überwachung der gesendeten Daten des im „i40-network“ vorhandenen Sensors. Eine weitere Ausführung der Komponenten im Netzwerk hätte keinen Einfluss auf die gewählten Anwendungsszenarien (Abschnitt 4.2 und Ab-

schnitt 4.3) gehabt, da die Kommunikation über ein Gateway der ausschlaggebende Faktor für die Integration des zusätzlichen Netzwerks war. Aus diesem Grund wurde auch kein DHCP und DNS für das Netzwerk konfiguriert. Die Adressvergabe verläuft statisch.

5.1.3 Containernetzwerk

Das Containernetzwerk beschreibt das grundlegende Testsystem und wird in Weber 2018 beschrieben. Die bereitgestellten Container der Industrie 4.0 Umgebung werden durch den Docker Server verwaltet. Dieser übernimmt das IPAM sowie den DNS. Die Kommunikation zwischen den Container sowie mit dem Hostsystem findet über eine Software-Bridge[1] statt. Diese Form der Netzwerkkommunikation ermöglicht es das Anwendungsszenario Abschnitt 4.1 durchzuführen. Die Software-Bridge nimmt den gesamten Netzwerkverkehr entgegen und leitet ihn weiter. Somit ist es möglich diesen vom System, welches die Container verwaltet, zentral zu untersuchen.

5.2 Bausteinsicht

Die aus den Anwendungsszenarien (Kapitel 4) hervorgehenden Anforderungen müssen durch die Komponenten im System umgesetzt werden. Hinzu kommt, dass die beschriebenen Netzwerke durch erforderlichen Dienste verwaltet werden müssen, um die Kommunikation der Komponenten zu gewährleisten. Die logische Darstellung der Verteilungssicht unterscheidet sich grundlegend von der Umsetzung der Komponenten in der Bausteinsicht. Die Virtualisierung der Maschinen erlaubt es mehrere Komponenten auf einem System zusammenzufassen und mit mehreren Netzwerkschnittstellen in verschiedenen Netzen auszustatten, um multiple Systeme darstellen zu können.

In Abbildung 5.2 werden die VMs sowie deren Dienste und Schnittstellen in die Netzwerke dargestellt. Die Dienste DHCP, DNS im Netzwerk „i40-network“ sowie die generellen Routingfunktionalitäten werden von der VM „mgmt“ bereitgestellt. Des Weiteren wird dort der CoAP Server umgesetzt, welcher sich im Netzwerk „i40-monitoring“ befindet. Die VM „i40“ beinhaltet die Komponenten zur Manipu-

[1] TODO - eine Bridge ...

lation des Netzwerks sowie das Containernetzwerk. Der CoAP Client des Netzwerk „i40-network“ wird auf der VM „comp“ ausgeführt.

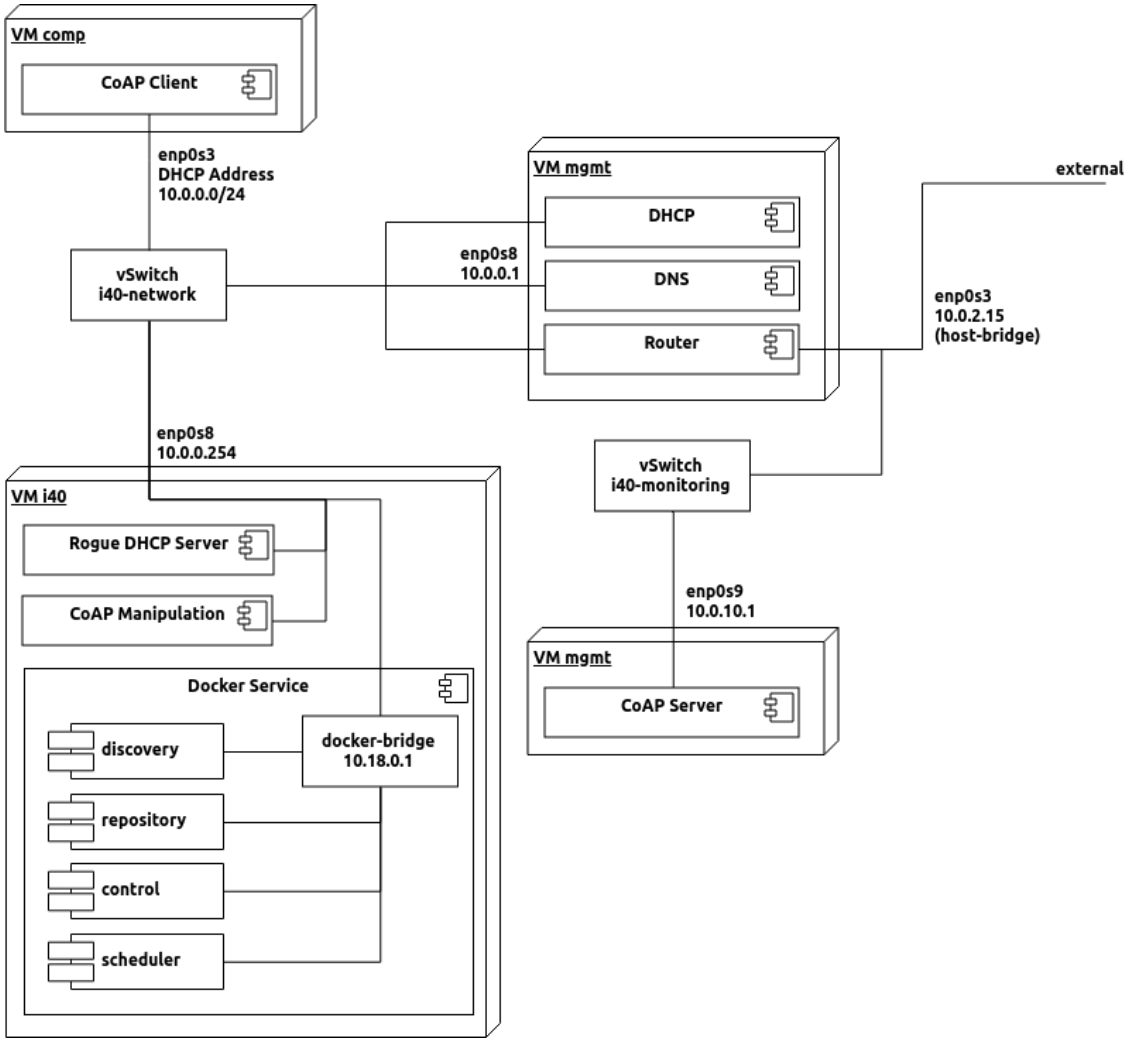


Abbildung 5.2: Virtuelle Maschinen und Dienste

5.2.1 Router

Die Routingfunktionalitäten werden auf der VM „mgmt“ umgesetzt. Die VM besitzt in jedem Netzwerk ein Interface, welches als Schnittstelle dient und zur Weiterleitung der Pakete genutzt wird. Die VM „mgmt“ ist die einzige Komponente der Architektur, welche eine Verbindung zum Hostsystem besitzt. Die Weiterleitung der Pakete wird auf der Internetschicht des TCP/IP Referenzmodells mit Hilfe von IPv4 Forwarding und NAT umgesetzt. Durch das Routing zum Hostsystem ist eine Verbindung der anderen, gekapselten Maschinen sowie zukünftiger Maschinen

für Installations- und Konfigurationszwecke in externe Netzwerke möglich. Um die Sicherheit des Netzwerkverkehrs sicherzustellen dürfen nur Pakete von Verbindungen, welche aus einem der internen Netze initiiert wurden weitergeleitet werden. Abbildung 5.3 stellt die Schnittstellen der Komponente dar.

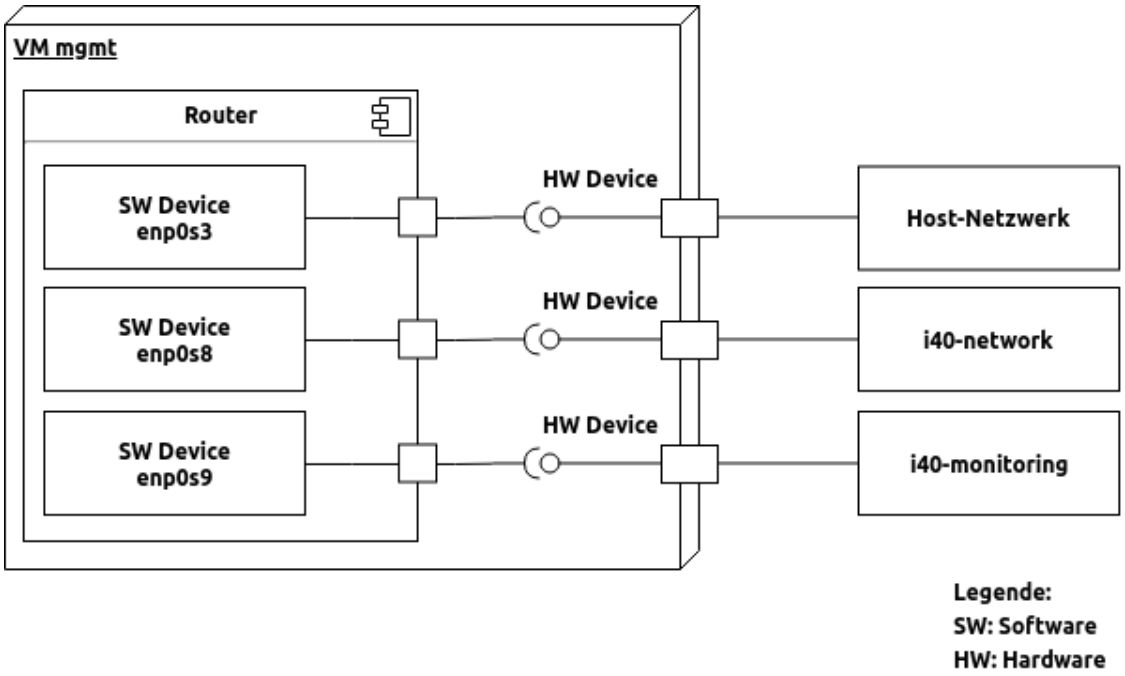


Abbildung 5.3: Routerkomponente

5.2.2 DHCP Server/DNS Server

Die Dienste DHCP und DNS werden zusammen auf der VM „mgmt“ konfiguriert stellen ihre Dienste im Netzwerk „i40-network“ bereit, indem sie auf die zuständige Schnittstelle gebunden werden. Zur Umsetzung der Anwendungsszenarien Abschnitt 4.2 und Abschnitt 4.3 liegt der Fokus der bereitzustellenden Funktionalität auf dem DHCP Server. Dieser muss den weiteren Komponenten im Netzwerk beim DHCP *discover*, welcher über einen Broadcast[2] durchgeführt wird, eine Netzwerkkonfiguration bereitstellen. Dabei ist die Vergabe eines Standardgateways für die spätere Durchführung des MitM Anwendungsszenario essentiell. Das IPAM sowie die Konfiguration des DNS und dessen dynamische Zonenaktualisierungen spielen bei der Umsetzung des Angriffs nur eine nebensächliche Rolle. Sie werden jedoch trotzdem umgesetzt, da die Konfiguration des IPAM und der *address range*

[2] TODO Broadcast

der DHCP *leases* eine Möglichkeit gibt den Wechsel des DHCP Servers zu Lehrzwecken visualisieren zu können. Die Bereitstellung eines mit dem DHCP Server kompatiblen DNS Servers vereinfacht durch die Namensauflösung die Verwaltung der virtuellen Maschinen in der Testumgebung und ermöglicht eine Skalierbare Erweiterung des Systems. Die Verwaltung eines separaten DNS Servers ermöglicht die Umsetzung weiterer Anwendungsszenarien und die Analyse des Sicherheitsmechanismus DNSSEC am Testsystem. Diese Anwendungsmöglichkeiten sind kein Bestandteil der Thesis und werden im folgenden nicht weiter erläutert. Sie bieten jedoch Ansatzmöglichkeiten für zusätzliche Erweiterungen am System und werden in Kapitel 9 näher beschrieben.

Abbildung 5.4 zeigt die Komponenten DHCP Server und DNS Server und die genutzten Schnittstellen in Bezug auf das Gesamtsystem. Beide Dienste werden auf das Netzwerkinterface des Netzwerks „i40-network“ gebunden. Somit werden die Dienste ausschließlich im beschriebenen Netzwerk bereitgestellt.

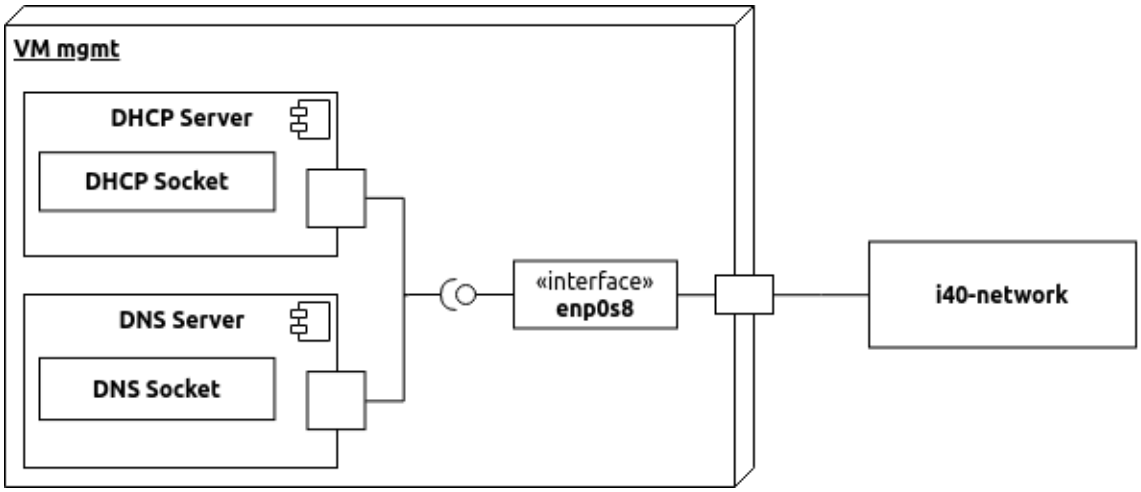


Abbildung 5.4: DHCP und DNS Server

5.2.3 CoAP Client/CoAP Server

Die Komponenten CoAP Client und CoAP Server dienen der Erweiterung des Testsystems um ein weiteres IIoT Protokoll zur Darstellung des in Abschnitt 4.3 beschriebenen Szenarios. Die Komponenten simulieren einen Temperatursensor sowie ein Monitoringsystem, welches ein Webinterface zur Darstellung der gemessenen Temperatur besitzt. Der CoAP Client wird auf der VM „comp“ realisiert und befindet sich im Netzwerk „i40-network“. Er sendet die gemessenen Daten zum zu-

ständigen CoAP Server im Netzwerk „i40-monitoring“. Der CoAP Server wird, da er den einzigen Dienst in diesem Netzwerk darstellt, auf der VM „mgmt“ bereitgestellt. Die VM „mgmt“ besitzt bereits ein Interface im Netzwerk „i40-monitoring“. Der Dienst kann auf die Adresse der Schnittstelle gebunden werden. Es muss keine weitere VM zum Netzwerk hinzugefügt werden.

Abbildung 5.5 und Abbildung 5.6 visualisieren die Bestandteile der betroffenen Komponenten, deren bereitgestellte Dienste und die genutzten Schnittstellen.

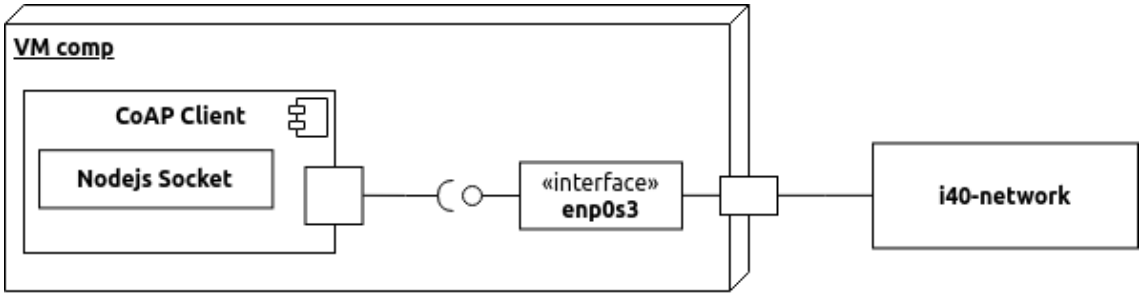


Abbildung 5.5: CoAP Client

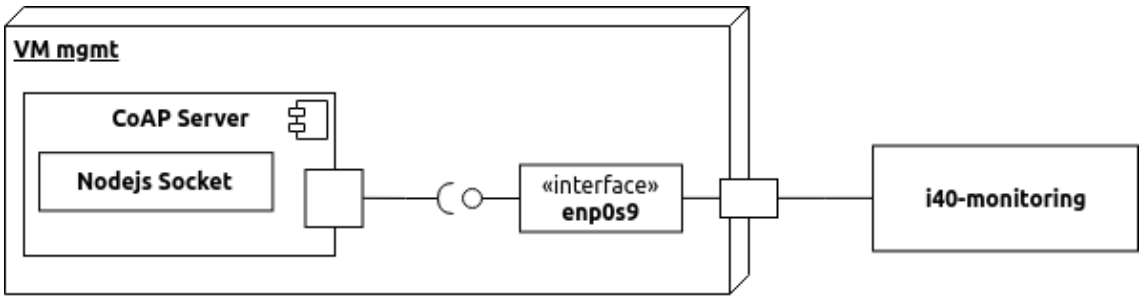


Abbildung 5.6: CoAP Server

5.2.4 Rogue DHCP Server

Die Bereitstellung des Rogue DHCP Servers findet auf der bestehenden VM „i40“ statt. Die bestehende virtuelle Maschine stellt bereits Tools zu Analyse der Netzwerkkommunikation für das Dockernetzwerk bereit. Eine Erweiterung dieses Systems ermöglicht es die bereitgestellten Anwendungen zur Analyse in den Anwendungsszenarien des Netzwerks der VMs ebenfalls zu nutzen. Dies wird durch die Trennung der Netzwerkschnittstellen ermöglicht. Die Analyse der Pakete findet fortan auf der Schnittstelle „enp0s8“ statt der Docker Netzwerkbrücke statt. Um die Netzwerkpakete der anderen Teilnehmer über die Schnittstelle der VM zu leiten, muss die Maschine selbst als Gateway für die Kommunikation zwischen den

Netzen genutzt werden. Der Rogue DHCP Server ermöglicht die Änderung des Standardgateways bei Erneuerung des DHCP *lease* der Client und kann somit den Verkehr auf das lokale System umleiten. **??** zeigt die Eingliederung des Dienstes in das vorhandene System. Der Dienst muss im Netzwerk „i40-network“ agieren und wird somit auf das Interface „enp0s8“ gebunden.

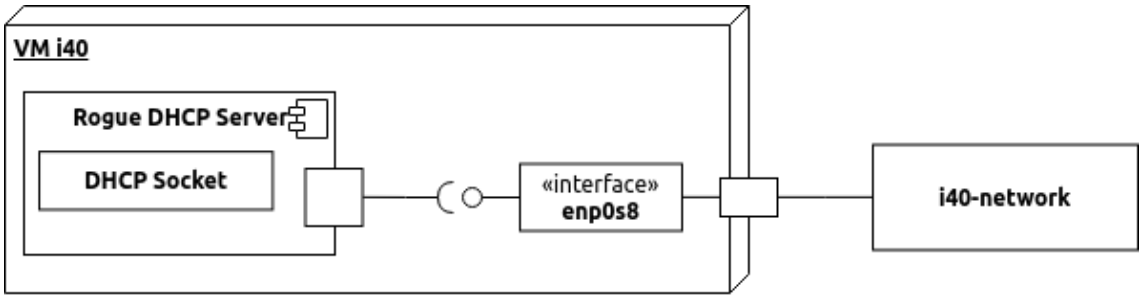


Abbildung 5.7: Rogue DHCP Server

5.2.5 CoAP Manipulationssystem

Das CoAP Manipulationssystem wird genutzt, um das CoAP Monitoringsystem durch falsche bzw. unzulässige Netzwerkpakete zu beeinflussen. Es wird ebenfalls auf der VM „i40“ bereitgestellt. Dies ermöglicht die weiterhin die zentrale Verwaltung des Systems und die Durchführung der Anwendungsszenarien. Eine Bereitstellung des Manipulationsdienstes auf dem gleichen System wie der Rogue DHCP Server und die Anwendungen zur Netzwerkanalyse ist sinnvoll, da die Umleitung der Netzwerkpakete durch den Rogue DHCP Server und die anschließende Analyse des Verkehrs die Grundlage für die Manipulation des Monitoringsystems darstellt und voneinander abhängig ist.

Das CoAP Manipulationssystem versendet Netzwerkpakete mit gefälschten Daten zum CoAP Monitoringsystem. Diese werden über die Netzwerkschnittstelle „enp0s8“ in das „i40-network“ versandt. Die Verbindung in das Netzwerk „i40-monitoring“ wird über die Routingfunktionalitäten der VM „mgmt“ bereitgestellt. Abbildung 5.8 beschreibt den Aufbau der Komponente.

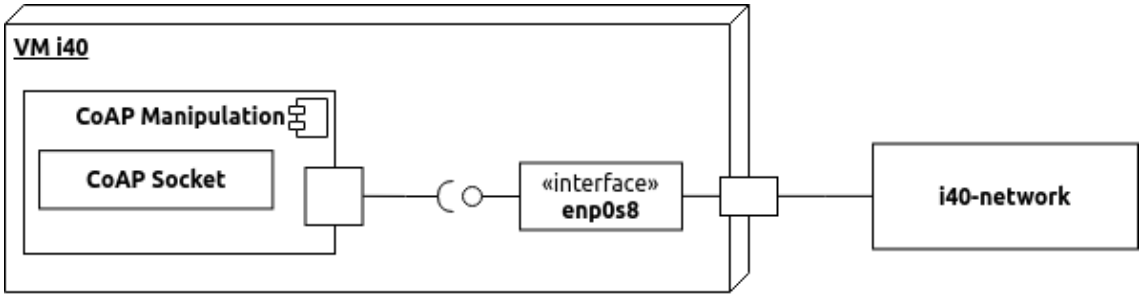


Abbildung 5.8: CoAP Manipulationssystem

5.2.6 Docker Service

An der Netzwerkkonfiguration sowie am auf der VM „i40“ bereitgestellten Docker Service wurden keine Änderungen vorgenommen. Die genutzte Konfiguration des Dienstes ist in Weber 2018 erläutert. Eine Integration der neuen Komponenten als zusätzliche Container im bestehenden Netzwerk war nicht möglich, da dies die Analyse des Netzwerks durch die Netzwerkimplementierung der Software Docker verfälscht bzw. verhindert. Eine genaue Beschreibung der Beschränkungen, welche bei der Umsetzung des Systems zum Tragen kamen, wird in **??** durchgeführt.

5.3 Laufzeitsicht

Die Laufzeitsicht dient der Darstellung der Kommunikation zwischen des einzelnen Komponenten zur Bereitstellung einer Gesamtfunktionalität. In diesem Abschnitt werden die essentiellen Ablaufe des Systems in Anlehnung an UML-Sequenzdiagramme dargestellt, welche die Netzwerkkommunikation und -konfiguration zwischen den Komponenten bereitstellen.

5.3.1 Routing

TODO - Routing darstellen.

5.3.2 DHCP

Im Netzwerk „i40-network“ werden die Dienste DHCP und DNS zur dynamischen Konfiguration und Namensauflösung der Hosts genutzt. Abbildung 5.9 zeigt die

Konfiguration eines Hosts beim Bezug der Netzwerkkonfiguration im „i40-network“ mit Hilfe des zuständigen DHCP Servers. Die Netzwerknachrichten der DHCP Konfiguration werden, da der Client zum Zeitpunkt der dynamischen Konfiguration noch keine IP Adresse im Netzwerk besitzt, über den Broadcast im Netzwerk versandt. Die Pakete werden ausschließlich von DHCP Servern angenommen und verarbeitet, weitere Clients im Netzwerk verwerfen die Pakete. Im Sequenzdiagramm ist, um die Übersichtlichkeit zu gewährleisten nur die Kommunikation zwischen den aktiven Beteiligten während der dynamischen Konfiguration des Hosts dargestellt.

Um vollständige Funktionalität des Clients im Netzwerk bereitzustellen, werden IP Adresse, Netzwerkmaske, DNS sowie ein Gateway zur Verbindung in andere Netze benötigt. Nach dem DHCPDISCOVER des Clients bietet der DHCP Server mit Hilfe des DHCPOFFER Netzwerkkonfigurationen für den Client an. Dieser bestätigt die Konfiguration mit einem DHCPREQUEST. Ist die vom Server angebotene Konfiguration gültig und die IP Adresse weiterhin frei, wird vom Server ein DHCPACK gesendet, um die Konfiguration zu bestätigen.

DNS und DHCP arbeiten im Testsystem zusammen. Der DHCP Server muss den DNS Server über neue Clients im Netzwerk informieren, um eine Namensauflösung dieser bereitstellen zu können. Die Kommunikation zur Aktualisierung der DNS Zonen geschieht wird mit Hilfe eines symmetrischen Schlüssels gesichert. Beide Komponenten müssen diesen Schlüssel besitzen, um sich beim anderen Dienst zu Authentifizieren. Die dynamische Aktualisierung einer DNS Zone wird ebenfalls in Abbildung 5.9 beschrieben.

5.3.3 DNS

Die Namensauflösung der internen Adressen findet direkt auf dem DNS Server statt. Bei einem DNS Request werden die lokalen Zonendateien durchsucht und in der Answer Section der DNS Response die gewünschten RR bereitgestellt. Das interne Netzwerk „i40-network“ besitzt eine Zone auf dem Nameserver. Dort müssen alle statischen Adressen zur Namensauflösung eingetragen werden. Die dynamischen Adressen werden ebenfalls in dieser Zone temporär hinterlegt. Um eine Namensauflösung in externe Netze zu gewährleisten, muss der DNS Server weitere Nameserver kennen, um Domainnamen, welche nicht in der Datenbank des lokalen Servers vorhanden sind, aufzulösen. Das DNS System nutzt Rekursion oder Iteration, um den *authorativen* Server im Netzwerk zu bestimmen und die Namensauflösung be-

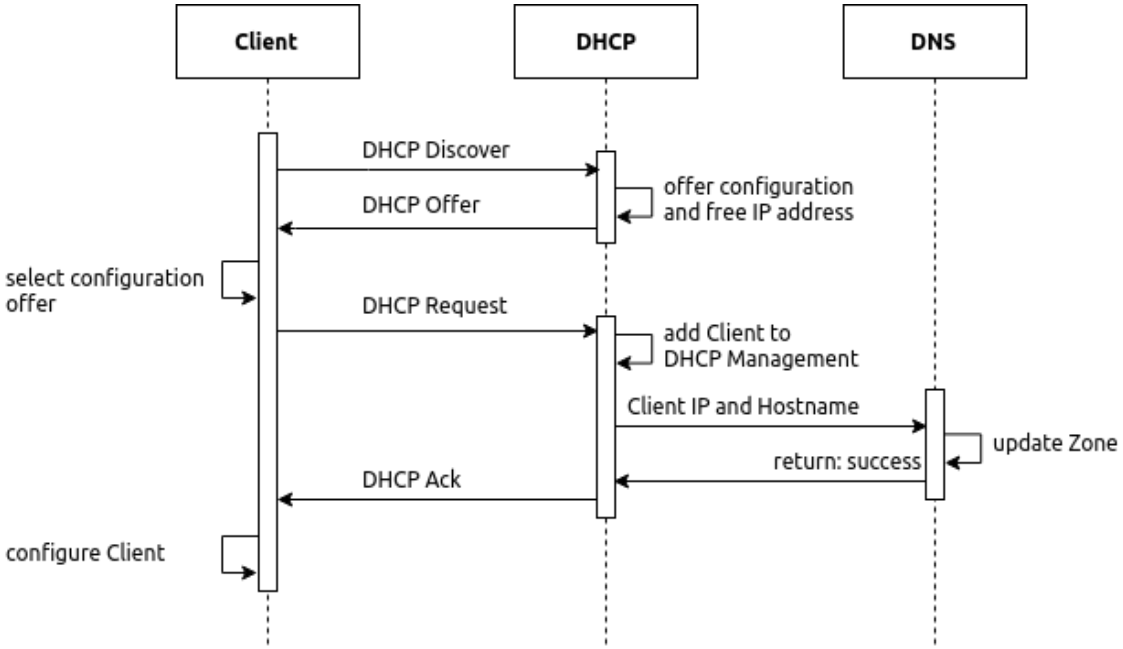


Abbildung 5.9: Dynamische Hostkonfiguration mit DHCP und DNS

reitzustellen. Der Ablauf der Namensauflösung wird vom zuständigen Nameserver bestimmt. Abbildung 5.10 stellt den Ablauf der rekursiven Namensauflösung einer externen DNS Zone vereinfacht dar. Der lokale DNS Server arbeitet als DNS-Forwarder. Er leitet die Anfrage zur unbekannten Zone an einen externen DNS weiter, welcher die Adresse auflöst und dem internen Nameserver das Ergebnis mitteilt. Die Rekursion der Befragung weiterer DNS Server findet solange statt, bis ein für die Domain zuständiger Nameserver gefunden wurde, welcher die Adresse auflöst. Bei der iterativen Namensauflösung sendet der DNS Server dem Client die Adresse des *authorativen* DNS Servers zu. Dieser führt dann einen erneuten DNS Query bei diesem Server durch. Der Ablauf der iterativen Namensauflösung wird in Abbildung 5.11 dargestellt.

Die genutzte Form der Namensauflösung hat auf die Umsetzung des Konzepts keinen Einfluss. Um eine Namensauflösung in externe Netze bereitzustellen muss der DNS Server lediglich eine erreichbare DNS Forward-Adresse besitzen.

5.3.4 Kommunikation CoAP Client und Server

Das Zusammenspiel der bisher beschriebenen Komponenten wird für die Kommunikation zwischen CoAP Client und CoAP Server benötigt. Im in Abschnitt 4.3

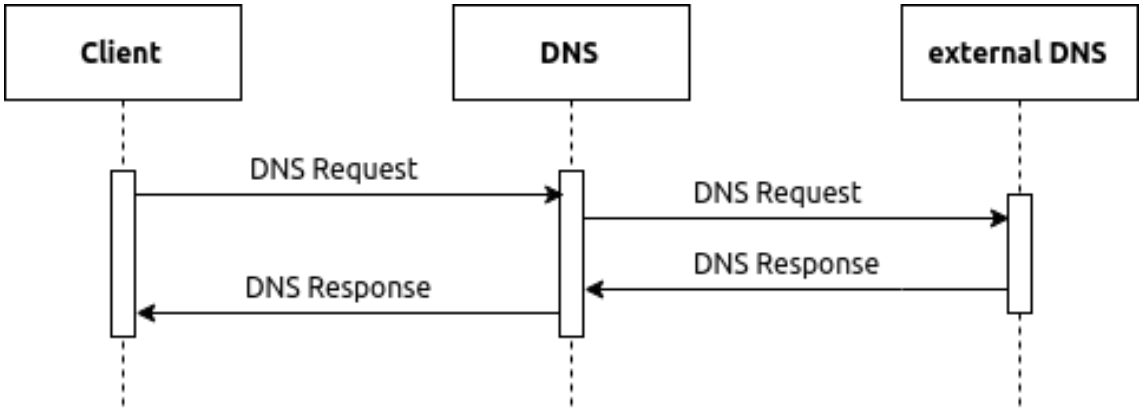


Abbildung 5.10: Rekursive DNS Namensauflösung

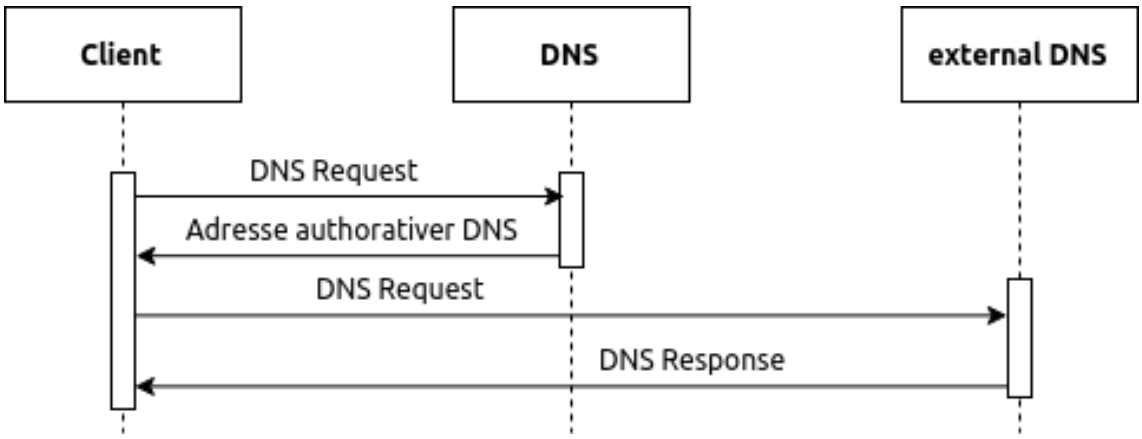


Abbildung 5.11: Iterative DNS Namensauflösung

beschriebenen Anwendungsszenario findet zwischen den Komponenten ausschließlich eine unidirektionale[3] Kommunikation vom Client zum Server statt. Abbildung 5.12 zeigt die Netzwerkkonfiguration des Clients und die genutzten Komponenten bei der Übermittlung einer Nachricht vom Client zum Server und steht beispielhaft für die Kommunikation aller Komponenten im Netzwerk. Um die Übersichtlichkeit der Darstellung zu erhalten, findet nur noch eine einfache Beschreibung der Initialisierung des Clients statt. Diese wurde bereits in Abbildung 5.9 beschrieben. Die Kommunikation zwischen CoAP Client und DHCP findet im Netzwerk „i40-network“ statt. Die Kommunikation zwischen dem Client und Server erfordert die Nutzung eines Routers, da sich die Serverkomponente im Netzwerk „i40-monitoring“ befindet. Die Struktur der Netzwerkkommunikation gilt für alle Komponenten des Netzwerks „i40-network“.

[3] TODO - Unidirektional -> in eine Richtung

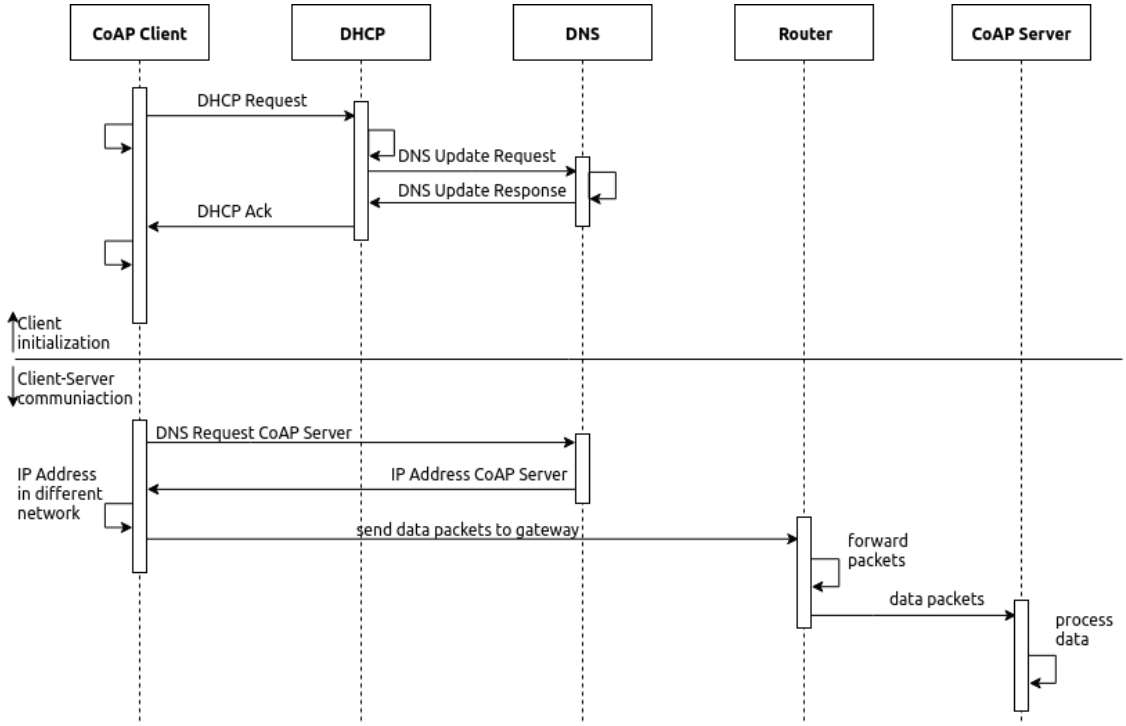


Abbildung 5.12: Initialisierung von CoAP Client und Kommunikation zwischen CoAP Client und CoAP Server

5.3.5 Manipulation des Netzwerkverkehrs

Durch die Aktivierung der Rogue DHCP Komponente auf der VM „i40“ wird ein zweiter DHCP Server im Netzwerk aktiviert. Komponenten, welche über einen DHCP Discover den für das Netzwerk zuständigen DHCP Server kontaktieren möchten, können nun auch eine Antwort des Rogue DHCP Servers erhalten. Dieser stellt eine DHCP Konfiguration mit sich selbst als Gateway bereit. Trifft der DHCP Offer des Rogue DHCP Servers vor dem DHCP Offer des eigentlichen DHCP Servers beim Client ein, sendet dieser den gesamten Netzwerktraffic, welcher für andere Netze bestimmt ist über den neuen DHCP Server. Um die Netzwerkkommunikation der Systeme, welche den Rogue DHCP Server als Gateway nutzen weiterhin bereitzustellen, muss dieser die Pakete wieder zum ursprünglichen Router weiterleiten. Der Ablauf der Kommunikation wird in Abbildung 5.13 vereinfacht beschrieben. DHCP Nachrichten werden nicht gerichtet an Systeme gesendet, diese werden über den Broadcast des Netzwerks kommuniziert. Die DNS Namensauflösung wurde in der Darstellung nicht aufgeführt, da sie zur Umleitung der Netzwerkpakete in diesem Beispiel keinen Beitrag leistet.

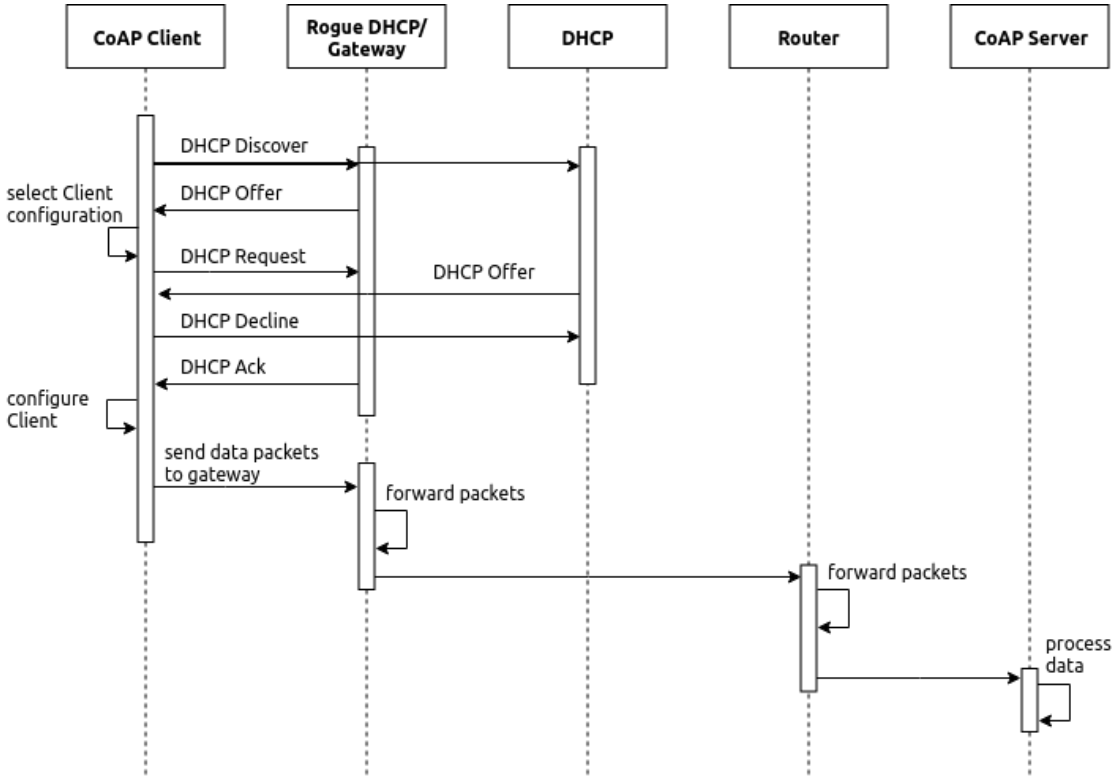


Abbildung 5.13: Umleitung der Netzwerkkommunikation durch Manipulation des Gateways

5.4 Anpassungen

Im Laufe der Entwicklung des Konzepts mussten immer wieder Anpassungen vorgenommen werden. Diese bezogen sich auf die Umsetzbarkeit des Konzepts in Bezug zu dem im Testsystem bisher verwendeten Softwarestack.

Ursprünglich wurde die Erweiterung des Testsystems um zusätzliche Docker Container, welche die aus den in Kapitel 4 beschriebenen Anwendungsszenarien hervorgehenden Anforderungen sowie die benötigte Netzwerkinfrastruktur umsetzen, geplant. Dies war nicht möglich, da die Containervirtualisierung kein vollständiges System inklusive Hardwarekomponenten simuliert, sondern lediglich ein isoliertes Gastsystem darstellt, welches den Kernel und die Komponenten des Hostsystems nutzt. Die Netzwerkimplementierung der genutzten Software Docker stellt vier Netzwerktreiber[4] zur Kommunikation der Container mit anderen Komponenten bereit. Jeder dieser Treiber stellt ein eigenes, statisches IPAM mit Hilfe des Docker Service bereit. Der Treiber *macvlan* simuliert zwar eine physikalische Schnittstelle der Container mit Hilfe einer **MAC!** (**MAC!**) Adresse, jedoch übernimmt der Do-

[4]Docker Netzwerktreiber - https://docs.docker.com/network/

cker Service weiterhin das IPAM. Die Adressvergabe und Netzwerkkonfiguration ist somit weiterhin nur statisch beim Start der Container möglich.

Eine Alternative würde die Nutzung von Pipeworks[5] darstellen. Die Software ermöglicht es die Container zu einem logischen, physikalischen Interface zu verbinden, indem sie im Container einen zusätzlichen Netzwerkadapter bereitstellt. Somit könnte die externe Verwaltung der Docker Netzwerkschnittstellen über einen DHCP Server bereitgestellt werden. Die Software stellt keine solide Implementierung bereit und rät dazu wenn möglich auf die offiziellen Implementierungen von Docker zurückzugreifen. Mit der Erweiterung des Systems um zusätzliche Softwareimplementierungen können weitere Nebeneffekte oder Probleme bei der Umsetzung der Anwendungsszenarien aufgrund von Softwarebeschränkungen entstehen.

Da keiner der Beschriebenen Ansätze eine zufriedenstellende Lösung zur Umsetzung der Anwendungsszenarien bereitstellt, wurde die Erweiterung des Systems und die Darstellung verschiedener Netzwerkkomponenten und deren Kommunikation mit Hilfe von VMs realisiert. VMs werden in Industrie 4.0 Netzwerken weitreichend und produktiv genutzt. Sie können die Infrastruktur repräsentieren und bieten auch für die Zukunft eine stabile Grundlage zur Integration weiterer Dienste im Testsystem. Sie stellen umfangreiche Konfigurationsmöglichkeiten für die virtualisierte Hardware der Clients sowie das Netzwerk bereit.

[5] Pipeworks - https://github.com/jpetazzo/pipework

Kapitel 6

Umsetzung

Die Umsetzung eines Prototyps soll der Validierung des erstellten Konzepts dienen. Um das in dieser Arbeit beschriebene Konzept zu überprüfen, wurden die in Kapitel 4 beschriebenen Anwendungsszenarien, welche die Analyse und Manipulation der Netzwerkkommunikation im System ermöglichen, umgesetzt. Die Bereitstellung dieser Anwendungsszenarien erfordert aufgrund der in Abschnitt 5.4 beschriebenen, durch die Softwarewahl auftretenden Beschränkungen der Netzwerkkonfiguration, eine Anpassung der Architektur sowie die Erweiterung des bestehenden Systems um weitere Komponenten.

Die vorhandene Implementierung des OPC UA Protokolls der Industrie 4.0 Testumgebung wurde um die Bereitstellung von Sicherheitsprofilen im Secure Channel erweitert. Dadurch ist es möglich, die Netzwerkkommunikation der Komponenten an der Netzwerkbrücke des Docker Services bzgl. der in Abschnitt 3.1 beschriebenen Anforderungen zu analysieren.

Des weiteren wurde das Testsystem um eine Netzwerkumgebung mit weiteren Netzwerkteilnehmern in verschiedenen Netzen erweitert. Diese stellen die Dienste DHCP sowie DNS bereit. Um die Kommunikation zwischen den Netzwerkteilnehmern und dem vorhandenen Containernetz bereitzustellen, wurde ein Router für den Transport und die Wegfindung der Pakete zwischen den verschiedenen Netzen konfiguriert. Dieser Verbund ermöglicht die Umsetzung einer MitM Attacke mit Hilfe eines Rogue DHCP Servers im Netzwerk.

Um den Eingriff und die Manipulation der Netzwerkkommunikation zweier Komponenten zu untersuchen, wurde eine CoAP Client-Server Architektur implementiert. Über die beschriebene MitM Attacke ist es möglich Informationen über die

Kommunikation der implementierten CoAP Komponenten zu erlangen und diese zu verändern.

Die Anzahl der benötigten virtuellen Komponenten wird gering gehalten, indem die vorhandenen VMs mit mehreren Netzwerkinterfaces bereitgestellt werden. Dies ermöglicht die logische Trennung der Dienste auf einer VM in verschiedene Netzwerke. Die Dienste werden auf die speziellen Netzwerkinterfaces gebunden. Somit kann eine Simulation der Kommunikation über Gateways ermöglicht werden.

6.1 Softwarewahl

Das vorhandene Testsystem (Weber 2018) basiert auf einer Ubuntu Desktop 18.04 (**LTS!** (**LTS!**))[1] virtuellen Maschine und der Containervirtualisierung Docker. Beim für die Erweiterung genutzten Softwarestack wurde sich weitestgehend am Testsystem orientiert, um die Kompatibilität zu bestehenden Komponenten zu gewährleisten und weiterhin ein flexibles System bereitzustellen. Für die Analyse der Netzwerkkommunikation des genutzten Protokolls OPC UA wurden die benötigten Änderungen direkt am Quellcode der NodeJS Implementierung des Industrie 4.0 Testsystems vorgenommen. Weitere Komponenten wurden, aufgrund von Abschnitt 5.4, in weiteren virtuellen Maschinen umgesetzt, um das IPAM des Netzwerks selbst bereitstellen und manipulieren zu können. Um Lizenzkosten zu vermeiden, wurde als Basis für die zusätzlichen virtuellen Maschinen das Betriebssystem Ubuntu Server in der letzten verfügbaren **LTS!** Version genutzt. Die genutzten Softwarepakete und Bibliotheken sind Bestandteil des GNU-Projekts[2] und sollten somit analog in anderen Unix Betriebssystemen genutzt werden können. Aufgrund der weiten Verbreitung und umfangreichen Dokumentation wurde die Software *isc-dhcp-server*[3] und *bind9*[4] genutzt. Die Routingfunktionalitäten wurden über *iptables*[5] bereitgestellt. Als Hypervisor kommt weiterhin die Software „Oracle VM Virtual Box“ zum Einsatz, um die Komplexität des Systems gering zu halten.

[1] LTS - Long Term Support TODO
[2] text
[3] DHCP Server unso TODO
[4] DNS Server unso TODO
[5] iptables sau geil

6.2 Integration

Die Integration des Systems beschreibt die Bereitstellung und Konfiguration der System- und Netzwerkarchitektur. Um den Zeit- und Arbeitsaufwand für die Bereitstellung des Betriebssystems zu minimieren, wurde, um die zwei zusätzlichen virtuellen Maschinen bereitzustellen, ein Minimalsystem des Ubuntu Server 18.04 (**LTS!**) generisch installiert und als Vorlage genutzt. Hierbei wurde der Benutzer „i40“ mit dem Passwort „industrie40“ angelegt, welcher für alle Maschinen gilt.

Aus der Vorlage wurden die Klone „mgmt“ und „comp“ erstellt. Die VM „mgmt“ dient der Bereitstellung der im Testnetzwerk benötigten Dienste DHCP und DNS und dient als Router zwischen den verschiedenen internen Netzwerken sowie zum Host System. Auf der VM „comp“ wurde der CoAP Client umgesetzt. Die vorhandene VM „i40“ wird um einen Rogue DHCP Server erweitert.

Zuerst wurde die Netzwerkkonfiguration aller vorhandener virtueller Maschinen wie in **??** beschrieben angepasst. Dies konnte in den Einstellungen des Hypervisors durchgeführt werden. Die Netzwerkkonfiguration der VM „i40“ wurde von NAT auf das interne Netzwerk „i40-netzwerk“ geändert. Der VM „mgmt“ wurden drei Netzwerkadapter hinzugefügt. Der erste Adapter gehört dem internen Netzwerken „i40-netzwerk“ an, der zweite Adapter dem internen Netzwern „i40-monitoring“, der dritte Adapter dient dem Host-NAT. Die VM „comp“ besitzt nur einen Netzwerkadapter, welcher dem internen Netzwerk „i40-monitoring“ angehört.

Die Installation und Konfiguration aller Komponenten sowie deren Konfigurationsdateien werden in den jeweiligen Readme Dateien des Git Repositories[6] detailliert beschrieben.

6.2.1 Netzwerkverwaltung

Um die grundlegenden Netzwerkdienste in den internen Netzwerken bereitzustellen und die Kommunikation anderer Netzwerkteilnehmer mit externen Netzen wie dem Internet zur Installation und Konfiguration weiterer Software zu ermöglichen, wurde mit der Installation und Konfiguration der VM „mgmt“ begonnen.

Nach dem Start dem VM wurde der Hostname zur Namensauflösung auf dem System geändert. Anschließend wurde die Netzwerkkonfiguration aller Adapter durch-

[6] TODO - Mein Git

geführt und den Schnittstellen der internen Netzwerke die statischen IP Adressen 10.0.0.1 und 10.0.10.1 zugewiesen. Der Adapter zum Hostsystem bleibt unverändert. Auf der IP Adresse 10.0.0.1 werden für das Netzwerk „i40-network“ die Dienste DHCP und DNS konfiguriert. Für das Netzwerk „i40-monitoring“ wird kein DHCP konfiguriert, da nur zwei Komponenten im Netzwerk vorhanden sind und die Form der Adressvergabe in diesem Netzwerk keinen Einfluss auf die Umsetzung des Konzepts hat.

DNS/DHCP

Eine korrekte Namensauflösung in einem Netzwerk mit dynamischen IP Adressen kann nur ermöglicht werden, wenn die Dienste DNS und DHCP zusammenarbeiten. Die Authentifizierung zwischen DNS und DHCP zur automatischen Erstellung und Erneuerung von Zonen für Teilnehmer mit dynamischen Adressen findet mit Hilfe eines symmetrischen Schlüssels statt, welcher auf beiden Servern hinterlegt sein muss. Der Schlüssel wurde mit Hilfe des Tools *rndc-confgen* erstellt.

Anschließend wurde der DNS Server installiert und konfiguriert. Dabei wurden die Berechtigungen zum Anfragen des DNS Servers auf das gewünschte Netz 10.0.0.0/24 beschränkt, um die Nutzung des Servers als Open DNS[7] zu verhindern, die DNS Server von Google (8.8.8.8) und OpenDNS (208.67.220.220) als Forwarder genutzt, der erstellte Schlüssel eingebunden und das Interface des Dienstes auf das Netzwerk „i40-network“ beschränkt. Auf die Konfiguration des Sicherheitsmechanismus DNSSEC wurde aus zeitlichen Gründen verzichtet.

Um eine Namensauflösung im Netzwerk bereitzustellen, wurde eine *Forward-* sowie *Reverselookup*-Zone für die Suchdomain „i40-network.lan“ erstellt und die statischen IP Adressen der VMs „i40“ und „mgmt“ hinterlegt. Nach dem Neustart des Dienstes wurde mit Hilfe der Software *dig* die korrekte Funktionalität der Namensauflösung im Netzwerk getestet.

Danach wurde der DHCP Server installiert. In der Konfiguration des Servers wurde ebenfalls der erstelle Schlüssel zur Aktualisierung des DNS Zonen hinterlegt. Es wurde das Feature „ddns-updates“ aktiviert, den Dienst auf das Interface der Netzwerks „i40-network“ beschränkt und die zu Verteilenden DHCP Informationen wie DHCP Range, Gateway und Nameserver definiert. Die DHCP Range des Servers

[7] OPENDNS TODO WAS DES

wurde auf die Adressen 10.0.0.2 - 10.0.0.100 beschränkt, um im späteren Verlauf einen Wechsel des DHCP Servers auf den anderen Systemen durch einen Wechsel der IP Adresse besser verdeutlichen zu können.

Die Validierung der Funktionalität des DHCP Servers wird durch die Konfiguration des DHCP Clients der VM „comp“ durchgeführt.

Routing

Die Umsetzung der Routingfunktionalität benötigte keine zusätzliche Installation von Software. Sie wurde durch IP Forwarding und NAT mit Hilfe von *iptables* umgesetzt. Das IP Forwarding muss auf Betriebssystembasis durch Änderung einer Konfiguration[8] aktiviert werden. Anschließend konnten die folgenden Regeln definiert werden, um die Pakete bei Verbindungen, welche von den internen Netzen hergestellt wurden, weiterzuleiten.

```
iptables -A FORWARD -i enp0s8 -o enp0s9 -m state --state RELATED,ESTABLISHED -j
    ACCEPT
iptables -A FORWARD -i enp0s9 -o enp0s8 -j ACCEPT
iptables -A FORWARD -i enp0s3 -o enp0s8 -m state --state RELATED,ESTABLISHED -j
    ACCEPT
iptables -A FORWARD -i enp0s8 -o enp0s3 -j ACCEPT
```

Listing 6.1: Iptables

Das Interface „enp0s3“ beschreibt die Verbindung zum Hostsystem, der Adapter „enp0s8“ ist Teil des Netzwerks „i40-network“, der Adapter „enp0s9“ stellt die Schnittstelle des Routers im Netzwerk „i40-monitoring“ dar.

Um das Regelwerk bei jedem Neustart bereitzustellen wird es in einer Datei[9] gespeichert und durch das Softwarepaket *iptables-persistent* bei jedem Systemstart wiederhergestellt.

Delay

Ein weiterer Bestandteil des Netzwerkstacks des Linux Kernels stellt das Tool *tc*[10] dar. Hiermit wurde im Netzwerkinterface des DHCP Servers ein Delay von 500ms

[8] IPv4 Forwarding : /etc/sysctl.conf
[9] iptables Konfiguration : /etc/iptables/rules.v4
[10] tc : traffic control

simuliert. Dies ermöglicht im weiteren Verlauf, dass die Netzwerkpakete des Rogue DHCP Servers früher beim anzugreifenden Client eintreffen, als die Pakete des zuständigen Servers.

6.2.2 CoAP Server

Das Monitoringsystem der CoAP Komponente wurde auf der VM „comp“ umgesetzt. Nach dem Start der Maschine wurde auch hier der Hostname angepasst und die Netzwerkkonfiguration auf DHCP gesetzt. Dies stellte eine optimale Gelegenheit zum testen des bereitgestellten DNS und DHCP Servers sowie der Routingfunktionalitäten dar.

Die Konfigurationsparameter des DHCP Servers, welche aus IP Adresse, DNS und Gateway bestehen, konnten auf dem Client mit Hilfe der Tools *ip*[11] und *systemd-resolv*[12] überprüft werden und somit die korrekte Funktionalität des DHCP Servers nachgewiesen werden. Die Funktionalität des DNS Servers im Netzwerk konnte auf diesem Server mit dem Tool *dig* nachgewiesen werden, indem die RR der Management-VM sowie die RR der dynamisch bezogenen Komponenten-VM abgefragt und überprüft wurden.

Zur späteren Ausführung des implementierten Überwachungssystems musste auf diesem System ein Webserver bereitgestellt werden. Dies geschieht durch das Paket *nodejs*, welches auf dem System installiert wurde.

6.3 Implementierung

Die Implementierung umfasst die Erweiterung und Anpassung des vorhandenen Systems sowie die Bereitstellung eines neuen Dienstes. Die vorhandene Implementierung über das Protokoll OPC UA wird angepasst, um die Kommunikation über den *Secure Channel* mit Hilfe verschiedener Sicherheitsprofile zu analysieren. Hierbei wird eine unverschlüsselte sowie verschlüsselte Kommunikation zwischen OPC UA Client und Server bereitgestellt.

Das Testsystem wurde um ein zusätzliches Monitoringsystem erweitert, welches eine CoAP Komponente überwacht. Die Ausführung dieser Architektur dient der

[11] ip a undso
[12] TODO - des auch

Analyse eines weiteren IIoT Protokolls in Bezug auf die Manipulation von Netzwerktraffic und Verdeutlichung der Auswirkungen dieser Bedrohung.

6.3.1 OPC UA Secure Channel

Um die Verwendung verschiedener Sicherheitsrichtlinien im OPC UA *Secure Channel* bereitzustellen muss die Form des Verbindungsaufbaus der vorhandenen OPC UA Clients im Quellcode geändert werden. Die OPC UA Server des Testsystems stellen die verschiedenen Sicherheitsprofile *None*, *Basic128Rsa15*, *Basic256* und *Basic256Sha256* bereit. Diese beinhalten den für die Nachrichtenübermittlung genutzten Verschlüsselungsalgorithmus. Bei der Übertragung der Daten im *Secure Channel* wird der OPC UA *MessageSecurityMode* auf das Sicherheitsprofil angewandt. Hierbei stehen die Optionen NONE, SIGN und SIGNANDENCRYPT zur Verfügung. Im Vorhandenen Testsystem werden keine Zertifikate verwaltet. Das Signieren der Nachrichten mit dem privaten Schlüssel des Absenders ermöglicht einen Zuwachs der Sicherheit der Netzwerkkommunikation, da dies die Integrität der Nachrichten sicherstellt. Da im Testsystem keine Verwaltung der Zertifikate stattfindet, ist kein Signieren der Nachrichten möglich. Im gegebenen System wurde jedoch die Verschlüsselung der Nachricht auf Anwendungsebene durch den Algorithmus *Basic256Sha256* implementiert.

Der Quellcode der OPC UA Clients in den Containern *scheduler* und *control* wurde so angepasst, dass eine Aktivierung und Deaktivierung der Verschlüsselung in der Konfigurationsdatei *config.json* der jeweiligen Server vorgenommen werden kann. Zur Anwendung einer Konfigurationsänderung ist ein erneutes Bauen sowie der Neustart des Containers notwendig. Die Installation und Konfiguration der Anpassungen am bestehenden Testsystem werden in den entsprechenden Readme Dateien des Repositories[13] beschrieben und anhand von Scripten unterstützt.

6.3.2 CoAP Monitoringsystem

CoAP Client und Server dienen der Repräsentation eines MitM Angriffsszenarios. Beide Komponenten wurden mit minimaler Funktionalität implementiert. Für die Implementierung dieses Prototyps bietet sich der NodeJs Stack an, welcher mit

[13] TODO - siehe Repo Readme

node-coap[14] eine Bibliothek für das CoAP Netzwerkprotokoll bereitstellt und wenig Ressourcen benötigt.

Der CoAP Client besteht aus einem Server, welcher einen Temperatursensor einer Gießmaschine simuliert. Die Temperatur wird alle fünf Sekunden zum zuständigen Überwachungssystem übermittelt. Die Übertragung erfolgt unverschlüsselt über das Transportprotokoll UDP, welches es möglich macht die Kommunikation im Netzwerk auch ohne Server analysieren zu können. Der Client sendet alle fünf Sekunden die Temperatur eines Sensors einer Gießmaschine zu einer in der Datei *config.json* konfigurierten Uniform Resource Locator (URL). Der CoAP Client wurde auf der VM „comp“ bereitgestellt.

Der CoAP Server stellt den Empfänger der Nachrichten dar. Er stellt ein Webinterface bereit, um die empfangenen Daten zu visualisieren. Aufgrund der Simplizität der bereitgestellten Funktionalitäten wurde zur Implementierung des Systems ausschließlich ein *Pen-and-Paper-Protoyping* durchgeführt. Der CoAP Server bietet in der Datei *config.json* die Möglichkeit die IP Adresse, auf welche der Socket des Webservers gebunden wird zu bestimmen. Da das System auf der VM „mgmt“ ausgeführt wird und um dem Server im Testsystem das Netzwerk „i40-monitoring“ zuzuweisen, wurde die IP Adresse 10.0.10.1 in der Konfiguration gesetzt. Das Graphical User Interface (GUI) des Webinterface wird in **??** dargestellt.

TODO - Screenshot GUI

6.3.3 CoAP Manipulationssystem

TODO - Implementierung u. Beschreibung

6.4 Quellcode

Der erstellte Quellcode ist im öffentlichen GitHub Repository[15] unter der Massachusetts Institute of Technology (MIT) Lizenz verfügbar. Das Lizenzierungsmodell erlaubt die freie Nutzung und Änderung der Software durch Dritte.

[14] TODO : Github - node-coap
[15] https://github.com/fjnalta/i40-testbed

Der gesamte Quellcode befindet sich im Verzeichnis „src“ des GitHub Respositories https://github.com/fjnalta/thesis. Die Ordner „CoAP_Client“ und „CoAP_Server“ repräsentieren die Implementierungen der CoAP Komponenten und deren Monitoringsystem. Im Ordner „scripts“ werden Scripte zur Installation und Konfiguration des Testsystems bereitgestellt. Das vorhandene, aktualisierte Testsystem (Weber 2018) ist im Ordner „i40-testsystem“ eingebettet.

6.5 Dokumentation

Die Dokumentation der implementierten Komponenten, deren Inbetriebnahme und Funktionsweise findet neben der schriftlichen Ausarbeitung in den jeweiligen *Readme* Dateien des Repositories https://github.com/fjnalta/thesis im Verzeichnis „testbed“ und dessen Unterverzeichnissen statt. Des Weiteren wurde der Quellcode der Software mit Kommentaren versehen, um eine Nutzung des Testsystems auch ohne die schriftliche Ausarbeitung zu ermöglichen.

Kapitel 7

Validierung

Mit der Implementierung und Durchführung der in Kapitel 4 beschriebenen Szenarien wird das in Kapitel 5 erstellte Konzept validiert.

TODO

Die folgenden Abbildungen zeigen die Ergebnisse der Paketanalyse mit der vorhandenen *Security Policies* „none“. Der OPC UA Client des Containers „control“, welcher die Liste der im Netzwerk vorhandenen OPC UA Server abfrägt besitzt die IP-Adresse 172.18.0.6, der Container des Discovery Servers die IP-Adresse 172.18.0.2. In Abbildung 7.1 ist der Request des Control Containers zum Aufbau eines *Secure Channel* dargestellt. Die verwendete *Security Policy* ist im Bereich „SecurityPolicyUri“ des OPC UA Protokolls beschrieben. In Abbildung 7.2 ist die Antwort des OPC UA Discovery Servers im *Secure Channel* dargestellt.

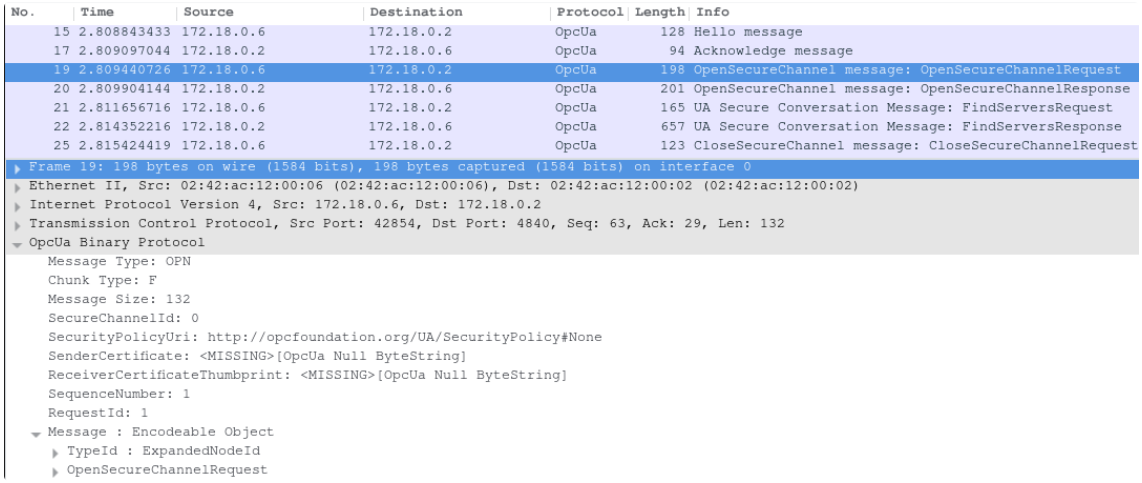
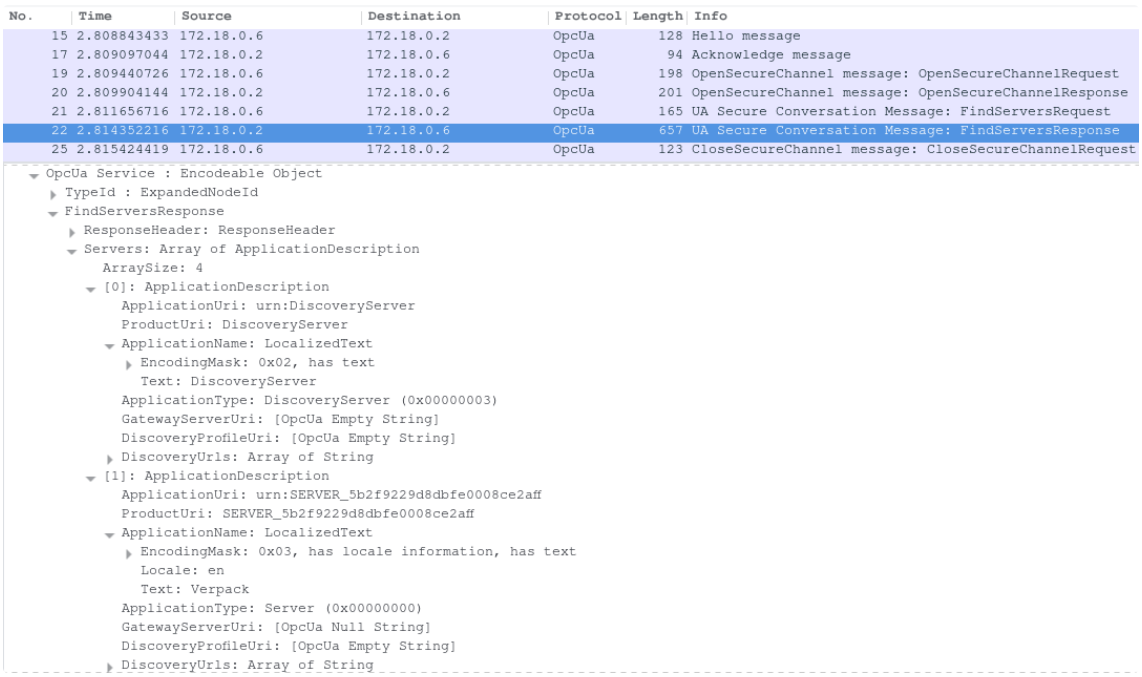


Abbildung 7.1: Paketanalyse OPC UA - Client Request bei Sicherheitsprofil "none"



No.	Time	Source	Destination	Protocol	Length	Info
15	2.808843433	172.18.0.6	172.18.0.2	OpcUa	128	Hello message
17	2.809097044	172.18.0.2	172.18.0.6	OpcUa	94	Acknowledge message
19	2.809440726	172.18.0.6	172.18.0.2	OpcUa	198	OpenSecureChannel message: OpenSecureChannelRequest
20	2.809904144	172.18.0.2	172.18.0.6	OpcUa	201	OpenSecureChannel message: OpenSecureChannelResponse
21	2.811656716	172.18.0.6	172.18.0.2	OpcUa	165	UA Secure Conversation Message: FindServersRequest
22	2.814352216	172.18.0.2	172.18.0.6	OpcUa	657	UA Secure Conversation Message: FindServersResponse
25	2.815424419	172.18.0.6	172.18.0.2	OpcUa	123	CloseSecureChannel message: CloseSecureChannelRequest

Abbildung 7.2: Paketanalyse OPC UA - Server Response bei Sicherheitsprofil "none"

Es ist zu erkennen, dass die Kommunikation, obwohl der *Secure Channel* genutzt wird, nicht verschlüsselt ist. Die Endpunkte sowie deren Adressen und bereitgestellte Methoden können aus den Paketen ausgelesen werden.

Im Folgenden wird erneut das Abfragen des Control Containers aller im Netzwerk vorhandenen Endpunkte beim Discoveryserver beschrieben, jedoch wird das Sicherheitsprofil "Basic256Sha256"mit dem *MessageSecurityMode* SSIGNANDENCRYPT"genutzt. Der OPC UA Client besitzt die IP-Adresse 172.18.0.7. Der Discoveryserver weiterhin die Adresse 172.18.0.2. Abbildung 7.3 zeigt den Request, Abbildung 7.4 die verschlüsselte Response im *Secure Channel*.

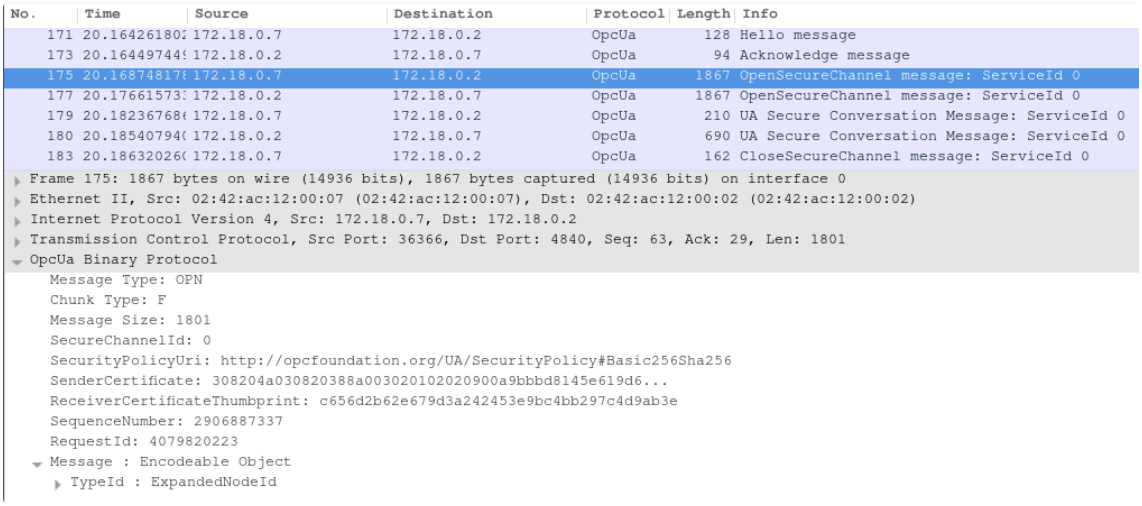


Abbildung 7.3: Paketanalyse OPC UA - Client Request bei Sicherheitsprofil "Basic256Sha256ünd MessageSecurityMode SSIGNANDENCRYPT"

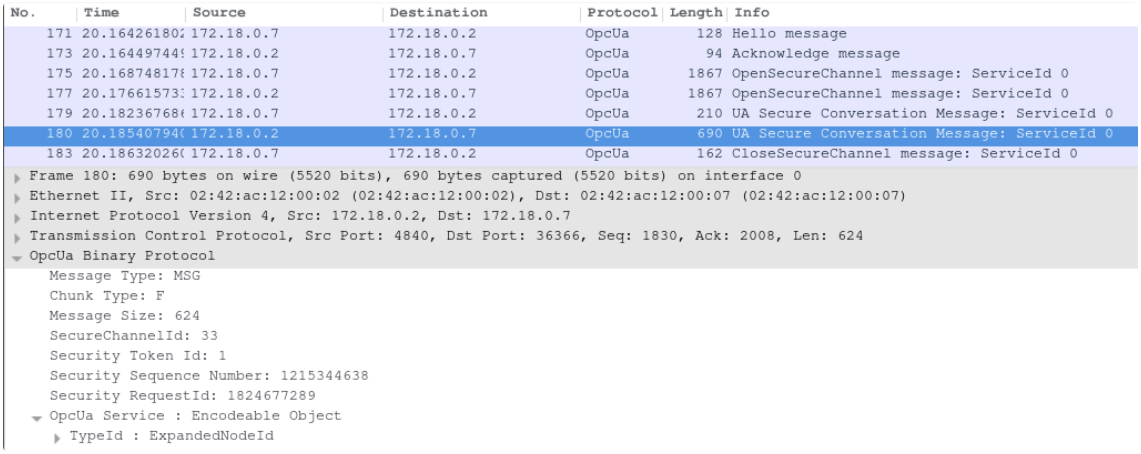


Abbildung 7.4: Paketanalyse OPCUA - Server Response bei Sicherheitsprofil "Basic256Sha256ünd MessageSecurityMode SSIGNANDENCRYPT"

Es ist zu erkennen, dass der gesamte Netzwerkverkehr im *Secure Channel* durch den Algorithmus SHA256 verschlüsselt wurde.

Kapitel 8

Fazit

Da die grundlegende Netzwerkstruktur der TCP/IP Netzwerke für Industrie 4.0 Kommunikation übernommen wird, sind auch die damit zusammenhängenden Voraussetzungen und Sicherheitsgedanken zu beachten. Plattform Industrie 4.0 2017

Vielzahl von Angriffen auf die verschiedenen Schichten des TCP/IP Referenzmodells, welches in Industrie 4.0 Umgebungen genutzt wird. In der Thesis wurden nur wenige Beispielhafte Angriffe dargestellt und durchgeführt.

Analyse der Netzwerkkommunikation in Industrie 4.0 Umgebungen und Erweiterung einer protoypischen Security Testumgebung zur Darstellung verschiedener Berohungsfaktoren

Security Mechanismen sind nicht umsonst und beeinträchtigen die Performance. Security sollte daher nur dort zur Anwendung kommen, wo sie auch benötigt wird. Diese Entscheidung soll aber nicht der Entwickler / Produktmanager treffen, sondern der Anlagenbetreiber (Systemadministration).

Die Nutzung von OPC UA bietet keinen automatischen Schutz der IT-Infrastruktur - Konfiguration ist notwendig. Obwohl Securiy by Design. Basiert auf Unteren Schichten.

TODO - Applikationssicherheit != Netzwerksicherheit != Betriebssystemsicherheit
TODO - Schutz auf allen Ebenen -> z.B. OPC UA basiert auf IP-Netz -> Angriffsvektoren von IP und genutzten Diensten immer noch zutreffend

Die Referenzarchitekturen RAMI4.0 und IIRA beschreiben den logischen Aufbau von Systemen und deren Prozessen in einem IIoT Umfeld. Der Fokus der Referenzarchitekturen liegt auf der Definition von Abstraktionsebenen und semantischen Zu-

sammenhängen. Bei dieser Betrachtung wird die Abbildung der logischen Architekturen auf verteilte Informations- und Kommunikationssysteme nicht beachtet.

Kapitel 9

Ausblick

TODO - LDAP - Firewall - CoAP Auth - SSL - OPC UA System Erweitern - Zertifikatsmanagement, PKI, IdM - weitere Angriffsmethoden - DNS Amplification, SYN-Flood, ARP Spoofing, usw. Ipv6 QoS TSN

TODO - das hier ist referenziert im Konzept! Durch die eigene Verwaltung des DNS Servers ist es in Zukunft möglich weitere Anwendungsszenarien am Testsystem darzustellen. Dazu gehören u. A. der in Abschnitt 3.6.1 beschriebene Angriff der DNS Amplification durch die Generierung eigener Zonen mit extrem vielen RR, um eine möglichst große DNS Response zu provozieren sowie das DNS Spoofing und die Analyse der Sicherheitsmechanismen von DNSSEC.

9.0.1 Defense in Depth

Auf der Netzzugangsschicht fallen, wie auf allen anderen Schichten, Betriebsdaten an, welche genutzt werden können, um Angriffe oder unregelmäßige Aktivitäten im Netzwerk zu erkennen. Es kann protokolliert werden, wann ein Gerät mit den Netzwerk verbunden war und welche Pakete andere Netzwerkteilnehmer von diesem Gerät erhalten haben (Plattform Industrie 4.0 2017). Die Norm IEC 62443[1] definiert die Defense in Depth Strategie. Sie stellt ein Konzept bereit, um die IT-Sicherheit der Anlagen, die Netzwerksicherheit und Systemintegrität nach dem Stand der Technik zu schützen. Sie gliedert eine Unternehmensinfrastruktur in multiple und redundante Sicherheitsschichten (Zonen), um ein höchstmögliches Sicherheitsniveau zu erreichen. Die unabhängigen Verteidigungslinien sollen An-

[1] ref. IEC 62443

griffe verzögern, um Zeit für Gegenmaßnahmen zu gewinnen. Die Kommunikation erfolgt in separierten Netzsegmenten, welche zusätzlich mit IDS nutzen, um Angriffe schnell zu erfassen und Gegenmaßnahmen einleiten zu können. Somit wird der Aufwand, um die unteren Netzwerkebenen zu kompromittieren durch den Einsatz von Demilitarized Zone (DMZ), IDS, Paketfilter und Time Access Control wesentlich erhöht. Zusätzlich ist das „Zone and Conduit“ Modell eines der zentralen Elemente der Defense in Depth Strategie. Die verschiedenen Zonen können nur mittels spezieller Leitungen (Conduits) miteinander kommunizieren.

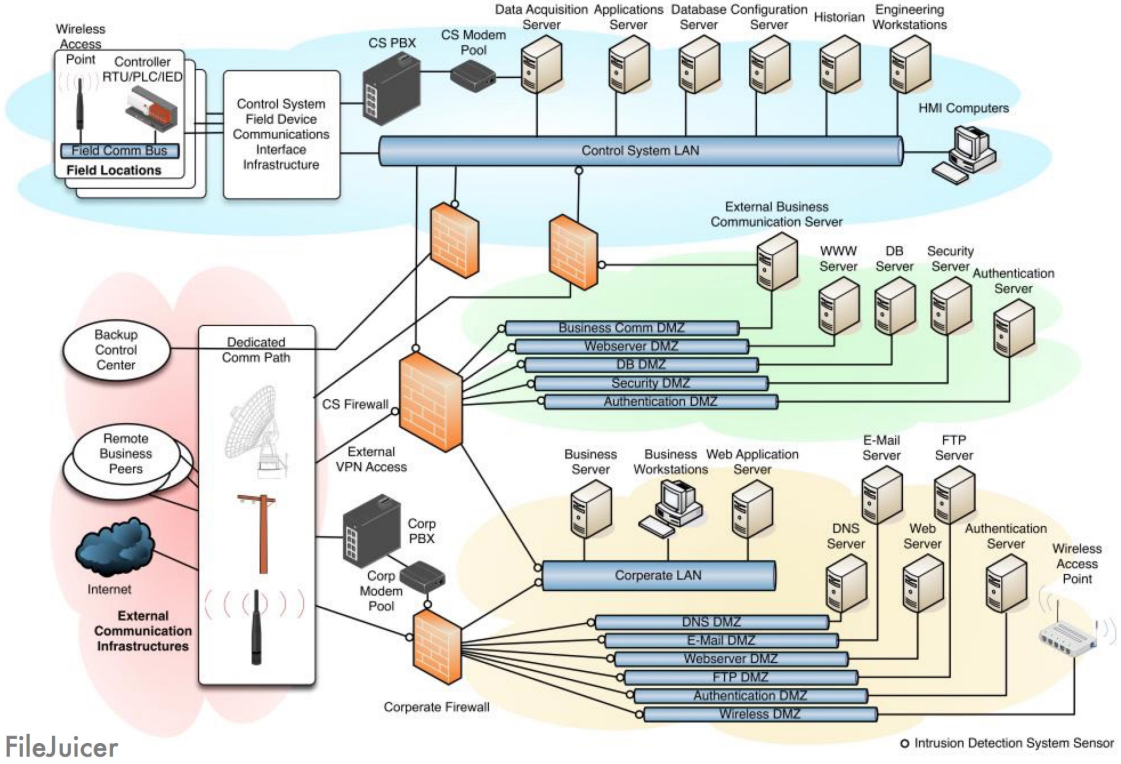


Abbildung 9.1: Defense in Depth Strategie - David Kuipers 2006

Das Defense in Depth Konzept stellt ein Konzept dar, um Industrieanlagen und Unternehmensnetzwerke vor Angriffen zu schützen. Bei der sich ständig ändernden Bedrohungslage in den komplexen Netzen wird bei dieser Strategie jedoch weniger ein vollständiger Schutz bereitgestellt, als eine Strategie zur Schadensbegrenzung im Falle eines Angriffs.

Abkürzungsverzeichnis

IEEE	Institute of Electrical and Electronics Engineers
KRITIS	Kritische Infrastrukturen
IPC	Industrie PC
SPS	speicherprogrammierbare Steuerungen
SCADA	Supervisory Control and Data Acquisition
ERP	Enterprise Resource Planning
MES	Manufacturing Execution System
RAMI4.0	Referenzarchitekturmodell Industrie 4.0
IIRA	Industrial Internet Reference Architetecture
IIAF	Industrial Internet Architecture Framework
IIC	Industrial Internet Consortium
IoT	Internet of Things
IIoT	Industrial Internet of Things
CPS	Cyber-physisches System
OPC UA	Open Platform Communications Unified Architecture
M2M	Machine to Machine
QoS	Quality of Service
ICS	Industrial Control System
REST	Representational State Transfer
IETF	Internet Engineering Task Force
MAN	Metropolitan Area Network
WAN	Wide Area Network
GAN	Global Area Network
OPC COM	Open Platform Communications
IIS	Industrial Internet Systems

DA	Data Access
A&E	Alarms and Events
HDA	Historical Data Access
IP	Internet Protocol
TCP	Transmission Control Protocol
DNS	Domain Name System
UDP	User Datagram Protocol
SOA	Service Oriented Architecture
OMG	Open Management Group
DDS	Data Distribution Services
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
SOAP	Simple Object Access Protocol
CoAP	Constrained Application Protocol
XMPP	Extensible Messaging and Presence Protocol
MQTT	Message Queue Telemetry Transport
AMQP	Advanced Message Queuing Protocol
VM	virtuelle Maschine
PKI	Public-Key Infrastructure
BSI	Bundesamt für Sicherheit in der Informationstechnik
DoS	Denial of Service
DDoS	Distributed Denial of Service
DMZ	Demilitarized Zone
IDS	Instrusion Detection System
IEC	International Electrotechnical Commission
ARP	Address Resolution Protocol
DHCP	Dynamic Host Configuration Protocol
RFC	Request for Comments
IETF	Internet Engineering Task Force
RR	Resource Record
BAF	Base Amplification Factor
DRDoS	Destributed-Reflected-Denial-of-Service

VLAN Virtual Local Area Network
IPAM IP Adress Management
SYN synchronise
ACK acknowledge
SYN-ACK synchronise-acknowledge
FIN final
VoIP Voice over IP
SNMP Simple Network Management Protocol
TLS Transport Layer Security
DTLS Datagram Transport Layer Security
SMTP Simple Mail Transfer Protocol
MOM Message oriented Middleware
ACL Access Control List
URL Uniform Resource Locator
URI Uniform Resource Identifier
UA Unified Architecture
XML Extensible Markup Language
CA Certificate Authority
RA Registration Authority
VA Validation Authority
UACP OPC UA Connection Protocol
IPsec Internet Protocol Security
USB Universal Serial Bus
AH Authentication Header
ESP Encapsulating Security Payload
VPN Virtual Private Network
SSL Secure Sockets Layer
TSIG Transaction Signature
WS Webservice
MitM Man in the Middle
MIT Massachusetts Institute of Technology
GUI Graphical User Interface

UML Unified Modeling Language

NAT Network Address Translation

DNSSEC Domain Name System Security Extensions

Tabellenverzeichnis

Abbildungsverzeichnis

2.1 Kommunikationsbeziehungen in einer Industrie 4.0 Umgebung . . . 7
2.2 Automatisierungspyramide - TODO ref. Langmann,2004 8
2.3 Das Internet der Dinge 9
2.4 RAMI 4.0 12
2.5 Die Grundebenen der IIRA 14
2.6 OPC UA Multi-Part Specification - OPC Foundation 2018a 18
2.7 OPC UA Client-Server Architektur - OPC Foundation 2018a 19

3.1 ARP Paketformat 34
3.2 TCP Verbindungsaufbau 39
3.3 Sockstress Sequenzdiagramm 42
3.4 Wireshark - ID im DNS Header 45
3.5 Schematisches Beispiel: DNS Amplification 47
3.6 DNS Amplification am Beispiel von isc.org 49
3.7 Netzlast bei DNS Amplification 50
3.8 OPC UA Security Architecture 53
3.9 OPC UA Kommunikationswege 55
3.10 CoAP Message Format 58

5.1 Netzwerkinfrastruktur 65
5.2 Virtuelle Maschinen und Dienste 68
5.3 Routerkomponente 69
5.4 DHCP und DNS Server 70
5.5 CoAP Client 71
5.6 CoAP Server 71
5.7 Rogue DHCP Server 72
5.8 CoAP Manipulationssystem 73
5.9 Dynamische Hostkonfiguration mit DHCP und DNS 75
5.10 Rekursive DNS Namensauflösung 76
5.11 Iterative DNS Namensauflösung 76
5.12 Initialisierung von CoAP Client und Kommunikation zwischen CoAP Client und CoAP Server 77
5.13 Umleitung der Netzwerkkommunikation durch Manipulation des Gateways 78

7.1 Paketanalyse OPC UA - Client Request bei Sicherheitsprofil "none" 91
7.2 Paketanalyse OPC UA - Server Response bei Sicherheitsprofil "none" 92
7.3 Paketanalyse OPC UA - Client Request bei Sicherheitsprofil "Basic256Sha256ünd MessageSecurityMode SSIGNANDENCRYPT" . 93
7.4 Paketanalyse OPCUA - Server Response bei Sicherheitsprofil "Basic256Sha256ünd MessageSecurityMode SSIGNANDENCRYPT" . 93

9.1 Defense in Depth Strategie - David Kuipers 2006 98

Listings

6.1 Iptables 85

Literatur

Bundesamt für Sicherheit in der Informationstechnik, BSI (2016). „Industrial Control System Security“. In:

Bundesministerium für Wirtschaft und Energie, BMWi (2016a). „Netzkommunikation für Industrie 4.0“. In: *Platform Industrie 4.0*.

— (2016b). „Technischer Überblick: Sichere unternehmensübergreifende Kommunikation“. In:

Burke, Manfred (2013). *Rechnernetze*. Springer.

Christoph Meinel, Harald Sack (2011). *Internetworking - Technische Grundlagen und Anwendung*. Springer.

David Kuipers, Mark Fabro (2006). „Control Systems Cyber Security“. In:

Dieter Spath, Oliver Ganschar, Stefan Gerlach, Moritz Hämmerle, Tobias Krause, Sebastian Schlund (2013). „Produktionsarbeit der Zukunft - Industrie 4.0“. In: *Frauenhofer Institut für Arbeitswirtschaft und Organisation IAO*.

DIN SPEC (2016). *DIN SPEC 91345:2016-04: Referenzarchitekturmodell Industrie 4.0 (RAMI4.0)*.

Dr.-Ing. Mike Heidrich, Dr. Jesse Jijun Lui (2016). „Industrial Internet of Things: Referenzarchitektur für die Kommunikation“. In: *Fraunhofer Institut für Eingebettete Systeme und Kommunikationstechnik ESK*.

Drath, Rainer (2014). „Industrie 4.0 - eine Einführung“. In: *openautomation.de*. URL: https://www.openautomation.de/fileadmin/user_upload/Stories/Bilder/oa_2014/oa_3/oa_3_14_ABB.pdf.

DTAG, Deutsche Telekom AG (2016). „Sicherheit im Industriellen Internet der Dinge“. In:

Hoppe, Stefan (2018). „OPC Foundation announces OPC UA PubSub release as important extension of OPC UA communication platform“. In: URL: https://opcfoundation.org/news/press-releases/opc-foundation-announces-opc-ua-pubsub-release-important-extension-opc-ua-communication-platform/.

Industrial Internet Consortium, IIC (2017a). „The Industrial Internet of Things - Volume G1: Reference Architecture“. In:

— (2017b). „The Industrial Internet of Things - Volume G4: Security Framework“. In: URL: https://www.iiconsortium.org/pdf/IIC_PUB_G4_V1.00_PB.pdf.

Lass Sander, Kotarski David (2014). „IT-Sicherheit als besondere Herausforderung von Industrie 4.0“. In: *Kersten W, Koller H, Lödding, H (ed) Industrie 4.0: Wie intelligente Vernetzung und kognitive Systeme unsere Arbeit verändern*.

Ledermüller, Thomas (2009). „DNS-Sicherheit im Rahmen eines IT-Grundschutz-Bausteins“. In:

Network Working Group (1981). *Transmission Control Protocol*. URL: https://tools.ietf.org/html/rfc793.

Olsen, Camilla (2003). „Security issues relating to the use of UDP“. In: *Global Information Assurance Certification Paper*.

OPC Foundation (2014). „OPC Unified Architecture - Wegbereiter der 4. industriellen (R)Evolution“. In:

— (2018a). „OPC Unified Architecture Specification Part 1: Overview and Concepts“. In: URL: https://opcfoundation.org/UA/Part1/.

— (2018b). „OPC Unified Architecture Specification Part 2: Security Model“. In: URL: https://opcfoundation.org/UA/Part2/.

— (2018c). „OPC Unified Architecture Specification Part 3: Address Space Model“. In: URL: https://opcfoundation.org/UA/Part3/.

— (2018d). „OPC Unified Architecture Specification Part 5: Information Model“. In: URL: https://opcfoundation.org/UA/Part5/.

— (2018e). „OPC Unified Architecture Specification Part 7: Profiles“. In: URL: https://opcfoundation.org/UA/Part7/.

Platform Industrie 4.0 (2015). „Umsetzungsstrategrie Industrie 4.0“. In:

Plattform Industrie 4.0 (2016). „Reference Architectural Model Industrie 4.0 (RAMI 4.0): An Introduction“. In: *Publikationen der Plattform Industrie 4.0*.

URL: https://www.plattform-i40.de/I40
/Redaktion/DE/Downloads/Publikation/rami40
-eine-einfuehrung.pdf?__blob=publicationFile&v=9.

— (2017). „Sichere Kommunikation für Industrie 4.0“. In: *Publikationen der Plattform Industrie 4.0*. URL: https://www.plattform-i40.de/I40
/Redaktion/DE/Downloads/Publikation/sichere-kommunikation-i40
.pdf?__blob=publicationFile&v=6.

Schleupner, Linus (2016). *Sichere Kommunikation im Umfeld von Industrie 4.0*. Springer.

Torscht, Dipl.-Ing. Robert (2014). „Kommunikation bei Industrie 4.0“. In: *SPS-Magazin, Fachzeitschrift für Automatisierungstechnik*.

Trapickin, Roman (2013). „Constrained Application Protocol (CoAP): Einführung und Überblick“. In:

W.A. Halang, H. Unger (Hrsg.) (2016). *Internet der Dinge*. Springer.

Weber, Martin (2018). „Ein Konzept für ein virtuelles Security Testbed für eine Industrie 4.0 Umgebung mit prototypischer Implementierung“. In: