Industrielle Gemeinschaftsforschung

Schlussbericht

zu IGF-Vorhaben Nr. 19117 N

Thema

Entwicklung einer IT -Sicherheitsinfrastruktur für verteilte Automatisierungssysteme

Berichtszeitraum

01.09.2016 bis 31.08.2019

Forschungsvereinigung

Forschungsvereinigung Elektrotechnik beim ZVEI e. V. (FE)

Forschungseinrichtungen

FE 1: Hochschule Hannover
Ricklinger Stadtweg 120
30459 Hannover
FE 2: Technische Hochschule Ostwestfalen-Lippe
Campusallee 6
32657 Lemgo

Hannover, den 23.12.2019 — Prof. Dr. Karl-Heinz Niemann

Lemgo, den 23.12.2019 — Prof. Dr. Stefan Heiss

Forschungsnetzwerk Mittelstand

Bundesministerium für Wirtschaft und Energie

aufgrund eines Beschlusses des Deutschen Bundestages

Autoren:

Tebbje, Sebastian	Hochschule Hannover
Karthikeyan, Gajasri	Technische Hochschule-Ostwestfalen Lippe
Friesen, Maxim	Technische Hochschule-Ostwestfalen Lippe
Steinke, Kai	Hochschule Hannover
Heiss, Stefan (Projektleiter)	Technische Hochschule-Ostwestfalen Lippe
Niemann, Karl-Heinz (Projektleiter)	Hochschule Hannover

Inhalt

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

1.1 Aufgabenstellung

Die zunehmenden Anwendungsfälle der vertikalen und horizontalen Vernetzung von Automatisierungssystemen erhöhen gleichzeitig auch die Bedrohungen der IT-Sicherheit der relevanten automatisierten technischen Prozesse. Zukünftige Anlagenstrukturen werden stärker vernetzt und dezentralisiert organisiert sein, oder sogar weltweit mit anderen technischen Systemen über das Internet kommunizieren. Dies erfolgt häufig über standardisierte Kommunikationsprotokolle, im Weiteren Middleware genannt. Daher ist die Bedeutung standardisierter Verfahren und Modelle zur Erleichterung der Sicherheitskonfigurationen der Middleware, die die folgenden Anforderungen erfüllen, von wesentlicher Bedeutung:

- Sichere Kommunikation (sichere Middleware)
- Authentifizierung und Autorisierung der Kommunikationspartner auch bei stark vernetzten Systemen und Ad-hoc-Verbindungen
- Einfache Verwaltung von Sicherheitsmaßnahmen ohne erheblichen Mehraufwand für die Organisation

Die bestehende Referenzarchitektur und die Sicherheitsmerkmale der entsprechenden Middleware werden untersucht, um deren Anwendbarkeit in verschiedenen Stufen der Automatisierungspyramide zu ermitteln. Es werden die Sicherheitsmerkmale für eine durchgängig sichere Kommunikation und deren Flexibilität bei der Integration in eine Public Key Infrastruktur (PKI) untersucht. Die bestehenden Zugangskontrollmechanismen und deren Zukunft im Rahmen von Industrie 4.0 werden bewertet. Die Möglichkeiten zur Integration des mit Attributzertifikaten aktivierten Berechtigungsmechanismus im Rahmen von OPC UA werden in diesem Projekt untersucht und als Teil des Demonstrators implementiert. Die Anwendungsrelevanz der entwickelten Lösungen wird gewährleistet, indem die Projektpartner des projektbegleitenden Ausschusses bei der Erstellung und dem Review von Konzepten frühzeitig beteiligt werden.

1.2 Wissenschaftlich-technischer Stand zu Beginn des Vorhabens

Mit dem zunehmenden Einsatz standardisierter Ethernet-basierter Kommunikationsprotokolle, wie z.B. PROFINET oder Ethernet IP, und einer weiter zunehmenden vertikalen und horizontalen Vernetzung von Automatisierungssystemen sind Bedrohungen hinsichtlich der IT-Sicherheit auch für automatisierte technische Prozesse relevant. Um die IT-Sicherheit in heutigen Anlagen zu gewährleisten, werden organisatorische Maßnahmen [1], [2], [3], [4], [5] und auch technische Maßnahmen auf Netzwerk- und Geräteebene empfohlen [4], [6], [7], [8]. Auf Netzwerkebene werden heute insbesondere sogenannte Security-Appliances mit Firewall- und VPN-Funktionalität eingesetzt, um den Zugriff auf das Automatisierungsnetzwerk oder auch nur auf Teile des Netzes gezielt einzuschränken (siehe z.B. [9], [10]). Der Einsatz und die Konfiguration dieser Security-Appliances sind jedoch aufwändig und setzen herstellerspezifisches Detailwissen voraus. Zukünftige Anlagenstrukturen werden stärker vermascht und dezentral organisiert sein, oder sogar weltweit mit anderen technischen Systemen über das Internet miteinander kommunizieren (Industrie 4.0 [11], Cyber-Physical Systems (CPS) [12] und Internet of Things (IoT) [13]). Hierfür werden meist serviceorientierte Middleware-Systeme (wie z.B. OPC-UA, Webservices, openIOT) [14] verwendet. Durch die Anwendung dieser Konzepte spielt die integrierte IT-Sicherheit aller beteiligten Komponenten eine immer größer werdende Rolle. Die fortschreitende Öffnung der

Netzwerke zeigt, dass die bisher verfolgte Strategie der Abschottung von Automatisierungsanlagen durch weitere Maßnahmen ergänzt werden müssen. In [11] wird beispielsweise dargestellt, dass mit Industrie 4.0 die Bedeutung der IT-Sicherheit weiter zunehmen wird. An der deutschen Normungs-Roadmap zu Industrie 4.0 [15] ist zudem erkennbar, dass im Kontext von Industrie 4.0 ein hoher Bedarf an standardisierten Vorgehensweisen und Modellen besteht.

Der Stand zu Beginn des Projektes ist zusammengefasst wie folgt zu beschreiben:

- Der Trend zur Vernetzung automatisierungstechnischer Komponenten nimmt stark zu (horizontale und vertikale Vernetzung).
- Die Nutzung cloudbasierter Dienste nimmt zu, damit verlassen Daten das Unternehmen in Richtung Cloud.
- Es ist absehbar, dass eine Reihe von Middleware-Protokollen (z. B. OPC-UA, DDS, MQTT) parallel im Einsatz ist. Auch in einer einzelnen Anlage.
- Diese Protokolle verfolgen unterschiedliche Konzepte in Bezug auf die IT-Sicherheit und insbesondere in Bezug auf die Administrationsfunktionen. Dies führt zu einem erheblichen Aufwand für die Betreiber, da für jedes der Protokolle unterschiedliche Funktionen zur Administration der IT-Sicherheit zum Einsatz kommen.
- Es besteht ein Bedarf an einer einheitlichen Lösung für die Administration der IT-Sicherheit in einer heterogenen Umgebung mit parallelem Einsatz verschiedener Middleware-Protokolle.

Dieser Situation soll das Projekt mit seinen Zielen Rechnung tragen.

2 Organisatorische Rahmenbedingungen

Die folgenden Kapitel beschreiben die organisatorischen Rahmenbedingungen im Projekt bei den ausführenden Stellen „Technische Hochschule Ostwestfalen-Lippe“ und „Hochschule Hannover“

2.1 Zusammenarbeit mit anderen Stellen

Im Rahmen des Projektes wurde mit folgenden anderen Stellen zusammengearbeitet:

- **Projektbegleitender Ausschuss**: Im projektbegleitenden Ausschuss haben sich in Summe zwölf Unternehmen beteiligt. Die Mitgliedsfirmen wurden während des Projektes in der Regel zweimal pro Jahr zu Statustreffen eingeladen. Hier wurden die Ergebnisse des Projektes vorgestellt und Teilergebnisse des Projektes durch die wissenschaftlichen Mitarbeiter vorgeführt. Neben der Teilnahme an den Statustreffen haben die Unternehmen Ergebnisdokumente von Teilergebnissen gelesen und kommentiert. Einigen der Unternehmen habe sich bereiterklärt bei gemeinsamen Veröffentlichungen mitzuwirken.
- **Plattform Industrie 4.0:** Über die Firma Phoenix Contact konnte über Herrn Lutz Jänicke ein intensiver Kontakt zur Arbeitsgruppe „Sicherheit vernetzter Systeme“ der Plattform Industrie 4.0 aufgebaut werden. Herr Jänicke konnte die Projektmitarbeiter in separaten Treffen über die Sichtweise der Plattform Industrie 4.0 informieren. Darüber hinaus hat er bei der Durchsicht der Arbeitsdokumente entsprechende Kommentierungen mit Hinweisen auf die Plattform Industrie 4.0 vorgenommen.
- **PROFIBUS Nutzerorganisation:** Das Arbeitspaket AP6 sollte sich mit der Absicherung eines Ethernet-basierten Echtzeitprotokolls befassen. Auf Grund der besonderen Bedeutung für den deutschen Markt war hier das Echtzeitprotokoll PROFINET ausgewählt worden. Zu diesem Protokoll lagen an beiden Standorten schon Erfahrungen aus dem vorangehenden Projekt SEC_PRO [16] vor. Während der Laufzeit des Projektes IT_SIVA stellte es sich im Jahr 2018 heraus, dass die PROFIBUS Nutzerorganisation (PNO) seine Arbeiten an einem Security-Layer für PROFINET deutlich intensiviert hat. Aus diesem Grund wurde die Abarbeitung des Arbeitspaktes AP6 geändert. Es wurde auf die Erarbeitung einer eigenen Lösung auf Basis des vorangehenden Projektes SEC_PRO verzichtet. Stattdessen arbeitet Herr Prof. Dr. Niemann in der Working Group Security (CB/WG10) mit und bringt so die Anforderungen des Projektes IT_SIVA in das Standardisierungsprojekt der PNO ein. Als Ergebnis dieser Arbeit hat Herr Niemann gemeinsam mit der PNO-Arbeitsgruppe ein White-Paper mit der Konzeptbeschreibung für ein PROFINET Security Layer [17] sowie einen Konferenzbeitrag [18] erstellt.

Die Zusammenarbeit mit anderen Stellen hat die Arbeiten im Projekt positiv beeinflusst. Durch die Schnittstelle zur Plattform Industrie 4.0 konnte deren Sichtweise gut in das Projekt eingebracht werden. Durch die Zusammenarbeit mit der PNO konnte das AP6 unter Zuarbeit von Prof. Dr. Niemann für das Projekt erarbeitet werden und gleichzeitig eine Vorarbeit für einen PROFINET Security Layer erbracht werden.

2.2 Verwendung der Zuwendungen

Hochschule Hannover (FE 1)

- Wissenschaftlich-technisches Personal (Einzelansatz A.1 des Finanzierungsplans)
  - HPA-A: 13,4 PM
  - HPA-B: 25,5 PM
- Geräte (Einzelansatz B des Finanzierungsplans): Keine
- Leistungen Dritter (Einzelansatz C des Finanzierungsplans): Keine

Technische Hochschule Ostwestfalen-Lippe (FE 2)

- Wissenschaftlich-technisches Personal (Einzelansatz A.1 des Finanzierungsplans)
  - HPA-A: 29,7 PM
  - HPA-B: 10,7 PM
- Geräte (Einzelansatz B des Finanzierungsplans): Keine
- Leistungen Dritter (Einzelansatz C des Finanzierungsplans): Keine

2.3 Wichtigste Positionen des zahlenmäßigen Nachweises

Die folgenden Unterkapitel listen die wichtigsten Positionen der zahlenmäßigen Nachweise der beteiligten Hochschulen.

2.3.1 Wesentliche Positionen zahlenmäßiger Nachweis Hochschule Hannover (FE 1)

A.1	Bruttoentgelte für wiss. techn. Personal	189.825,37 €
A.2	Bruttoentgelte für übriges Fachpersonal	0,00 €
A.3	Bruttoentgelte für Hilfskräfte	3.487,30 €
A.4	Pauschale für Personalausgaben	12.014,13 €
B.	Ausgaben für Gerätebeschaffung	0,00 €
C.	Ausgaben für Leistungen Dritter	0,00 €
D.	Pauschale für sonstige Ausgaben	10.043,39 €
	Summe	**215.370,19 €**

2.3.2 Wesentliche Positionen zahlenmäßiger Nachweis der Techn. Hochschule Ostwestfalen-Lippe (inIT) (FE 2)

A.1	Bruttoentgelte für wiss. techn. Personal	180.130,85 €
A.2	Bruttoentgelte für übriges Fachpersonal	0,00 €
A.3	Bruttoentgelte für Hilfskräfte	10.161,54 €
A.4	Pauschale für Personalausgaben	13.320,47 €
B.	Ausgaben für Gerätebeschaffung	0,00 €
C.	Ausgaben für Leistungen Dritter	0,00 €
D.	Pauschale für sonstige Ausgaben	40.722,57 €
	Summe	**244.335,43 €**

2.4 Notwendigkeit und Angemessenheit der geleisteten Arbeit

Die während des Projektes ausgeführten Arbeiten entsprechen den laut Projektantrag geplanten Arbeiten und waren daher notwendig und angemessen. Die Ziele des Projektes konnten erreicht werden. Die zu entwickelnden Demonstratoren sind betriebsfertig und können zur Vorführung der Projektergebnisse verwendet werden. Durch die Mitarbeit von Herrn Prof. Dr. Niemann in der Arbeitsgruppe Security der PROFIBUS Nutzerorganisation, konnte das im Projekt gewonnene Knowhow zeitnah in Arbeiten zur Absicherung von Ethernet-basierten Echtzeitprotokollen eingebracht werden.

2.5 Voraussichtlicher Nutzen der Ergebnisse des Vorhabens

Tabelle 1 zeigt die während des Projektes durchgeführten Transfermaßnahmen für die Mitglieder des projektbegleitenden Ausschusses.

Tabelle 1: Ergebnistransfer

Maßnahme	**Ziel**	**FE**	**Rahmen**	**Datum / Zeitraum**
Treffen mit dem projektbegleitenden Ausschuss	Übersicht der bisherigen Ergebnisse und Ansätze für den weiteren Verlauf	FE1+ FE2	Präsentation	25.01.2018
Treffen mit dem projektbegleitenden Ausschuss	Übersicht der bisherigen Ergebnisse und Ansätze für den weiteren Verlauf	FE1+ FE2	Präsentation	19.09.2018
Treffen mit dem projektbegleitenden Ausschuss	Übersicht der bisherigen Ergebnisse und Ansätze für den weiteren Verlauf	FE1+ FE2	Präsentation	29.01.2019

Treffen mit dem projektbegleitenden Ausschuss	Übersicht der bisherigen Ergebnisse und Ansätze für den weiteren Verlauf	FE1+ FE2	Präsentation	13.06.2019

Zusätzlich wurden Veröffentlichungen im Verlauf des Forschungsprojekts und Wissenstransfer in den Hochschuleinrichtungen in Form von Abschlussarbeiten zum Projektthema durchgeführt. Siehe hierzu die folgenden Kapitel 2.8 und 2.9. Außerdem können die Ergebnisse an den Demonstratoren an den Standorten Hannover und Lemgo vorgeführt werden.

Mit der Erstellung des Demonstrators liegen Konzepte und Architekturbeschreibungen vor, welche danach von allen interessierten Unternehmen in eigene Produkte umgesetzt werden können (die Spezifikation der Lösung wird in Form dieses Abschlussberichtes zur Verfügung gestellt). Ziel dabei ist, dass die Konzepte und Architekturen an kein spezifisches Busprotokoll gebunden sein sollen, sondern als generischer Ansatz auf verschiedene industrielle Kommunikationsprotokolle anwendbar sind. Der konkrete wissenschaftlich-technische Nutzen für interessierte KMU besteht darin, dass sie über die Projektergebnisse Zugang zu Systemkonzepten und Referenzarchitekturen erhalten, die sie mit eigenen Mitteln in der Regel nicht entwickeln könnten. Größere Unternehmenkönnen die gewonnenen Erkenntnisse ebenfalls als Basis für weitere Eigenentwicklungen heranziehen. Auf Wunsch erhalten interessierte Firmen (insbesondere KMU) eine Unterstützung bei der Einbindung der entwickelten Lösung in eigene Produkte. Zudem können die Modellfabriken des inIT und der Hochschule Hannover zu Demonstrations- und Vorführungszwecken genutzt werden. Außerdem können KMU, die als Technologie- und Dienstleistungsanbieter tätig sind, das nötige Know-How erwerben, um die Integration der entwickelten Lösungen ihren Kunden als Dienstleistung anzubieten. Tabelle 2 sowie Tabelle 3 zeigen die Transfermaßnahmen während Projektlaufzeit und nach Projektende und deren bisherigen Abarbeitungsgrad bzw. die Einschätzung der Realisierbarkeit.

Tabelle 2: Umsetzung der Transfermaßnahmen während der Projektlaufzeit

Maßnahme	Ziel	FE	Ort/Rahmen	Datum/Zeitraum
Projektinterne Abstimmung mit dem PA	Ausführliche Diskussion und Vorstellung erzielter Ergebnisse, Prüfung auf Praxisrelevanz, Einflussnahme auf Konzepte und Ansätze, Beratung durch Expertenkreis, enge und zielgerichtete Zusammenarbeit mit dem PA	FE1+ FE2	Abwechselnd HsH/inIT Projekt-Newsletter	Erfüllt, siehe Auflistung der Treffen mit dem projektbegleitenden Ausschuss in Tabelle 1. Zusätzlich diverse Dokumentenreviews und Kommentierungen durch den Ausschuss
Meeting zur Aufnahme von Anforderungen der Firmen des PA	Die Firmen des PA können eigene Anforderungen an die IT-Sicherheitsarchitektur definieren	FE1+ FE2	HsH oder inIT	Erfüllt. Review und Kommentierung der Anforderungsspezifikation durch die Unternehmen ist erfolgt.

Bereitstellung der Referenzmodelle für interessierte Firmen	Nutzbarmachung der erarbeiteten Systemkonzepte und Referenzmodelle	FE1	HsH oder InIT	Erfüllt: Alle im Projekt erstellten Arbeitsdokumente können ‚auf Anfrage durch die Forschungseinrichtungen, für interessierte Unternehmen bereit gestellt werden.
Studien-/ Abschlussarbeiten zum Projektthema	Wissenstransfer in den Hochschuleinrichtungen	FE1+ FE2	Studierende der Technischen Hochschule OWL bzw. Hochschule Hannover	Erfüllt: Siehe Auflistung der Abschlussarbeiten in Kapitel 2.9.
Einbindung in die Lehre	Vermittlung von Grundlagen und Ergebnissen in der Lehre	FE1+ FE2	Lehrveranstaltung Datensicherheit und Network Security Lehrveranstaltung Integrierte Automation und Schulung: IT-Sicherheit in der Automatisierungstechnik an der HsH	**TH-OWL**: Erfüllt: Behandlung ausgewählter Inhalte in den genannten Lehrveranstaltungen **HsH:** Erfüllt. Das Fach IT-Sicherheit in Produktionsanlagen wurde in das Curriculum der Fakultät II als Pflichtveranstaltung ab der Akkreditierung im Jahr 2019 aufgenommen
Einrichten einer Projekthomepage	Verbreitung der Projektthematik und Lösungsansätze	FE2	Homepage des inIT und der Hochschule Hannover	**inIT**: Erfüllt: Projekthomepage eingerichtet: https://www.th-owl.de/init/forschung/projekte/b/filteroff/418/single.html
Berichterstattung im jährlich erscheinenden Institutsbericht (inIT)	Verbreitung der Projektergebnisse	FE2	Institutsbericht (inIT)	**inIT**: Erfüllt: Darstellung des Projekts im Jahresbericht 2017-2018 des inIT
Teilnahme an der AG4 der Plattform Industrie 4.0	Verbreitung von Projektergebnissen, Ideenaustausch	FE1+ FE2	AG4 „Sicherheit vernetzter Systeme" in der Plattform Industrie 4.0	Erfüllt: Regelmäßige Treffen / Telefonkonferenzen mit Vertreter der AG4 haben stattgefunden. Review der Dokumente im Projekt

				durch Vertreter der AG4.
Veröffentlichung der Ergebnisse im Rahmen von Workshops, Konferenzen und Fachzeitschriften	Verbreitung der Projektergebnisse, zeitnaher Transfer	FE1+ FE2	Jahreskolloqium KommA, IEEE ETFA, IEEE WFCS	Erfüllt: Siehe Auflistung in Kapitel 2.8.

Tabelle 3: Umsetzung der Transfermaßnahmen nach Projektende

Maßnahme	Ziel	FE	Ort/Rahmen	Datum/Zeitraum
Projekthome-page	Verbreitung der Projekter-gebnisse über die Projekt-laufzeit hinaus	FE2	Homepage des inIT und der Hochschule Hannover	**inIT**: Dauerhaft abruf-bar auch nach Projekt-ende, Projekthome-page: https://www.th-owl.de/init/for-schung/projekte/b/fil-teroff/418/single.html
Einbindung in die Lehre	Vermittlung von Grundlagen und Ergebnissen in der Lehre	FE1+ FE2	Lehrveranstal-tung Datensi-cherheit und Network Security Lehrveranstal-tung Integrierte Automation und Schulung: IT-Si-cherheit in der Automatisie-rungstechnik an der HsH	**TH-OWL**: Erfüllt. Pro-jektspezifische Grund-lagen (PKI) werden auch in zukünftigen Lehrveranstaltungen behandelt. **HsH:** Erfüllt. Das Fach IT-Sicherheit in Pro-duktionsanlagen wurde im Sommer 2019 in das Curriculum der Fa-kultät II als Pflichtver-anstaltung aufgenom-men.
Veröffentlichung der Spezifika-tion	Die Spezifikation der entwi-ckelten IT-Sicherheitsarchi-tektur wird im Rahmen des Abschlussberichtes frei zur Verfügung gestellt.	FE1	SerWiss-Server der Hochschule Hannover. Registrierung des Dokumen-tes mit einer fes-ten Dokumen-tennummer (DOI und URN). Aufnahme des Dokumentes in den Katalog der Deutschen Nati-onalbibliothek	Erfolgt in Form des ausführlichen Projek-tabschlussberichtes.

Unterstützung beteiligter KMU	Die an dem Projekt beteiligen KMU werden bei der Integration der entwickelten Lösung in eigene Produkte unterstützt.	FE1+ FE2		Den beteiligten Unternehmen wurde ein Übergabeworkshop für die erarbeiteten SW-Komponenten angeboten, dieser konnte aber aus organisatorischen Gründen nicht mehr stattfinden. Softwarequellen können von den Forschungseinrichtungen auf Anfrage bereitgestellt werden.
Vorstellung der Projektergebnisse in Verbandsgremien	Verbreitung der Projektergebnisse in die Wirtschaft	FE 1 oder FE 2	ZVEI-Forschungsgemeinschaft Automation	Einladung für Sitzung im März 2020 ist bereits erfolgt.
Erstellung eines Abschlussberichts	Umfassende Darstellung aller Projektergebnisse	FE1+ FE2	Gemeinsamer Abschlussbericht von HsH und inIT, Website der Forschungsvereinigung, tib Hannover	Ca. 5 Monate nach Projektende
Projektpräsentation	Verbreitung der Projektergebnisse	FE1+ FE2	Messestand, regionale Veranstaltungen (z.B. des CIIT), Veranstaltungen des ZVEI	Planung für die Messe TWENTY2X[1] (Nachfolger CEBIT) für das Jahr 2020 im Rahmen des niedersächsischen Zukunftslabors Produktion[2]

2.6 Veröffentlichung der Ergebnisse

Die im Rahmen des Forschungsprojektes veröffentlichten Konferenzbeiträge und Projektarbeiten sind nachfolgend aufgeführt.

AUTOMATION 2018 – Konferenz der GMA im VDI

Konferenzbeitrag mit dem Titel „Sichere Middleware-Lösungen für die Industrie 4.0 – Eine IT-Sicherheitsanalyse aktueller Kommunikationsansätze“ [19]. In dem Beitrag werden etablierte Middleware-Lösungen unter Berücksichtigung verschiedener Kriterien evaluiert und miteinander verglichen. Der Beitrag wurde angenommen und anlässlich der AUTOMATION 2018 (Baden-Baden) vorgestellt.

[1] https://www.twenty2x.de/
[2] https://www.zdin.de/zukunftslabore

ETFA 2018 – IEEE-Konferenz „Emerging Technologies & Factory Automation“

Internationaler Konferenzbeitrag mit dem Titel „PKI and User Access Rights Management for OPC UA based Applications“ [20]. Die Sicherheitsmerkmale des Open Platform Communications Unified Architecture (OPC UA) Frameworks werden analysiert, um eine durchgängig sichere Kommunikation in Bezug auf Authentifizierung und Autorisierungsoptionen zu gewährleisten. Möglichkeiten der Online- und Offline-Validierung von X509-Zertifikaten mit Public-Key-Infrastrukturen (PKI) für OPC UA-basierte Anwendungen werden detailliert dargestellt. Der Beitrag wurde angenommen und anlässlich der ETFA 2018 (Turin) vorgestellt.

KommA 2018 – Konferenz zur „Kommunikation in der Automation“

Angenommener Beitrag mit dem Titel „A comparative evaluation of security mechanisms in DDS, TLS and DTLS” [21]. In diesem Beitrag werden die End-to-End-Sicherheitsmechanismen des Transport Layer Security (TLS) sowie des Datagram Transport Layer Security (DTLS) Standards und der sicherheitsrelevanten Plugins innerhalb der Data Distribution Service (DDS) Spezifikation analysiert und verglichen. Die Ergebnisse geben Aufschluss darüber, ob und warum die Verwendung eines DDS-spezifischen Sicherheitsprotokolls anstelle des Einsatzes von TLS/DTLS erforderlich ist. Darüber hinaus werden die grundlegenden Unterschiede zwischen TLS und DTLS diskutiert und die Besonderheiten der DDS-Sicherheit hervorgehoben. Vorgetragen auf der KommA 2018 (Lemgo).

AALE 2019 – Konferenz zur “Angewandten Automatisierungstechnik in Lehre und Entwicklung”

Konferenzbeitrag mit dem Titel „Änderung von IT-Security Anforderungen durch den Wandel der Automatisierungsstrukturen im Kontext von Industrie 4.0” [22]. In dem Beitrag werden Anforderungen an die IT-Sicherheit beschrieben, die bei gegenwärtigen und zukünftigen Automatisierungsstrukturen berücksichtigt werden müssen. Der Beitrag wurde vom auf der AALE 2019 (Heilbronn) vorgestellt.

AUTOMATION 2019 – Konferenz des VDI des Bereichs GMA

Gemeinschaftlicher Konferenzbeitrag mit dem Titel „Entwicklung einer IT-Sicherheitsinfrastruktur für verteilte Automatisierungssysteme“ [23]. In diesem Beitrag werden Teilergebnisse des Projekt IT_SIVA bis zum Veröffentlichungszeitpunkt vorgestellt. Der Vortrag fand auf der AUTOMATION 2019 (Baden-Baden) statt.

INDIN 2019 – IEEE-Konferenz „Industrial Informatics”

Internationaler Konferenzbeitrag mit dem Titel „Enhancing Authorization Mechanisms using Attribute Certificates for OPC UA based Applications” [24]. Der Beitrag beschreibt die Möglichkeit, eine Public-Key-Infrastruktur durch die Nutzung von Attribut-Zertifikaten um eine zusätzliche Autorisierungsoption zu erweitern. Die Veröffentlichung fand auf der INDIN 2019 (Helsinki) statt.

INDIN 2019 – IEEE-Konferenz „Industrial Informatics"

Internationaler Konferenzbeitrag mit dem Titel „IT security extensions for PROFINET". Der Beitrag beschreibt die grundlegenden Überlegungen zu Etablierung einer sicheren Kommunikation mit dem Echtzeitprotokoll [18]. Dieser Beitrag ist nicht direkt im Projekt entstanden. Prof. Dr. Niemann hat jedoch in Umsetzung des Arbeitspaketes 6 in der WG Security der PROFIBUS-Nutzerorganisation an diesem Konzept mitgearbeitet.

2.6.1 Abschlussarbeiten

Abschluss- und Projektarbeiten, die im Förderzeitraum angefertigt wurden und einen direkten Bezug zur Durchführung des Vorhabens IT_SIVA hatten:

Arbeiten an der Technischen Hochschule Ostwestfalen-Lippe

- „Konzept und Implementierung einer Benutzerauthentisierung an einem OPC-UA-Server mittels einer Smart-Card und eines mobilen Endgerätes". Oliver Konradi, Bachelorarbeit 2018
- „Implementierung eines Schlüsselverwaltungssystems mithilfe eines Trusted Platform Moduls (TPM)". Wadim Halle, Bachelorarbeit 2018
- "Evaluation of Certificate Provisioning Protocols for Public Key Infrastructures in Distributed Systems". Maxim Friesen, Projektarbeit 2019 [25].

Arbeiten an der Hochschule Hannover

- „Entwurf einer IT-Security Referenzarchitektur für Industrie 4.0 basierte Systeme". Kai Steinke, Masterarbeit 2018 [26]
- „Evaluierung eines einheitlichen Sicherheitskonzeptes für die Middleware-Protokolle MQTT und DDS". Sebastian Tebbje, Masterarbeit 2019 [27]

3 Ergebnisse

In diesem Kapitel werden die wichtigsten Ergebnisse der jeweiligen Arbeitspakete gemäß Antrag des Projektes IT_SIVA zusammengefasst.

3.1 AP 1 Voruntersuchung

Das Arbeitspaket 1 „Voruntersuchung“ dient im Wesentlich dazu, den aktuellen Stand der Technik bezüglich der Kommunikation in verteilten Systemen über Middleware-Lösungen im Kontext von Industrie 4.0 zu erörtern. Zunächst wird beschrieben, was eine Middleware genau adressiert und wofür sie genutzt wird. Die stärkere Vernetzung und dezentralisierte Organisation zukünftiger Anlagenstrukturen sowie die weltweit möglich gewordene Kommunikation von technischen Systemen über das Internet fordern den Einsatz von Middleware-Systemen für die Umsetzung der I4.0-Kommunikation. Middleware entspricht einer zusätzlichen Schicht in einer komplexeren Software-Struktur, die Zugriffsmechanismen auf untergeordnete Schichten vereinfacht, Details dieser Infrastruktur jedoch verbirgt. Die Middleware stellt Funktionen zur Verteilung und Dienste zur Unterstützung der Anwendung bereit, fungiert demnach als eine Art Plattform bestehend aus einem oder mehreren Protokollen. Eine Middleware sorgt dafür, die Anwendungsprogramme zu entlasten und ermöglicht mit erhöhter Produktivität eine Optimierung des Entwicklungsprozesses.

Um eine Grundlage für die weitere Bearbeitung zu schaffen, wurden verschiedene, standardisierte und am Markt frei verfügbare Middleware-Lösungen näher betrachtet und beschrieben. Tabelle 4 gibt einen Überblick über die betrachteten Protokolle und deren Einsatzgebiete.

Tabelle 4: Untersuchte Protokolle und deren Einsatzgebiete

Protokoll	**Einsatzgebiet**
WS SOAP	Remote Procedure Calls
WS REST	Remote Procedure Calls
COAP	Resourcenbeschränkte Geräte
MQTT	Smart Home, Resourcenbeschränkte Geräte
AMQP	Finanzsektor
XMPP	Instant Messaging
IoTivity	Smart Home
LWM2M	Machine-to-Machine
OPC UA	Machine-to-Machine
DDS	Machine-to-Machine

In den folgenden Unterkapiteln werden die für die spätere Bearbeitung ausgewählten Protokolle exemplarisch vorgestellt. Die Beschreibung der restlichen Protokolle aus Tabelle 4 befinden sich in dem Dokument „Zusatzdokument zu AP1[3]“. Die Evaluationskriterien, sowie ein Vergleich der einzelnen Protokolle erfolgt unter Kapitel 3.5 Middleware.

3.1.1 Open Platform Communications Unified Architecture (OPC UA)

3.1.1.1 Allgemeine Beschreibung

Die Open Platform Communication Unified Architecture (OPC UA) ist eine plattformunabhängige, serviceorientierte Architektur für verteilte Automatisierungssysteme mit sicheren Transportmechanismen und Datenmodellierung. Sie kann als Schnittstelle zwischen Automatisierungssystemen in verschiedenen Ebenen der Automatisierungspyramide [28] eingesetzt werden. Abbildung 1 zeigt verschiedene OPC UA Protokolloptionen, die die Kompatibilität mit verschiedenen Kommunikationsprotokollen in industriellen Netzwerken zeigen [29]. Durch die Sicherheitsfunktionen dieser Middleware kann eine durchgängige Sicherheit für die unternehmensübergreifende Kommunikation ermöglicht werden. Die von OPC UA bereitgestellten Informationsmodellierungsstandards ermöglichen die Implementierung eines Standarddatenmodells für den Informationsaustausch. Es gibt mehrere Anwendungsfälle für diese Architektur, z.B. verwendet die Überwachung eines Offshore-Windparks OPC UA Server zur Datenbereitstellung für SCADA-Systeme (Global Tech I-Projekt) [30].

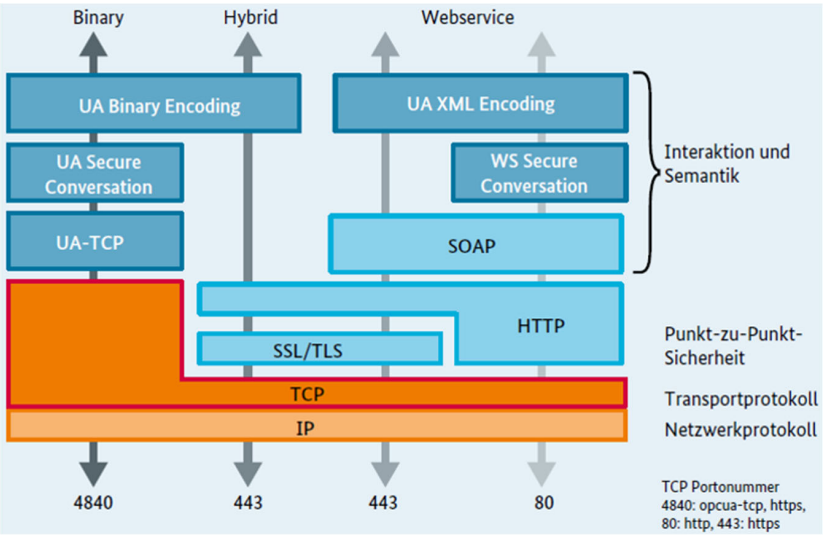


Abbildung 1: Optionen des OPC UA-Protokolls [29]

3.1.1.2 Ebene

OPC UA ist ein Middleware-Framework, das verschiedene Protokolloptionen von der Transportschicht bis zu höheren Schichten unterstützt (vgl. Abbildung 1).

[3] Abrufbar unter www.fe-zvei.org

3.1.1.3 Standardisierung

OPC UA ist in der Normreihe IEC 62541 mit 13 Spezifikationsteilen standardisiert, von denen die ersten 7 Teile Kernspezifikationen wie Konzepte, Sicherheitsmodell, Adressraummodell, Dienste, Informationsmodell, Mappings und Profile sind. Die anderen Teile sind Zugriffsartenspezifikationen wie Datenzugriff, Alarm und Bedingungen, Programme, Zugriff auf Historie und Aggregate, ein Teil über die Discovery Services. Dies sind verschiedene Teile der Spezifikationen, die derzeit als stabile Version [31] verfügbar sind.

3.1.1.4 Architektur

OPC UA ist eine serviceorientierte Architektur. Die Informationsmodellierung ist einer der Vorteile von OPC UA [32]. Im Gegensatz zu klassischem OPC ermöglicht die Informationsmodellierung von OPC UA die Darstellung der Semantik von Daten.

Im Allgemeinen ist es nicht erforderlich, dass ein OPC UA-Client über ein integriertes OPC UA-Informationsmodell verfügt und diese Informationen an einen OPC UA-Server weitergeben muss [28].

3.1.1.5 Nachrichten Paradigma

OPC UA basiert auf einem Request/Response-Paradigma für Client/Server-Anwendungen.

Neben dem regulären Nachrichtentyp Request/Response, wurde OPC UA kürzlich um eine Spezifikation für Publish/Subscribe-Dienste erweitert [31].

3.1.1.6 Echtzeitfähigkeit

OPC UA TSN (Time Sensitive Networking) ist eine in Arbeit befindliche standardisierte Spezifikation, die die harte Echtzeitkommunikation mit dem Publish/Subscribe-Modell [33], [34], [35] unterstützt. Das Pub/Sub-Modell wird mit Real-Time Physical Layer (Ethernet mit TSN) kombiniert, um echtzeitfähige OPC UA [34] zu erreichen.

3.1.1.7 Servicequalität

Die Parameter für die Servicequalität basieren auf den Transportprotokollen. Für das allgemeine Servicekonzept bietet OPC UA Timeout-Handling und Fehlerbehandlung [28].

3.1.1.8 Sicherheit

Die OPC UA Sicherheitsarchitektur [36] definiert einen mehrschichtigen Ansatz für die Sicherheit, wie in Abbildung 2 dargestellt. Die Anwendungsschicht oben in Abbildung 2 dient der Übertragung von Anlageninformationen, Einstellungen, Anweisungen und Echtzeitdaten von Geräten zwischen einem Client und einem Server während einer Sitzung. Die Sitzung dient der Authentifizierung und Autorisierung der Benutzer, die mit dem Client und bestimmten Produkten arbeiten. Eine Sitzung läuft über den SecureChannel. Der SecureChannel verfügt über drei verschiedene Betriebsarten, wie nachfolgend erläutert. Die Integrität wird durch digitale Signaturen und die Vertraulichkeit durch Verschlüsselung der Informationen der übertragenen Nachrichten garantiert. Die unterste Schicht ist die Transportschicht, die die Übertragung von gesicherten Daten über Socket-Verbindungen ermöglicht [28].

Nach der Socket-Erstellung gibt es drei Modi, die ein Client beim Öffnen eines sicheren Kanals nutzen kann [28].

- „None“ - OpenSecureChannel-Nachricht wird nicht gesichert.
- „Sign“ - Die Nachricht wird mit dem zugehörigen privaten Schlüssel des Application Instance Certificate des OPC UA Clients signiert.
- „Sign and Encrypt“ - Die Nachricht wird zusätzlich mit dem öffentlichen Schlüssel des Application Instance Certificate des Servers verschlüsselt (d.h. zusätzlich zum Sign-Modus).

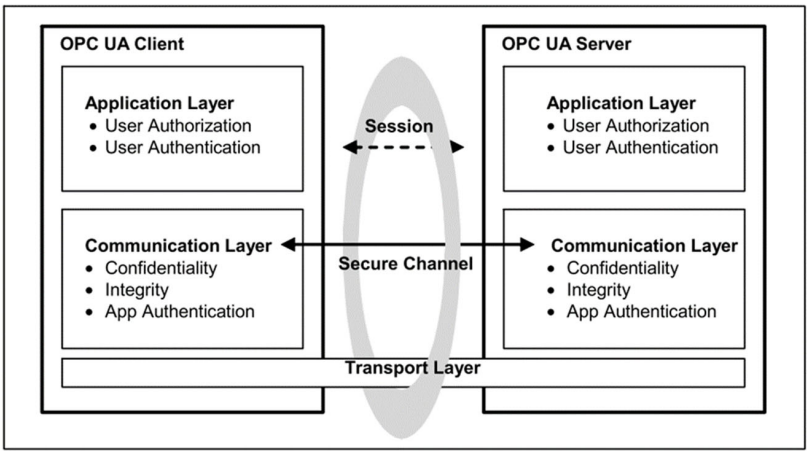


Abbildung 2: OPC UA Sicherheitsarchitektur [36]

Die Zertifikate der Anwendungsinstanz werden in anderen Modi als dem Modus "None" ausgetauscht. Asymmetrische Schlüssel werden verwendet, um die SecureChannel-Nachrichten zu sichern, und symmetrische Schlüssel werden aus Geheimnissen generiert, die zwischen Client und Server während der Einrichtung des SecureChannels geteilt werden. Weitere Nachrichten werden gemäß MessageSecurityMode und SecurityPolicy gesichert, die mit den erzeugten symmetrischen Schlüsseln ausgehandelt werden.

Die folgenden Arten der Benutzerauthentifizierung können mit dem OPC UA-Protokoll verwendet werden [37].

- AnonymousIdentityToken - Es sind keine Benutzerinformationen verfügbar (wenn keine Benutzerauthentifizierung erforderlich ist).
- UserNameIdentityToken - Ein Benutzer, der durch Benutzername und Passwort identifiziert wird.
- X509IdentityToken - Ein Benutzer, der durch ein X509v3-Zertifikat identifiziert wird.
- IssuedIdentityToken - Ein Benutzer, der durch einen WS-SecurityToken identifiziert wird.

Daher sind sowohl die Anwendungsschicht als auch die Kommunikationsschicht von der Standardinfrastruktur zur Verwaltung der in der Kommunikation verwendeten Zertifikate und Schlüsselpaare abhängig. Der OPC UA Standard spezifiziert nicht, wie eine solche Infrastruktur aussieht, da es viele verschiedene Konzepte gibt, die alle von den konkreten Umgebungen und Anforderungen abhängen [28]. Ein solches Beispiel ist eine PKI (Public Key Infrastructure) [38] die für den Bereich der industriellen Automatisierung verwendet werden kann.

So kann beispielsweise das mehrschichtige Sicherheitsmodell mit PKI [38] implementiert werden, wie in Abbildung 3 dargestellt. Es kann verwendet werden, um den gesicherten Modus der Datenübertragung im OPC UA-Protokoll zu demonstrieren.

Die Abbildung 3 stellt sich wie folgt dar. Es gibt 3 verschiedene Einheiten in einer PKI, die Certification Authority (CA), die Registration Authority (RA) und die Validation Authority (VA). Die CA stellt das Vertrauensverhältnis zwischen den beiden an einer Kommunikation beteiligten Partnern her, indem sie das signierte Zertifikat für den anfragenden Partner ausstellt. Die RA dient als Registrierungsbehörde, die die Identität dem angeforderten Benutzer/Client zuordnet. Die VA ist für die Überprüfung der Validierung des Zertifikats verantwortlich, das für die vertrauenswürdige Kommunikation verwendet wird. Diese Validierung kann offline oder online erfolgen, z.B. durch Verwendung einer Certificate Revocation List (CRL) [38] oder eines Online Certificate Status Protocol (OCSP) [39]. Die CA an RA und VA delegiert die oben beschriebenen Aufgaben. Abbildung 3 zeigt auch, dass ein Benutzer die RA um ein signiertes Zertifikat von einer CA bittet, indem er eine Certificate Signing Request (CSR) [40] sendet, dann überprüft die RA die Identität des Benutzers an die CA, woraufhin das CA signierte Zertifikat an den angeforderten Benutzer ausgestellt wird. Nun verwendet der Benutzer das empfangene Zertifikat, um eine vertrauenswürdige Kommunikation mit der Anwendung zu ermöglichen, die auch der gleichen CA vertraut, die das Benutzerzertifikat ausgestellt hat. Ähnlich wie bei dem vorstehend erläuterten Beispiel kann eine PKI-Infrastruktur für ein Zertifikat aufgebaut werden, das für den Aufbau einer sicheren Kommunikation über das OPC-UA-Protokoll erforderlich ist.

Die Attribute der Zugriffsebene und der Benutzerzugriffsebene sind in [41] beschrieben. Diese Attribute bestimmen die für die Knoten konfigurierten Zugriffsebenen und die durch die Konfiguration der Benutzerzugriffsebenen bestimmte Benutzerberechtigung. Die in [41] beschriebene Rollenspezifikation ist eines der neuesten Updates der OPC Foundation, um ihre Unterstützung für die rollenbasierte Autorisierung hervorzuheben. Das OPC UA Adressraummodell beschreibt eine Base NodeClass, von der alle anderen NodeClasses abgeleitet sind. Beim Vergleich der möglichen Attribute (d.h. der obligatorischen und optionalen Attribute) einer in Tabelle 2 in [41] definierten Base NodeClass mit Tabelle 7 in [41] werden zusätzliche optionale Attribute zur Unterstützung von Rollen und Zugriffsbeschränkungen betrachtet. Diese Attribute sind nun optional für alle abgeleiteten Nodeklassen. Ihre Vorteile und Herausforderungen liegen in der Klarheit der Umsetzung.

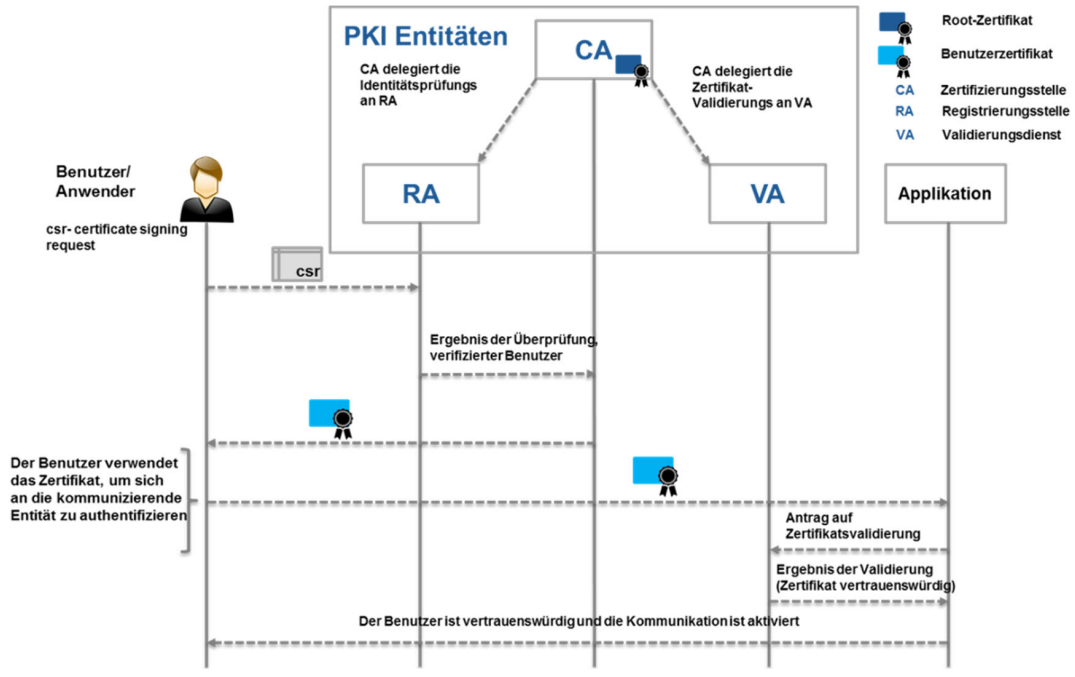


Abbildung 3: Einfache Public-Key-Infrastruktur

3.1.1.9 Physikalische Anforderungen

OPC UA unterstützt eingeschränkte Geräte. Die Protokolloptionen bieten binäre Verschlüsselung über TCP, um dies zu erleichtern.

3.1.1.10 Verfügbare Implementierungen

Es gibt verschiedene Open Source OPC UA (Stack und SDK) Implementierungen wie Milo (Java) [42], open62541 (C basierte Bibliothek) [43], Free OPC UA (Python) [44], etc. sowie kommerzielle Produkte wie Prosys OPC (Java) [45], und den OPC-UA Stack der Firma Softing (.NET-basiert) [46]. Alle oben genannten Implementierungen haben verschiedene OPC –UA-Funktionen implementiert.

3.1.2 Message Queuing Telemetry Transport (MQTT)

3.1.2.1 Allgemeine Beschreibung

Message Queuing Telemetry Transport (MQTT) [47] ist ein leichtgewichtiges, Broker-basiertes Publish/Subscribe-Messaging-Protokoll und wurde von der OASIS standardisiert. Es ist so konzipiert, dass es offen, einfach zu implementieren und für eingeschränkte Geräte mit begrenzten Verarbeitungs- und Speicherkapazitäten geeignet ist, um Daten über Netzwerke mit niedriger Bandbreite zu senden. Das Protokoll ist für latente oder unzuverlässige Netzwerke und für die Machine-to-Machine-Kommunikation im Internet der Dinge (IoT) ausgelegt. Die Designprinzipien von MQTT zielen darauf ab, den Bedarf an Netzwerkbandbreite und Geräteressourcen zu minimieren und gleichzeitig Zuverlässigkeit zu gewährleisten, indem drei Quality of Service Levels definiert werden.

3.1.2.2 Ebene

MQTT ist ein Application Layer Protocol und hat Implementierungen in verschiedenen Programmiersprachen wie Java, C, C++, JavaScript, Lua, Python und auch C#. Diese ermöglichen eine einfache Bereitstellung auf bestehenden Geräten und die Integration in Unternehmenssysteme.

3.1.2.3 Standardisierung

MQTT wurde 2013 von der Organisation zur Förderung von strukturierten Informationsstandards (OASIS) standardisiert.

3.1.2.4 Architektur

MQTT ist eine nachrichtenorientierte Middleware.

Im Gegensatz zu High-Level-Konzepten wie SOA oder ROA ist ein MOM eine eigentliche Implementierung einer Middleware-Plattform. Es kann zur Implementierung einer SOA, einer EDA oder anderer Architekturen verwendet werden. Normalerweise bereichert ein MOM eine Reihe von Anwendungen mit asynchronem Messaging, bei dem ein MOM-Server die Nachrichten speichert und weiterleitet.

3.1.2.5 Nachrichten Paradigma

MQTT basiert auf dem Publish/Subscribe-Paradigma. Sie besteht aus mindestens einem Publisher, einem Subscriber und einem zentralen Broker. Der MQTT Broker ist ein Server, der alle

Nachrichten von den Clients empfängt und sie dann an bestimmte Ziele weiterleitet. In einer typischen Publish/Subscribe-Architektur wie in Abbildung 4 gibt es mehrere Publisher, die sich mit dem Broker verbinden, um ihre Daten zu senden, z.B. Geräte, die Sensordaten bereitstellen. Topics ermöglichen es dem Publisher, die Daten zu kategorisieren. Die Clients können Anwendungen oder andere Geräte sein, die sich für bestimmte Topics interessieren und für diese eine Subscription vornehmen können. Sie erhalten alle neuen Daten, die zu den Topics veröffentlicht werden, für die eine Subscription vorliegt. Das Publish/Subscribe-Modell hat gegenüber dem Request/Response-Modell [48] Vorteile, da Push-Messaging in der Regel schneller ist und keine vorherigen Ressourcenanfragen erfordert.

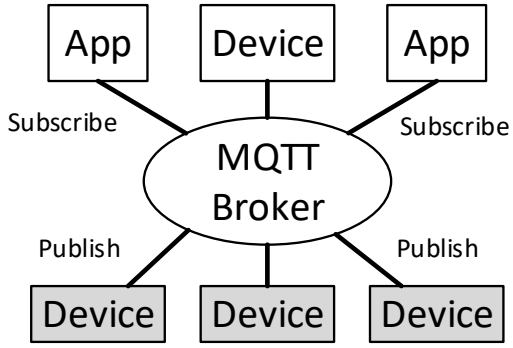


Abbildung 4: MQTT Pub/Sub-Architektur

3.1.2.6 Echtzeitfähigkeit

MQTT ist darauf ausgerichtet, den Overhead zu minimieren und hochfrequentes Telemetrie-Messaging anzubieten. MQTT ermöglicht die Bereitstellung von Daten in weicher Echtzeit [49].

3.1.2.7 Servicequalität

MQTT definiert drei Quality of Service (QoS)-Level, um die Zuverlässigkeit der Nachrichtenzustellung zu gewährleisten:

QoS-Level 0: Nachrichten werden höchstens einmal zugestellt. Es kann zu einem Nachrichtenverlust kommen.

QoS-Level 1: Nachrichten werden mindestens einmal unter Nutzung eines Acknowledgments (mit der Funktion PUBACK/SUBACK) zugestellt. Die gleiche Nachricht kann mehrmals empfangen werden.

QoS Level 2: Nachrichten werden genau einmal zugestellt. In einem ersten Schritt sendet der Publisher einen PUBLISH, dann erhält er einen PUBACK, dann sendet er einen PUBREL und am Ende dieser Transaktion erhält er einen PUBCOMP. Nachrichten werden nicht mehrfach gesendet oder empfangen.

MQTT bietet weitere QoS-Funktionen wie:

- "Last Will and Testament": Wenn der Client unerwartet die Verbindung zum Broker trennt, wird die letzte Nachricht gespeichert, die der Client durch die Verbindung zum Broker angibt. Wenn der Broker feststellt, dass sich der Client abrupt trennt, sendet der Broker die Nachricht an alle Subscriber zu dem jeweiligen Topic.

- "Message Persistence": Wenn die Verbindung zwischen Broker und Client getrennt wird, kann der Broker eine neue Nachricht für diesen Client speichern, um sie später zu übertragen.
- "Heartbeat & Keep Alive": Clients können die Keep-Alive-Zeit definieren, die inaktive Verbindungen hält. Beim Senden einer PINGREQ kann der Client die Keep-Alive-Zeit auf null setzen [50].

3.1.2.8 Sicherheit

MQTT bietet ein leichtes und einfach zu bedienendes Kommunikationsprotokoll, daher sind nicht viele Sicherheitsmerkmale spezifiziert. Die Idee von MQTT ist es, es durch anerkannte Standards zu ergänzen. Die Authentifizierung erfolgt mit der eindeutigen Client-ID, die der Client in der MQTT CONNECT-Nachricht bereitstellt. Im Authentifizierungsprozess werden neben der Benutzername-Passwort-Authentifizierung auch Client-IDs verwendet. Dies ermöglicht es, den Zugriff des Clients auf bestimmte Topics zu beschränken und das Publishen oder das Subscriben von nicht autorisierten Topics zu verhindern [51], [52]. MQTT kann in Kombination mit TLS verwendet werden. TLS (Transport Layer Security) bietet einen sicheren Kommunikationskanal zwischen einem Client und einem Server [53].

3.1.2.9 Physikalische Anforderungen

Fixed header, present in all MQTT Control Packets
Variable header, present in some MQTT Control Packets
Payload, present in some MQTT Control Packets

Abbildung 5: Struktur eines MQTT Control Packets [47]

Bit	7	6	5	4	3	2	1	0
byte 1	Packet Identifier MSB							
byte 2	Packet Identifier LSB							

Abbildung 6: Fixed-Header Format [47]

MQTT-Clients benötigen nur wenige Ressourcen und können auf Geräten wie Sensoren und Aktoren laufen, die durch wenig Speicher, Rechenleistung eingeschränkt sind. Der Overhead wird minimal gehalten, indem die Kontrollpakete so klein wie möglich gehalten werden. Wie in Abbildung 5 zu sehen ist, geschieht dies durch eine Unterteilung in drei Teile, den fixed-Header, den variablen Header und die Nutzlast (Payload). Der 2 Byte große fixed Header (Abbildung 6) ist immer vorhanden, während der größere variable Header und die Nutzlast (Payload) nur bei Bedarf hinzugefügt werden.

Durch minimale Header, geringen Speicherbedarf und begrenzte Abhängigkeit von Bibliotheken ist MQTT speziell für eingeschränkte Geräte konzipiert.

3.1.2.10 Verfügbare Implementierungen

MQTT ist ein offener Standard. Kostenlose Versionen sind z.B. Eclipse Mosquitto mit C und C++ Client-Bibliotheken [54], HiveMQ [55] und Eclipse Paho [56].

3.1.3 Data Distribution Service (DDS)

3.1.3.1 Allgemeine Beschreibung

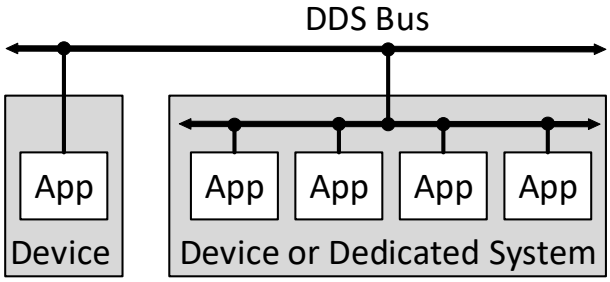


Abbildung 7: DDS-Architektur in einem Edge-Netzwerk mit Edge-Geräten

Der Data Distribution Service (DDS) [57] ist ein Middleware-Protokoll und API-Standard für datenzentrische Konnektivität, der von der Object Management Group (OMG) entwickelt wurde. DDS bietet eine hohe Anzahl von Quality of Service (QoS)-Parametern, die sich auf die Interoperabilität, die geringe Latenzzeit der Datenverbindung, die Zuverlässigkeit und eine skalierbare Architektur beziehen. DDS-Implementierungen bieten Datenkommunikation, die für harte Echtzeit-Systeme geeignet ist. Durch die Reduzierung der Anzahl der Speicherzuweisungen [58] während der Laufzeit versucht DDS, seine Leistung zu verbessern. Der Datenverteilungsservice verwendet ein brokerloses Publish/Subscribe-Modell. Es gibt demnach keinen Broker als zentrale Vermittlungsinstanz zwischen Publisher und Subscriber. Das Design bietet viele Vorteile, wie Fehlertoleranz und Skalierbarkeit. Darüber hinaus benötigen die Anwendungen keine Informationen über die andere teilnehmende Anwendung. Der Hauptzweck von DDS ist die Bereitstellung von Punkt-zu-Punkt-Verbindungen zwischen Geräten. Wie in Abbildung 7 zu sehen ist, arbeitet DDS in einem Edge-Netzwerk. Ein Edge-Netzwerk befindet sich an den logischen Extremen eines Netzwerks. Um die Kommunikationsbandbreite zwischen Sensoren und dem zentralen Rechenzentrum zu reduzieren, werden Rechenleistung und Dienste direkt in Edge-Geräte geleitet, die sich in diesen Edge-Netzen befinden. Anwendungsspezifische Datentypen, sogenannte Topics, werden auf den Geräten gehostet und Daten werden ohne den Einsatz eines Brokers veröffentlicht und untereinander abonniert. Ein Broker kann ein zentrales Element in einem Nachrichtenparadigma sein, das die empfangenen Nachrichten der Publisher verwaltet und an die jeweiligen Subscriber sendet. Der datenzentrierte Middleware-Standard ist fernzugriffsfähig und kann Millionen von Nachrichten pro Sekunde an viele gleichzeitige Teilnehmer veröffentlichen. DDS wird in Technologien wie industriellen Embedded-Anwendungen, militärischen Anwendungen und der Weltraumforschung eingesetzt [59].

Die DDS-Spezifikation ist in mehrere Teile gegliedert und beschreibt die bereitgestellten Dienste in der Unified Modeling Language (UML) auf Basis des Platform Independent Model (PIM). Das PIM gewährleistet die Portabilität von Implementierungen in jeder Programmiersprache und auf jedem Betriebssystem. Die DDS Core Spezifikationen beinhalten die Data-Centric Publish-Subscribe (DCPS)-Schicht und das Real-Time Publish-Subscribe (RTPS)-Protokoll. Das DCPS spezifiziert die Datenverteilungsarchitektur, während RTPS das Protokoll ist, das für den Transport der Daten verantwortlich ist.

3.1.3.2 Ebene

Der Datenverteilungsdienst arbeitet auf der Anwendungsebene. Das DDS Real-Time Publish-Subscribe (RTPS) Protokoll bietet Interoperabilität zwischen Implementierungen, Plattformen und Programmiersprachen wie C, C++, C# und Java. Es ist polyglott und plattformunabhängig [58], [60].

3.1.3.3 Standardisierung

DDS wurde von der OMG [60] standardisiert und ist in vielen großen Anwendungen, die eine verteilte Kommunikation erfordern, weit verbreitet, einschließlich Gesundheitswesen, Militär, Industrie und öffentliche Infrastrukturen.

3.1.3.4 Architektur

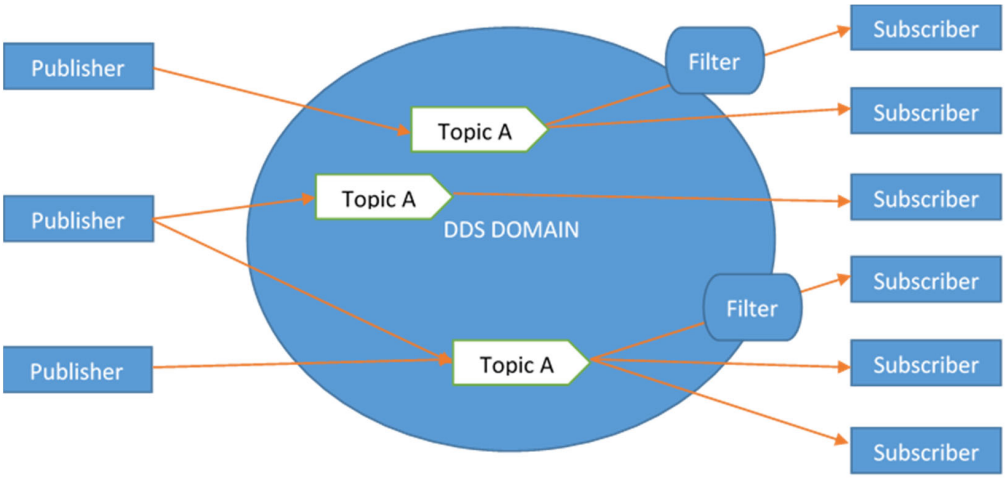


Abbildung 8: Eine DDS-Domäne im globalen Datenraum

Die DDS-Architektur ist dezentralisiert, um eine hohe Zuverlässigkeit und Datenkonnektivität mit geringer Latenz für geschäftskritische IoT-Anwendungen zu gewährleisten. Daten in DDS werden in einem globalen Datenraum aus DDS-Domänen bereitgestellt (Abbildung 8). Einzelne Anwendungen speichern ihre erforderlichen Daten lokal zwischen, während der Rest in entfernten Knoten gehalten und über APIs wie im nativen Speicher der Anwendung abgerufen wird. Basierend auf den Prinzipien des Publish/Subscribe-Paradigmas werden Daten gelesen oder in Topics geschrieben. DDS-Domänen teilen den Datenraum logisch auf, um die Kommunikation und Adressierung zu optimieren. Teilnehmer können sich zu Topics innerhalb einer Domäne anmelden und erhalten jede Nachricht, die von anderen Teilnehmern zu diesem Topic veröffentlicht wird.

Aus Sicht einer Anwendung hat sie direkten lokalen Zugriff auf den gesamten globalen Datenraum, während in Wirklichkeit relevante Daten bei Bedarf aus der Ferne abgerufen werden. Der globale Datenraum ist somit eine Sammlung von lokalen Anwendungsspeichern, die allen Teilnehmern auf Peer-to-Peer-Basis zur Verfügung stehen. Alle Geräte sind im Wesentlichen an einen DDS-Bus angeschlossen (Abbildung 8) und kommunizieren miteinander, ohne dass eine Cloud oder ein Broker benötigt wird. Diese dezentrale Architektur ermöglicht es Systemen, in Echtzeit zu kommunizieren und gleichzeitig eine große Anzahl von Teilnehmern zu skalieren, von Edge- bis hin zu Cloud-Netzwerken.

DDS bietet eine dynamische Ermittlung von Publishern und Subscribern. Die dynamische Erkennung macht DDS-Anwendungen erweiterbar, die Endpunkte für die Kommunikation werden automatisch erkannt und konfiguriert [61]. Einzelne DDS-Instanzen können als lose gekoppelte interoperable Dienste betrachtet werden, die eine serviceorientierte Architektur bilden. Aufgrund der Art des verwendeten Publish/Subscribe-Paradigmas und des Fokus auf Daten und Datenbeziehungen, die das Gefüge von Geschäftsprozessen bilden, kann DDS jedoch als ereignisgesteuerte Architektur betrachtet werden

3.1.3.5 Nachrichten Paradigma

Der Data Distribution Service verwendet in erster Linie ein Publish/Subscribe-Modell und bietet QoS-gesteuerten Datenaustausch. Die Anwendungen kommunizieren, indem sie Topics publishen und subscriben, die durch den Namen ihrer Topics identifiziert werden. Die Publisher erklären die Topics, zu denen sie ihre Daten veröffentlichen wollen, und die Datenkonsumenten subscriben die Topics von Interesse. Die Subscriber können Zeit- und Inhaltsfilter für den Datenempfang festlegen. Alle Subscriber, die ein bestimmtes Topic abonniert haben, erhalten die Daten bei der Veröffentlichung. Der Globale Datenraum identifiziert Daten, die im System zirkulieren, und bietet Datenisolierungsmechanismen zur Verbesserung der System-Skalierbarkeit. Das Data-Centric Publish/Subscribe (DCPS) ist eine Low-Level-Schicht, die eine effiziente Bereitstellung von Daten ermöglicht, die von den Publishern an die Subscriber weitergegeben werden [62], [63].

3.1.3.6 Echtzeitfähigkeit

DDS bietet harte Echtzeitkommunikation. Es ermöglicht einen vorhersehbaren und deterministischen Datenaustausch zwischen Anwendungen, indem es den Benutzern die Kontrolle über das Echtzeitverhalten des Systems ermöglicht. QoS kann spezifiziert werden, um Timing, Priorität des Kommunikationskanals und Ressourcenauslastung zu steuern [64].

3.1.3.7 Servicequalität

DDS bietet einen umfangreichen Satz von Quality of Service (QoS)-Richtlinien, um das Verhalten der Kommunikation anzupassen. Über zwanzig QoS-Richtlinien können einzeln oder gemeinsam verwendet werden, um eine Vielzahl von Kommunikationsaspekten zu beeinflussen, einschließlich Zuverlässigkeit, Leistung, Datenpersistenz und Sicherheit [65]. Die Stärke von DDS liegt darin, all diese Funktionalitäten gleichzeitig, bei extrem hohen Geschwindigkeiten und in sehr dynamischen, anspruchsvollen und unvorhersehbaren Umgebungen bereitzustellen [66], [61].

3.1.3.8 Sicherheit

DDS bietet standardisierte Authentifizierungs-, Verschlüsselungs-, Zugriffskontroll- und Protokollierungsfunktionen, um eine sichere End-to-End-Datenverbindung in einem verteilten System zu ermöglichen [67] DDS sieht folgende Punkte für eine sichere Kommunikation vor:

- Vertraulichkeit der Datenproben
- Integrität der Datenproben und der Nachrichten, die sie enthalten
- Authentifizierung von DDS-Schreibern und Lesern
- Autorisierung von DDS-Schreibern und Lesern
- Unleugbarkeit der Daten

Alle kommunizierenden Parteien, einschließlich Geräte und Benutzer, werden separat authentifiziert. Die Implementierung der Zugriffskontrolle ermöglicht es, das Publishen oder Subscriben

bestimmter Topics im DDS-Netzwerk zu erlauben oder zu verhindern. Die Sicherheitsimplementierungen in DDS befinden sich oberhalb der Transportschicht, die Verwendung von TLS/SSL oder DTLS ist zwar möglich, aber nicht erforderlich. Außerdem wird der Funktionsumfang von DDS durch die Verwendung von TLS beeinträchtigt, beispielsweise würde es eine Echtzeit- und Multicast-Kommunikation verhindern. Die Authentifizierungsimplementierung wird verwendet, um alle Teilnehmer wie Benutzer und Geräte zu authentifizieren. Die Verschlüsselung von Daten, die sich über das DDS-Netzwerk bewegen, ist wichtig, erhöht aber Bandbreite und Overhead, so dass der Entwickler die Möglichkeit hat, die zu verschlüsselnden Daten auszuwählen oder nicht [58].

3.1.3.9 Physikalische Anforderungen

DDS hat einen geringen Speicher- und Bandbreitenbedarf und wurde entwickelt, um eine Echtzeitkommunikation zwischen Geräten zu ermöglichen. Es gibt Implementierungen wie Vortex Lite Webcast, die sich ganz auf die Optimierung von DDS für Embedded-Umgebungen konzentrieren [68].

3.1.3.10 Verfügbare Implementierungen

DDS ist ein offener Standard. Es gibt verschiedene Open-Source- und kommerzielle proprietäre Implementierungen [69]. Während die Implementierung der DCPS-Architektur frei verfügbar ist, ist die DDS Security-Implementierung zum jetzigen Zeitpunkt nur kommerziell verfügbar.

3.1.4 Weitere Protokolle

Im Rahmen des Projektes wurden alle in Tabelle 4 aufgelisteten Protokolle wie in den vorangehenden Kapiteln beschrieben und auf ihre Eignung für einen Einsatz im Projekt untersucht. Die vollständige Analyse aller Protokolle steht in einem separaten Dokument [70] zur Verfügung.

3.2 AP 2 Referenzarchitektur

Unter Berücksichtigung der Ergebnisse aus AP1 werden in diesem Kapitel zwei verschiedene Referenzarchitekturen für ein verteiltes (Automatisierungs-)System auf Grundlage von Ergebnissen aus Arbeitskreisen und dem aktuellen Stand der Forschung beschrieben. Aus diesen Beschreibungen werden Anforderungen an eine Referenzarchitektur abgeleitet. Anschließend werden zusätzliche Anforderungen anhand von industrietypischen Anwendungsfällen im Sinne der Industrie 4.0 definiert. Die Zusammenstellung dieser Anforderungen dient als Basis für das zu entwickelnde IT-Sicherheitskonzept.

3.2.1 Das Referenzarchitekturmodell für Industrie 4.0 (RAMI 4.0)

Das Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0) (Abbildung 9) ist ein dreidimensionales Schichtenmodell und gilt als Stand der Technik in Bezug auf die industrielle Produktion als Anwendungsbereich im Kontext von Industrie 4.0 [71]. In RAMI 4.0 wird auf eine eindeutige Aufteilung zwischen der physischen Welt und der Informationswelt gesetzt. Der Asset Layer repräsentiert die Assets der physischen Welt und der Integration Layer repräsentiert die digitalen Anteile in der Informationswelt. Mit Hilfe der drei Achsen von RAMI 4.0 können alle wesentlichen Elemente im Lebenszyklus eines Assets beschrieben werden.

Die drei Achsen Layer, Life Cycle & Value Stream und Hierarchy Levels werden im Folgenden ausführlich beschrieben:

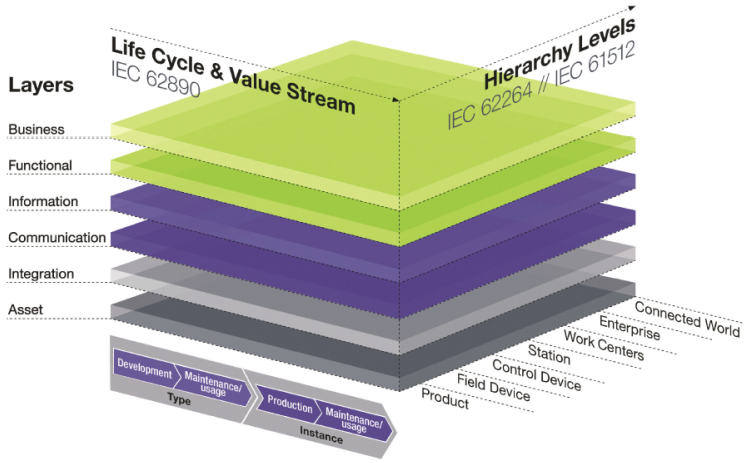


Abbildung 9: Referenzarchitektur für Industrie 4.0 [71]

Die Referenzarchitektur für Industrie 4.0 (RAMI4.0) dient der Darstellung von Sachverhalten in der produktionstypischen Umgebung, um die Standardisierung von zu nutzenden Protokollen voranzutreiben. Dabei steht die industrielle Produktion, von der diskreten Fertigung bis hin zur Prozesstechnik, im Fokus. Damit distanziert sich RAMI 4.0 im Bezug zur Industrie 4.0 vom weiter gefassten Internet-of-Things-Ansatz, welchen das Industrial Internet Consortium (IIC) für die Industrial Internet Reference Architecture (IIRA) nutzt.

3.2.1.1 Hierarchie Levels – Achse

Auf der rechten horizontalen Achse sind die Hierarchiestufen angeordnet. Die Hierarchie-Level-Achse besteht im Wesentlichen aus den Ebenen der Automatisierungspyramide, welche im Rahmen des Themenkomplex Industrie 4.0 um neue Ebenen ergänzt wird. Bei dieser funktionalen Zuordnung der Hierarchien steht das „Product“ bzw. „Smart Product“ nun an unterster Stelle, da Produkte im Kontext von Industrie 4.0 ihren eigenen Fertigungsprozess beeinflussen können [72]. Oberhalb der Automatisierungspyramide ist gegenüber bestehenden Definitionen die Hierarchieebene „Connected World“ ergänzt, um die Vernetzung von Firmen im Kontext von Industrie 4.0 einzuordnen. Die Hierarchie-Achse ist in Abbildung 10 dargestellt.

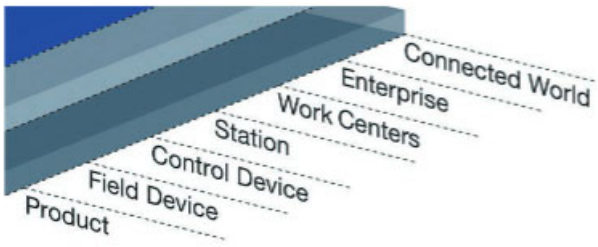


Abbildung 10: Hierarchie-Achse [72]

3.2.1.2 Architektur-Achse (Layer)

Die vertikale Achse des RAMI 4.0 ist die Architektur-Achse. Die Architektur-Achse ist in sechs Schichten aufgeteilt, in denen das digitale Abbild eines Assets (beispielsweise einer Maschine)

Schicht für Schicht beschrieben wird. In Abbildung 11 wird die Architektur-Achse und zusätzlich die Zugehörigkeit zur realen Welt oder Informationswelt der verschiedenen Architektur-Schichten dargestellt.

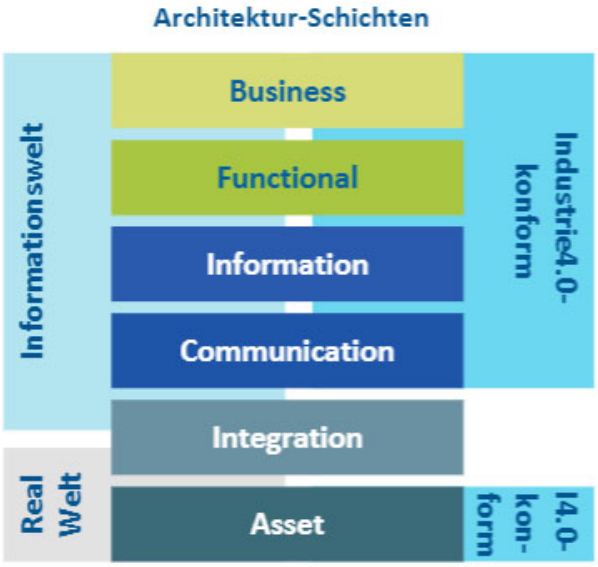


Abbildung 11: Aufteilung Informationswelt / physische Welt [72]

Asset- und Integration-Layer

Der Asset-Layer und der Integration-Layer sind im RAMI 4.0 beschrieben. Der Asset-Layer ist die unterste Schicht im RAMI 4.0 und repräsentiert die Assets der physischen Welt, wie beispielsweise eine Maschine in einer Produktionsumgebung. Der Integration-Layer ist die nächst höhere Schicht im RAMI 4.0 und fungiert außerdem als Bindeglied zwischen der physischen Welt eines Assets und der Informationswelt von Industrie 4.0. Das bedeutet, dass der Integration-Layer im Zusammenhang mit bisherigen Industrie 3.0 Technologien steht, die vier darüber liegenden Schichten jedoch Industrie 4.0 – konform agieren.

Communication Layer

Der Communication Layer ist oberhalb des Integration Layer angesiedelt. Im Kontext der Entwicklungen im Zusammenhang des Themenkomplexes Industrie 4.0 ist das Ziel entstanden, dass jedes Asset im Verbund mit einer Verwaltungsschale, als Industrie 4.0 – Komponente, mit jedem sich im Netzwerk befindlichen Asset kommunizieren und Informationen austauschen kann. Dabei ist es von keiner entscheidenden Bedeutung zu welcher hierarchischen Ebene das Asset zugeordnet wird. Um dieses Ziel zu erreichen, ist eine erfolgreiche Kombination von industrieller Kommunikation und Lösungen aus der Informationstechnik erforderlich [72], um eine Umsetzung der Industrie 4.0 – Kommunikation voranzutreiben.

In der Abbildung 12 sind die von der Plattform Industrie 4.0 gegenwärtigen Festlegungen im Communication Layer dargestellt. Dieses Modell beschreibt den Communication Layer des RAMI 4.0

im Bezug zu der Life-Cycle- und Value-Stream-Achse und der Hierarchie-Level-Achse.

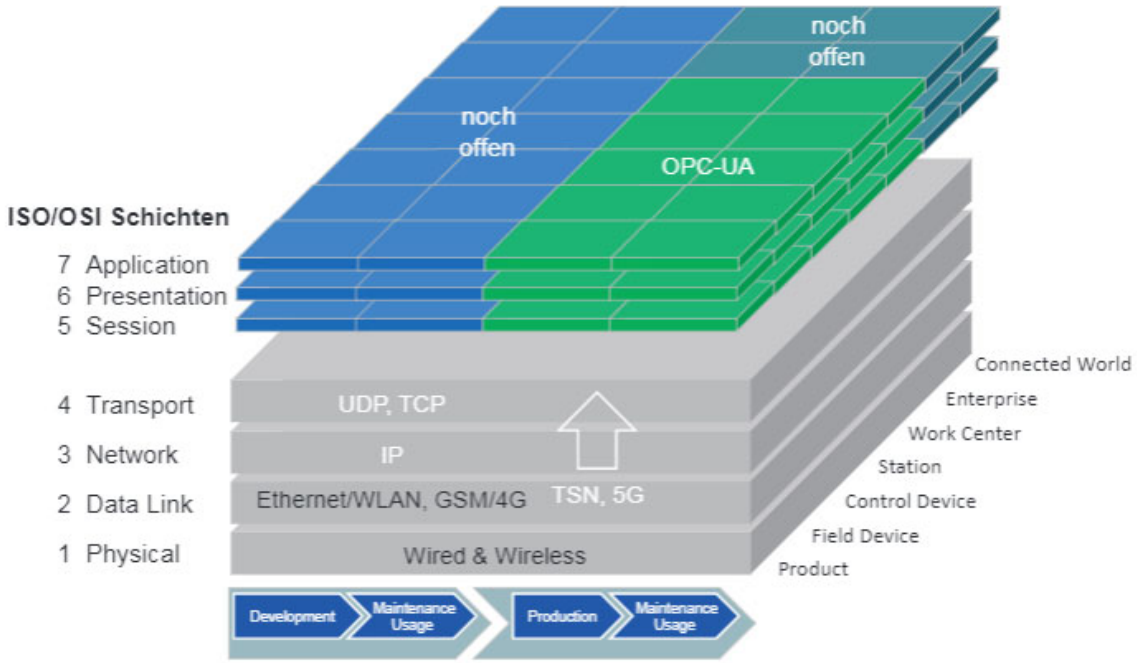


Abbildung 12: Gegenwärtige Festlegungen im Communication Layer [72]

Kommunikationssysteme für die Übertragung von echtzeitkritischen Prozessdaten, wie PROFIBUS oder PROFINET, sind derzeit nicht als gegenwärtige Festlegungen im Communication Layer beschrieben. Diese Systeme siedeln sich im Integration Layer an, da sie noch keine I4.0 – konformen Kommunikationseigenschaften besitzen [72].

Information Layer

Der Functional-Layer beschreibt die verschiedenen anwendbaren Funktionen und Services in einem Prozess der Wertschöpfung. Dabei dient der Functional-Layer als Plattform für die horizontale Integration der Funktionen und für die Erzeugung von Ereignissen und Regeln auf Grundlage des Information Layer [73]. Die Funktionen beziehen ihre Informationen in Form von Daten vom Information Layer und legen ihre Ergebnisse als Industrie-4.0-Daten im Information Layer ab [72]. Zur Sicherstellung der Integrität der Informationen kann ein Fernzugriff und horizontale Integration ausschließlich innerhalb des Functional-Layer durchgeführt werden.

Business Layer

Der Business Layer enthält die Geschäftslogik der Industrie 4.0 Anwendung und beschreibt die geschäftlichen Rahmenbedingungen eines Assets. Der Business Layer modelliert die zu folgenden Regeln des Systems und dient als Verbindungselement zwischen den verschiedenen Geschäftsprozessen [73]. Dabei wird sich nicht auf konkrete Systeme wie das Enterprise-Ressource-Planning (ERP) bezogen, sondern lediglich ERP-Funktionen benutzt, die sich im Functional-Layer wiederfinden [73].

3.2.1.3 Life Cycle & Value Stream-Achse

Die linke horizontale Achse beschreibt den Lebenszyklus von Assets auf Grundlage der der IEC 62890 zum Life-Cycle-Management [73]. Der Lebenszyklus von Assets wird dabei in zwei (Lebens-) Phasen unterteilt. Die erste Phase „Typ“ steht sinnbildlich für die Entwicklung eines Assets. Während dieser Phase im Lebenszyklus eines Assets werden alle relevanten Informationen des Assets gesammelt, um mit Produktion beginnen zu können. Im Vergleich zur Softwareentwicklung könnte man hier auch die Definition einer Klasse nennen. Die zweite Phase ist die „Instance“ Phase und schließt direkt an die erste Phase an. Ein Asset kommt in die zweite Phase, wenn die Entwicklung beendet wurde und die Herstellung beginnen kann. Diesen beiden beschriebenen Phasen werden jeweils zwei weitere Phasen zugeordnet, um den Lebens-zyklus eines Assets vollumfänglich beschreiben zu können.

Abbildung 13: Die "Life Cycle & Value Stream" Achse des RAMI 4.0 [72]

Die Abbildung 13 beschreibt die Achse „Life Cycle & Value Stream“ des RAMI 4.0 und stellt die zwei beschriebenen Phasen mit ihren Unterteilungen in jeweils zwei weitere Phasen im Lebenszyklus eines Assets dar. Die „Development“ Phase lässt sich der „Typ“ Phase zuordnen und steht für die Entwicklung eines Assets. In der „Maintenance Usage“ Phase, die ebenfalls zur „Typ“ Phase gehört, und beschreibt die Wartungsnutzung des Assets. Die „Production“ Phase ist die Phase im Lebenszyklus eines Assets, in der Instanzen mit relevanten Produktionsinformationen, z.B. bezüglich des Materials eines Assets entstehen. Die erneute „Maintenance Usage“ Phase ist in der „Instance“ Phase angesiedelt.

3.2.2 Die Industrial Internet Reference Architecture (IIRA)

Die Industrial Internet Reference Architecture (IIRA) [74] des Industrial Internet Consortiums (IIC) ist eine Referenzarchitektur aus dem Jahr 2015 mit der Orientierung auf Geschäftsprozesse und logische Aufbauten von Gesamtsystemen im Industrial Internet of Things Umfeld zu beschreiben. Grundlage für die Entwicklung der Industrial Internet Reference Architecture ist die ISO/IEC/IEEE 42010 [75]. Auf Basis der Beschreibung von verschiedenen Abstraktionsebenen und semantischen Zusammenhängen, wird im Kern der IIRA zwischen vier geschäftlichen und technischen Sichtweisen (engl. Viewpoints) unterschieden. Die Industrial Internet Reference Architecture des IIC dient der grundlegenden Definition von Ende-zu-Ende Anwendungssysteme für industrielle

Aufgaben, die als Industrial Internet Systems (IIS) bezeichnet werden [76]. In Abbildung 14 ist die IIRA dargestellt.

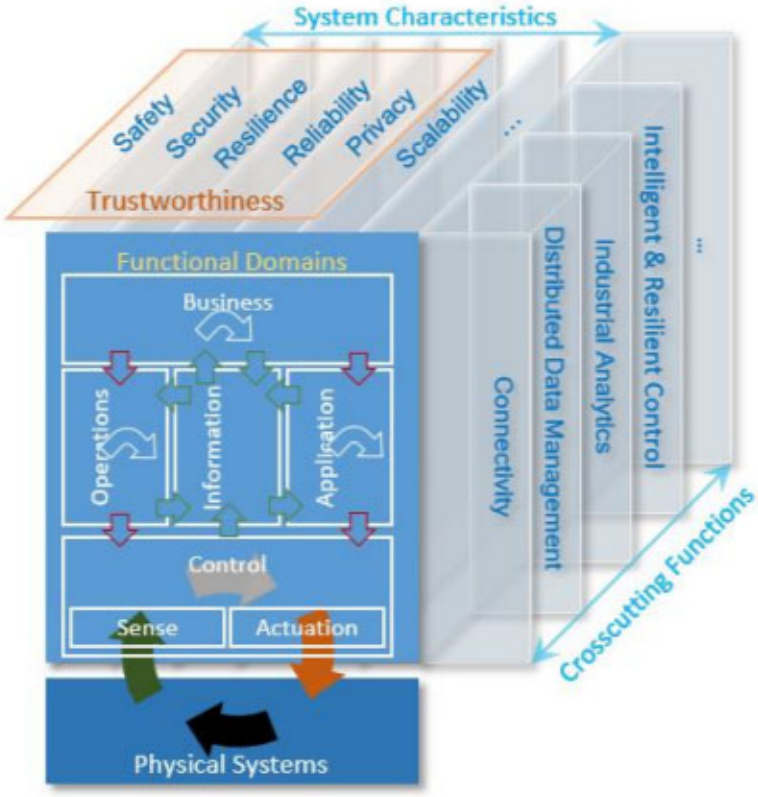


Abbildung 14: Die Industrial Internet Reference Architecture (IIRA) [76]

Die IIRA besteht aus einer eindimensionalen Schicht, die die funktionellen Bereiche (engl. functional domains) eines industriellen Internet Systems darstellt. Das IIRA wird in vier sogenannte Standpunkte (engl. Viewpoints) unterteilt (Abbildung 15).

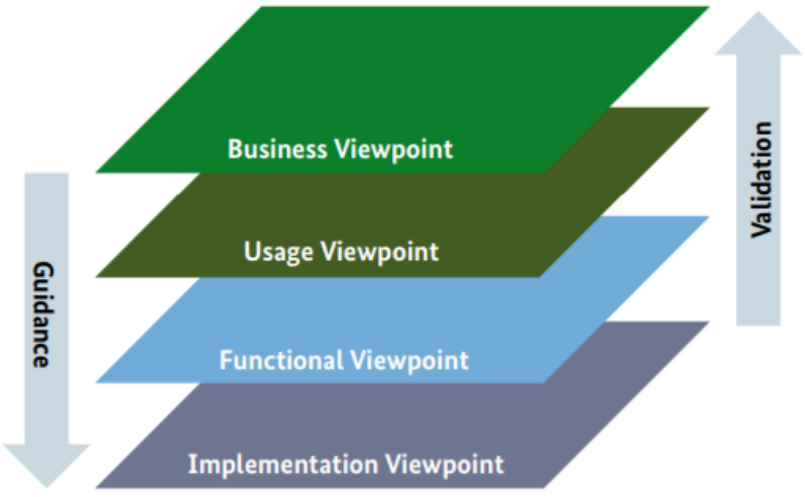


Abbildung 15: Die vier Viewpoints der IIRA [74]

Dabei handelt es sich um den Business-Viewpoint, den Usage-Viewpoint, den Functional-Viewpoint und den Implementation-Viewpoint. Die vier verschiedenen Standpunkte werden im Folgenden erläutert: Der Business-Viewpoint beschreibt die Identifikation von Stakeholdern und deren wirtschaftlichen Ziele und Zusammenhänge bei der Entwicklung eines Industrial Internet Systems in seinem geschäftlichen Umfeld. Durch die Definition von grundlegenden Systemfunktionen durch die Entscheidungsträger (z.B. Produktmanager), kann das Industrial Internet System geforderte Ziele erreichen. Der Usage-Viewpoint beschreibt in seiner Betrachtungsweise die Nutzung eines IIS. Die Funktionalitäten und Systemfähigkeiten werden in Form von Bedienabläufen oder Aktivitätsabfolgen beschrieben. Der Functional-Viewpoint setzt den Fokus auf die funktionalen Komponenten in einem Industrial Internet System. Dabei wird sich vor allem auf die Interaktionen zwischen IIS und externen Elementen in der Umgebung konzentriert. Zudem ist die Betrachtung der Struktur von Industrial Internet Systemen und ihren Schnittstellen von Bedeutung, um die möglichen Nutzungen und Aktivitäten des IIS als Gesamtsystem zu erkennen und zu unterstützen. Der Implementation Viewpoint befasst sich mit der Implementierung von Komponenten und Kommunikationssystemen. Diese Komponenten sind durch Aktivitäten (Usage-Viewpoint) koordiniert und unterstützen die Systemfähigkeiten (Business-Viewpoint) [76].

3.2.3 Anforderungen an die Referenzarchitektur

Im Folgenden werden die Anforderungen beschrieben, die an die IT-Security Referenzarchitektur in Industrie 4.0 Umgebung gestellt werden [76].

- Skalierbarkeit: Es muss gewährleistet sein, dass sich auf Basis der Referenzarchitektur Cyber-physische Systeme (CPS) in verschiedenen Größenordnungen, d.h. auch Geräte mit geringer Rechenleistung, benutzen lassen.
- Echtzeitfähigkeit: Es muss gewährleistet sein, dass eine Klassifizierung bezüglich zeitlicher Anforderungen von industriellen Prozessen durch die Referenzarchitektur abgebildet werden kann. Diese Klassifizierung trägt dazu bei, dass Prozesse mit Echtzeit-Anforderungen priorisiert kommunizieren können, um die Verfügbarkeit des Systems zu garantieren. Echtzeitfähigkeit gilt nicht als zwingende Anforderung, wird hier jedoch trotzdem aufgeführt, da es bereits echtzeitfähige Middleware-Protokolle und Forschungsansätze für die Echtzeitfähigkeit von Middleware-Protokollen (z.B. OPC UA mit TSN [77]) gibt.
- Interoperabilität: Die Geräte in dem Automatisierungssystem, z.B. Cyber–physischen Systeme, müssen auf offenen Kommunikationsstandards beruhen. Spezialsysteme, die z.B. für harte Echtzeitaufgaben benutzt werden und keine offene Kommunikation erlauben, müssen über Gateways in eine offene Kommunikationsarchitektur eingebunden werden können. Damit wird eine einheitliche Kommunikation im System gewährleistet, in der Geräte und Anlagenteile verschiedener Hersteller integriert werden können.
- Informationssicherheit: Der Schutz von Informationen ist in der Referenzarchitektur durch geeignete Mechanismen und Konzepte sicherzustellen. Es ist unbedingt nötig, unerlaubte Nutzung oder auch Manipulation von Daten zu unterbinden.

Anforderungen an die Referenzarchitektur im Kontext von Industrie 4.0

Im Folgenden werden allgemeine Anforderungen im Kontext von Industrie 4.0 beschrieben [76]:

- Informationssicherheit: Der Schutz von Daten ist in der Referenzarchitektur durch geeignete Mechanismen und Konzepte sicherzustellen. Es ist unbedingt nötig, unerlaubte Nutzung oder auch Manipulation von Daten zu unterbinden.

- Safety (funktionale Sicherheit): Es muss gewährleistet sein, dass das System bei Auftreten zufälligen und auch systematischen Ausfällen mit gefahrbringender Wirkung, einen sicheren Zustand einnimmt.
- Skalierbare Integrierbarkeit: Skalierbarkeit wird durch das Integrieren von Cyber–physischen Systemen in das System erreicht. Beispielsweise in Bezug auf das Verhalten des Systems und den Datendurchsatz.
- Interoperabilität: Die Geräte in dem Automatisierungssystem, z.B. Cyber–physischen Systeme, müssen auf offenen Kommunikationsstandards beruhen. Spezialsysteme, die z.B. für harte Echtzeitaufgaben benutzt werden und keine offene Kommunikation erlauben, müssen über Gateways in eine offene Kommunikationsarchitektur eingebunden werden können. Damit wird eine einheitliche Kommunikation im System gewährleistet, in der Geräte und Anlagenteile verschiedener Hersteller integriert werden können.
- Transparente Vernetzung: Die Vernetzung der Cyber–physischen Systeme im System muss transparent sein. Das heißt, dass die geschaffenen Kommunikationsbeziehungen bekannt sind.
- Integrierte zuverlässige Datenverwaltung: Gesammelte Daten des Systems sind zuverlässig zu speichern und zu verwalten.
- Erweiterte Fähigkeiten der Datenanalyse: Das System muss die Fähigkeit besitzen, die gesammelten Daten zu analysieren.
- Robuste Reaktion: Automatisierungskomponenten müssen bei Beanspruchung stabil sein und wichtige Funktionen müssen auch bei erhöhter Beanspruchung erhalten bleiben.
- Ad-hoc-Fähigkeiten und Plug-and-play: Das Automatisierungssystem besitzt die Fähigkeit eine drahtlose und drahtgebundene Netzwerktopologie zwischen mehreren Endgeräten aufzubauen, die ohne feste Infrastruktur auskommt. Das System muss in der Lage sein, andere Geräte in die Kommunikation aufzunehmen, ohne die Installation von Gerätetreibern durchzuführen, das Netzwerk zu konfigurieren oder die Änderung an Einstellungen vorzunehmen.
- Security by Design: Konzepte und Funktionen der IT-Security für die Automation sind bei der Entwicklung von automationstechnischen Komponenten und Lösungen zu berücksichtigen. Die Konzepte Security by Default, Security by Implementation und Security by Deployment werden vorausgesetzt.

3.2.4 Anforderungen definiert anhand von Anwendungsfällen

In Abbildung 16 ist, zusätzlich zu dem Produktionsbereich der Industrie 4.0 – konformen Anlagenstruktur, der Office Bereich dargestellt. Im Office Bereich befindet sich ein diagnosefähiges Gerät; außerdem verfügt diese Anlagenstruktur über eine unternehmenseigene Cloud (private Cloud). Der erste betrachtete Anwendungsfall beschreibt die Industrie 4.0-Kommunikation eines Aktors aus der Feldebene mit einem Diagnose-PC im Office Bereich (Abbildung 16, „Anwendungsfall 1“). Der zweite Anwendungsfall beschreibt die Middleware-Kommunikation eines Aktors aus der Feldebene bis hin zur Unternehmens-Leitebene oder möglicherweise sogar bis zur Unternehmens-Cloud (Abbildung 16, „Anwendungsfall 2“). Beide Anwendungsfälle betrachten ein Feldgerät als Kommunikationspartner. Ist die Industrie 4.0-Kommunikation mit einem Feldgerät als Kommunikationspartner realisierbar, so ist es mit großer Wahrscheinlichkeit auch mit Geräten als Kommunikationspartner möglich, die größere Datenmengen transportieren können. Die beiden Anwendungsfälle schließen die Betrachtung von funktionaler-Sicherheit (Safety) zunächst aus. Als dritten Anwendungsfall wird die Kommunikation zwischen einem externen Unternehmen (Unternehmen 2) und der bereits in Anwendungsfall 2 beschriebenen Cloud von Unternehmen 1

betrachtet (Abbildung 16, „Anwendungsfall 3“). Bei dem Unternehmen 2 könnte es sich um einen Zulieferer von Feldgeräten für das Unternehmen 1 handeln, der Zugriff auf Statusdaten der Messumformer in der Cloud von Unternehmen 1 erhält, um Informationen für mögliche Wartungseinsätze zu erhalten. Die vorherige private Cloud ändert sich nun zu einer „Shared Cloud“, damit ein Zugriff vom Unternehmen 2 gewährleistet werden kann.

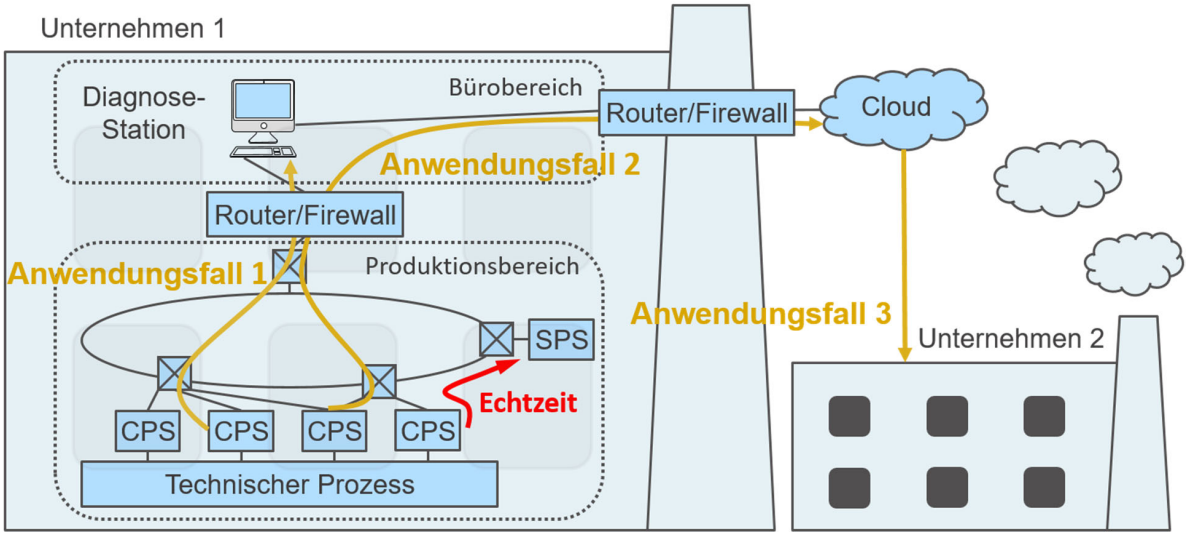


Abbildung 16: Anwendungsfälle 1-3

3.2.4.1 I4.0-Kommunikation mit einem Diagnosegerät (Anwendungsfall 1)

Die ersten Anforderungen ergeben sich bei der Betrachtung eines Feldgerätes (Sensor / Aktor) als Kommunikationspartner (Abbildung 16). Der Einsatz von Cyber-physischen Systemen (CPS) mit geringen Rechenleistungen und intelligenter Sensorik/Aktorik ist zu berücksichtigen [78]. Da die Sensor-/Aktor-Komponenten nun über eine direkte Industrial Ethernet Anbindung verfügen, gelten für sie nun die gleichen Bedrohungen, wie für Remote I/O und Steuerungen. Die Geräte mit geringen Rechenleistungen dürften keine zu großen Mengen an Daten erhalten. Ein Load Balancer könnte eingesetzt werden, um eine Lastverteilung zu ermöglichen. Dabei wird die zu erbringende Rechenlast gleichmäßig auf die entsprechenden Geräte verteilt, sodass einzelne Geräte nur eine bestimmte Menge an Datenpaketen pro Zeiteinheit erhalten. Bei Überlastung muss das Gerät robust reagieren und in der Lage sein, qualifiziert Datenpakete geringer Wichtigkeit zu verwerfen, um die stetige Verfügbarkeit der Anlage zu gewährleisten. Vor allem muss gewährleistet sein, dass die Echtzeitkommunikation stets gegenüber der Nicht-Echtzeit-Kommunikation priorisiert wird. Hochverfügbare Systemstrukturen sind zu berücksichtigen. Allgemein kommt es darauf an, eine hohe Verfügbarkeit bei gleichzeitiger Gewährleistung der IT-Sicherheit der industriellen Kommunikation zur Minimierung von Produktionsausfällen zu erreichen [79]. Schutzmaßnahmen gegen Bedrohungen, wie Distributed Denial-of-Service-Angriffe (DDOS) könnten mit der Installation von Abwehrfunktionen oberhalb des Data Link Layers verbunden sein. Mit der Gewährleistung einer sicheren Kommunikation wird eine weitere nicht-funktionale Anforderung definiert. Dazu gehört vor allem die Identifizierung und Authentifizierung von allen Netzwerkteilnehmern innerhalb und außerhalb des Automatisierungsnetzes, sowie die allgemeine Nutzungskontrolle [80]. Um neben der sicheren Kommunikation die Identifizierung und Authentifizierung von allen Netzwerkteilnehmern im Automatisierungsnetz zu gewährleisten, muss das Automatisierungssystem an allen Schnittstellen die Fähigkeit haben, die allen Nutzern

(menschlicher Nutzer, Softwareprozesse oder Geräte) zugewiesenen Nutzerrechte durchzusetzen [81]. Um die Systemintegrität zu gewährleisten, muss das Automatisierungssystem außerdem in der Lage sein, Schutzvorkehrungen einzusetzen, um übertragenen Schadcode zu erkennen und dessen Ausführung zu verhindern [79]. Versucht eine Anwendung zu starten, kann mit Hilfe von Whitelisting überprüft werden, ob diese Anwendung über die Berechtigung dazu verfügt. Das Automatisierungssystem muss die Fähigkeit besitzen, die Vertraulichkeit von Informationen und Daten bei Übertragung über nicht vertrauenswürdige Netze zu schützen [81]. Weitere Kernanforderungen ergeben sich zum Beispiel aus den Arbeiten der NAMUR-Interessengemeinschaft Automatisierungstechnik der Prozessindustrie e.V. In diesen Ergebnissen werden neben der Vorgabe, dass die Nutzbarkeit von neuen Konzepten sowohl für bestehende Produktionsanlagen, als auch für neue Produktionsanlagen zu berücksichtigen ist, eine agile Umsetzung auf bestehenden Standards, Normen und Richtlinien gefordert [82]. Außerdem werden Anwendbarkeit, Reduktion der Komplexität und Wirtschaftlichkeit als wesentliche Erfolgsfaktoren gesehen [81].

Die NAMUR benennt in dem Paper „NAMUR Open Architecture“ [83] Automation Security als integralen Bestandteil neuer Konzepte und Lösungen und verweist dabei auf das Security-by-Design-Prinzip [84], durch das noch detailliertere Anforderungen definiert werden. So sollen Konzepte zur Verfügung gestellt werden, die helfen die Komponenten im Laufe der Lebensdauer an veränderte Bedrohungs- und Verwundbarkeitssituationen zu adaptieren. Werden externe IT-Security Maßnahmen eingesetzt, müssen auch diese unter den vorhersehbaren Betriebsbedingungen beschrieben und getestet werden. Die definierten Leistungsmerkmale der automatisierungstechnischen Funktionen müssen stets nachweislich aufrechterhalten werden [83]. Werden ergänzende Maßnahmen eingesetzt, so sind die Aufwände für die Integration ergänzender Sicherheitsmaßnahmen bei erhöhtem Schutzbedarf zu minimieren. Die Qualität der Anwendungssoftware von Automatisierungskomponenten und –lösungen beeinflussen maßgeblich die IT-Security in der Automation. State-of-The-Art Methoden zur Qualitätssicherung von Software sind nachweisbar anzuwenden. Die Kommunikation zwischen den Software-Modulen einer automatisierungstechnischen Komponente ist über eindeutig spezifizierte Schnittstellen, wie die Quelle und das Ziel, sicher zu stellen und zu überprüfen. Es ist ein deterministisches Zeitverhalten in der Kommunikationsbeziehung zwischen allen Modulen einer Komponente zu gewährleisten. Weiterhin werden Anforderungen an die rechtzeitige Reaktion des Systems auf Ereignisse gestellt [79]. Die VDI/VDE 2182 [85] stellt vor allem Anforderungen an die Maschinenbauer und die Hersteller. Neben der Berücksichtigung des Verhaltens bei Hardwareausfall, Stromausfall und Wiederanlauf nach Stromausfall werden verschiedene Anforderungen an die benutzten Betriebssysteme gestellt. So sollen nur erforderliche Dienste und Applikationen für den ordnungsgemäßen Betrieb ausführbar und nicht erforderliche Dienste deaktiviert sein. Außerdem sollte es möglich sein, redundante Controller und Komponenten der Infrastruktur zu implementieren, um die Erreichbarkeit zu erhöhen [86].

3.2.4.2 I4.0-Kommunikation mit der privaten Cloud (Anwendungsfall 2)

Die Bezeichnung Cloud-Computing beschreibt die Bereitstellung von IT-Infrastrukturen lokal oder über das Internet. Cloud-Computing bietet dem Anwender z.B. Speicherplatz und Dienste, die über Schnittstellen und Protokolle abgerufen werden können und nicht auf dem lokalen System installiert sein müssen. In dem BSI-Anforderungskatalog: Cloud Computing C5 [87] werden Mindestanforderungen zur Nutzung von Cloud-Diensten definiert. Es gelten unter anderem Anforderungen zur Organisation der Informationssicherheit, Kommunikationssicherheit und Kryptographie und Schlüsselmanagement. Auf diese Mindestanforderung wird in späteren Kapiteln eingegangen. Im Folgenden werden die wesentlichen Anforderungen aufgegriffen: Allgemein gilt es,

die Benutzung von Cloud-Diensten (Unternehmens-Cloud, externe Cloud) zu berücksichtigen. Zur sicheren Nutzung von Cloud-Diensten [88] gehört ein sicheres Authentisierungsverfahren, wobei bei dieser Authentisierung keine Backdoors, in Form von Universalpasswörtern zugelassen werden dürfen. Weitere Anforderungen an diese Art der Kommunikation zwischen Aktor und Cloud ergeben sich bei Betrachtung der Cloud. Es muss eine sichere Synchronisation stationärer und mobiler Geräte über Cloud-Infrastrukturen erfolgen. Die Erkennung und Bekämpfung von Schadsoftware in der Cloud muss gewährleistet sein und die Integration von Technologien zur systematischen Datennutzungskontrolle in Cloud-Infrastrukturen muss möglich sein [79].

3.2.4.3 I4.0-Kommunikation mit Shared Cloud (Anwendungsfall 3)

Es ergeben sich weitere Anforderungen für die Industrie 4.0-Kommunikation mit einer Shared Cloud (Anwendungsfall 3). Es wird eine Cloud als Kombination einer privaten Cloud und einer Community Cloud definiert. Cloud-Dienste können in diesem besonderen Fall der Shared Cloud nur den Mitarbeitern eines Unternehmens vorbehalten sein, wobei andere Dienstleistungen den Mitarbeitern beider Unternehmen zugänglich gemacht werden.

Es gelten weiterhin die Mindestanforderungen aus dem BSI-Anforderungskatalog: Cloud Computing C5 [87] zur Nutzung von Cloud-Diensten sowie die Anforderungen aus dem Anwendungsfall 2. Da es sich um die Nutzung von Cloud-Diensten einer (unternehmens-) fremden Cloud handelt werden zusätzliche Anforderungen definiert.

Das BSI empfiehlt eine Multi-Faktor Authentifizierung [89] bei Cloud-Diensten die über das Internet angebunden sind, sodass durch die Prüfung von zwei Faktoren die Identität des Anwenders bestätigt werden kann. Diese zwei Faktoren müssen unabhängig sein, zum Beispiel Hardware-Token und Passwort oder Fingerabdruck und PIN. Eine duale Verwaltung von Nutzerrechten sollte gewährleistet sein, um zwischen internen und externen Anwendern zu unterscheiden. Für das Unternehmen 2 muss ein gesonderter Account mit eingeschränkten Zugriffsrechten erstellt werden. Damit können dem Unternehmen 2 bei Bedarf nur Lese-Funktionen erlaubt werden. Des Weiteren ist ein sicheres Abrufen der Daten des Unternehmen 2 zu gewährleisten, dies könnte beispielsweise mit VPN ermöglicht werden.

3.3 AP 3 Rollenbeschreibung

Unter den Anforderungen aus Kapitel 3.2.3 wird für ein sicheres System unter anderem eine entsprechende Autorisierungsstrategie für Ressourcen aufgeführt. Zur Verwaltung und Durchsetzung entsprechender Zugriffsrechte bietet sich eine rollenbasierte Rechtevergabe („RBAC“ – Role Based Access Control“ [90]) an. Im Folgenden wird kurz auf die Komponenten von RBAC eingegangen. Anschließend folgt eine Auflistung möglicher Rollen.

3.3.1 Komponenten

In Abbildung 17 wird das Grundkonzept von RBAC mit den wesentlichen Komponenten dargestellt.

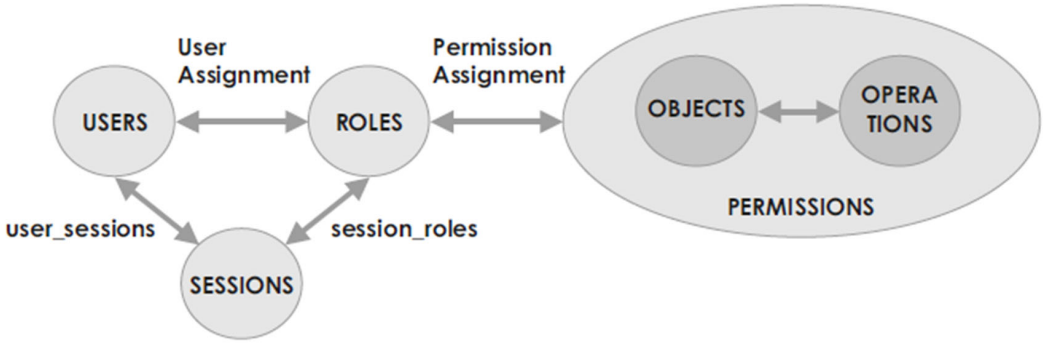


Abbildung 17: RBAC-Konzept [91]

Benutzer (Users)

Das Subjekt das auf ein bestimmtes Objekt zugreifen möchte. Es kann sich dabei neben Personen auch um IT-Systeme oder IT-Applikationen handeln.

Rollen (Roles)

Die Rolle ist das Verbindungselement zwischen dem Benutzer und der Berechtigung. Jeder Rolle werden dabei eine oder mehrere Berechtigungen zugewiesen (Permission Assignment). Die jeweilige Rolle wird einem oder mehreren Benutzern zugewiesen (User Assignment).

Sitzung (Sessions)

Jedem Benutzer werden entsprechend seiner Aufgabe eine oder mehrere Rollen zugeteilt, allerdings werden beim Login nicht automatisch alle zugeteilten Rollen aktiviert. Es wird eine Session erzeugt, die die benötigte Rolle aktiviert.

Objekte (Objects)

Objekte sind Ressourcen im Unternehmen, die vor unbefugter Nutzung und Missbrauch geschützt werden müssen. Dazu zählen physische Objekte (Räume, Fahrzeuge, Maschinen) oder logische Objekte (Daten, Softwarefunktionen).

Berechtigungen (Permissions)

Zugriffsberechtigungen bestehen aus zwei Komponenten. Dies sind die nutzbaren Ressourcen (Objects) und die Operationen (Operations), die auf diesen Ressourcen ausgeführt werden dürfen.

3.3.2 Auswahl möglicher Rollen

In Tabelle 5 werden mögliche Rollen nach [92] beschrieben. Diese Rollen wurden zunächst als Grundlage für die weitere Arbeit verwendet. Durch zusätzliche Attribute werden diese Rollen später ergänzt (s. Kapitel 3.4.3.1).

Tabelle 5: Rollenbeschreibung nach [92]

Rolle	**Beschreibung (mögliche Berechtigung)**
Anonymous	Da die Anmeldung anonym erfolgt, ist dies die Rolle mit den geringsten Berechtigungen. Beispielsweise besitzt diese Rolle lediglich Leseberechtigungen für unkritische Daten.
AuthenticatedUser	Sitzung erfolgt über authentifizierten Benutzer. Es fehlt eine konkrete Rollenzuweisung, daher sind die Berechtigungen ebenfalls stark begrenzt. Beispielsweise könnten die Leseberechtigungen erweitert werden.
Observer	Die Rolle hat ausschließlich Leseberechtigungen. Beispielsweise darf der Inhaber der Rolle bestimmte Server/Broker-Daten durchsuchen, Live-Daten lesen, historische Daten/Ereignisse lesen oder Daten/Ereignisse abonnieren.
Operator	Die Rolle hat die gleichen Berechtigungen wie der Observer. Zusätzlich könnten dem Operator Schreibberechtigungen für Live-Daten gegeben werden, sowie die Möglichkeit bestimmte Methoden aufzurufen.
Engineer	Die Rolle darf bestimmte Server/Broker-Daten durchsuchen, Konfigurationsdaten lesen/schreiben, historische Daten/Ereignisse lesen, Methoden aufrufen oder Daten/Ereignisse abonnieren.
Supervisor	Die Rolle darf durchsuchen, Live-Daten lesen, historische Daten/Ereignisse lesen, Methoden aufrufen oder Daten/Ereignisse abonnieren.
ConfigureAdmin	Die Rolle darf die nicht sicherheitsrelevanten Konfigurationseinstellungen einsehen und ändern.
SecurityAdmin	Die Rolle darf sicherheitsrelevante Einstellungen ändern.

3.4 AP 4 Public Key Infrastructure (PKI)

Das Arbeitspaket konzentriert sich auf die Konzeption einer Sicherheitsinfrastruktur, die die Authentifizierung mit sicheren Identitäten für Geräte, Anwendungen oder Benutzer sicherstellt und die Anwendbarkeit von Attributzertifikaten erleichtert, um die Autorisierung mit der gewählten Middleware von AP5 zu ermöglichen. Die Ergebnisse von AP2 und AP3 werden verwendet, um die Middleware-Features und deren Unterstützung für Role Based Access Control (RBAC) zu bestimmen.

Die folgenden Abschnitte werden wie folgt beschrieben,

- Auswertung von Zertifikatsbereitstellungsprotokollen - Projektarbeit von Herrn Maxim Friesen
- Sichere Identitäten - Hervorhebung der Bedeutung von Geräteidentitäten für die vertrauenswürdige Kommunikation
- Attributzertifikate - beschreibt die Anwendbarkeit von Attributzertifikaten im Zusammenhang mit der Verbesserung von Berechtigungsmechanismen im industriellen Umfeld
- Benutzeridentitäten - beschreibt die Verwendung der SmartCard als Qualifikationsträger für sichere Identitäten und gibt auch einen Überblick über ein Beispiel für AC, das in OPC UA zur Benutzerautorisierung verwendet wird

3.4.1 Bewertung von Zertifikatsverwaltungsprotokollen für Public Key Infrastrukturen in verteilten Systemen

Die Verfolgung und Zuordnung von Netzwerkaktivitäten zu einer eindeutigen Identität, die Anwendung von Zugriffskontrollmechanismen und die Einleitung einer End-to-End-Verschlüsselung erfordern eine vorherige Authentifizierung der Netzwerkteilnehmer. In industriellen Anwendungen ist eine sichere Kommunikation besonders wichtig, da Maschinendaten oder Wartungssteuerungen vor unbefugtem Zugriff geschützt werden müssen. Zu diesem Zweck werden Benutzer-IDs und Passwörter als Standardverfahren eingesetzt. Zu den anspruchsvolleren und sichereren Ansätzen gehören digitale Zertifikate. Spezifikationen wie X.509 bieten eine standardisierte Beschreibung einer Entität und ihrer digitalen Signatur auf Basis der Public-Key-Kryptographie. Zertifikate werden in verschiedenen Instanzen innerhalb einer Public Key Infrastructure (PKI) ausgestellt und validiert. Neue Zertifikate werden für neue Benutzer ausgestellt und abgelaufene Zertifikate werden gesperrt oder verlängert. In einem Netzwerk mit vielen Kommunikationspartnern können solche Prozesse mit verschiedenen Zertifikatsbereitstellungsprotokollen implementiert werden.

Eines der Ziele in diesem Projekt war die Entwicklung einer gemeinsamen Proof of Concept PKI für mehrere Kommunikationsprotokolle (OPC UA, MQTT, DDS). Ziel war es, den Verwaltungsaufwand für die Verwaltung mehrerer PKIs zu minimieren, wenn mehr als ein Kommunikationsprotokoll innerhalb derselben Netzwerkumgebung eingesetzt wird. Zur Umsetzung dieses Konzepts wurde ein Demonstrator entwickelt und eine Administrationsoberfläche erleichtert das Zertifikatsmanagement. Dazu war ein Zertifikatsbereitstellungsprotokoll erforderlich. Dies erforderte jedoch eine vorherige Bewertung der verschiedenen verfügbaren Protokolle. Hauptsächlich, um ein besseres Verständnis dafür zu erhalten, wie sie sich in Grundfunktionalität, Funktionsumfang und Sicherheitsniveau unterscheiden.

Im Rahmen einer studentischen Projektarbeit [25] wurde diese Bewertung unter Berücksichtigung der folgenden Protokolle durchgeführt:

- Automatic Certificate Management Environment (ACME) [93]
- Simple Certificate Enrollment Protocol (SCEP) [94]
- Certificate Management Protocol (CMP) [95]
- Certificate Management over Cryptographic Message Syntax (CMC) [96]
- Enrollment over Secure Transport (EST) [97]

Die allgemeine Funktionsfähigkeit der gennannten Protokolle wurde individuell überprüft und ihre verschiedenen Merkmale wurden verglichen und bewertet. Eine Reihe von Anforderungen, die Folgendes enthalten, wurde definiert:

- Automatisierte Zertifikatswiedereinschreibung oder -verlängerung
- Zertifikatswiderruf, Statusabruf
- Optimierung für eingeschränkte Geräte
- Standardisierung, Verfügbarkeit und Support
- Sicherer Transport

Diese Anforderungen wurden verwendet, um festzustellen, welches Protokoll für den Anwendungsfall im Rahmen dieses Projekts am besten geeignet ist.

Die Ergebnisse in [25] haben gezeigt, dass EST am besten zu den Anforderungen passt. Eine kurze Zusammenfassung der Bewertungsergebnisse wird in den folgenden Abschnitten erläutert.

3.4.1.1 Automatic Certificate Management Environment (ACME)

ACME ist zwar sehr beliebt und weit verbreitet, aber ein für den Einsatz in Web-CAs optimiertes Protokoll. Es wird hauptsächlich zur automatischen Ausstellung und Verlängerung von Zertifikaten für Webdomains eingesetzt und ist daher nicht der ideale Kandidat für einen Anwendungsfall mit einer starken vertikalen Integration der Kommunikation.

3.4.1.2 Simple Certificate Enrollment Protocol (SCEP)

SCEP wurde ursprünglich von Cisco mit der Absicht entwickelt, die Ausstellung von Zertifikaten in Netzwerken mit einer großen Anzahl von Kommunikationsgeräten zu erleichtern. SCEP ist HTTP-basiert und spezifiziert den Prozess der Beschaffung eines neuen Zertifikats und sichert die gesamte Kommunikation mit PKCS#7 - Cryptographic Message Syntax (CMS) [98]. Es befindet sich jedoch seit über 16 Jahren in der Entwicklung und hat über 30 Revisionen durchlaufen, ohne seinen Entwurfsstatus innerhalb der IETF zu überwinden. Wichtige Funktionen, wie zum Beispiel die Neuregistrierung von Geräten zur Verlängerung eines auslaufenden Zertifikats, waren ein nachträglicher Gedanke und wurden erst in späteren Revisionen implementiert. Darüber hinaus verwendet es ein einfaches Challenge-Passwort, um die Anforderung der Zertifikatsregistrierung zu autorisieren, was als unsicher gilt. Es wurde daher nicht in die nähere Auswahl aufgenommen.

3.4.1.3 Certificate Management Protocol (CMP)

CMP wurde von der IETF initiiert und deckt den größten Teil des Lebenszyklus eines Zertifikats einschließlich Antrag, Ausstellung, Verlängerung und Widerruf ab. Im Vergleich zu SCEP ist es viel komplexer und bietet eine große Vielfalt an Funktionen, die die Interaktionen aller relevanten PKI-Einheiten abdecken. Während CMP-Nachrichten auch in Abstract Syntax Notation One (ASN.1) angegeben sind, verwendet es kein CMS wie SCEP. Ein weiterer Unterschied sind die CA-Rollover-Mechanismen. Ein CA-Rollover ist der Sonderfall, wenn ein CA-Zertifikat abläuft und erneuert werden muss. Anschließend müssen auch alle ausgestellten ID-Zertifikate dieser CA erneuert werden. Während dies in SCEP an einem sogenannten "Flag Day" geschieht, der einen bestimmten Zeitpunkt beschreibt, an dem die alten Zertifikate ablaufen und die neuen gültig werden, verwendet CMP eine "Übergangsfrist". Dieser Zeitraum legt mehrere Phasen fest, in denen alte, neue und Zwischenzertifikate miteinander verkettet werden, um Geräten einen längeren Zeitraum für die Erneuerung ihrer Zertifikate aus der neuen CA zu ermöglichen. Diese Methode

ist weniger anfällig für organisatorische Fehler, da mehr Zeit für einen reibungslosen Übergang bleibt, anstatt den kompletten Switch an einem einzigen "Flag Day" fehlerfrei ausführen zu müssen. CMP spezifiziert keine Sicherheitsschicht für den sicheren Transport von Provisionierungsnachrichten. Es wurde in RFC 4210 [95] standardisiert, ist aber aufgrund seiner Komplexität nicht weit verbreitet.

3.4.1.4 Certificate Management over Cryptographic Message Syntax (CMC)

CMC ist dem CMP sehr ähnlich, mit dem Unterschied, dass es das CMS für seine Nachrichtensyntax verwendet. Es deckt die gleiche Funktionalität ab wie CMC und CMP, die beide von der IETF innerhalb kurzer Zeit mit den gleichen Zielen entwickelt wurden. Da sowohl SCEP als auch CMC auf CMS basieren, kann SCEP als ein vereinfachtes Profil von CMC mit einem kleineren Satz von Funktionen betrachtet werden. Ähnlich wie CMP, standardisiert in RFC 2511 [99], ist es aber aufgrund seiner Komplexität nicht populär.

3.4.1.5 Enrollment over Secure Transport (EST)

EST wurde von Cisco als Nachfolger für SCEP eingeführt und von der IETF unter RFC7030 [97] standardisiert. Es ist einfach gehalten, ähnlich wie SCEP, kombiniert aber verschiedene Funktionen anderer Protokolle, um vergangene Fehler zu beheben. Im Gegensatz zu SCEP ist die Zertifikatsverlängerung ein Hauptmerkmal, das von Anfang an integriert wurde. Darüber hinaus ermöglicht es die Unterstützung kryptographischer Algorithmen, wie z.B. Elliptic Curve Cryptograhphy (ECC), während SCEP RSA aufgrund seiner Abhängigkeit vom CMS nur RSA erlaubt. Zusammen mit seiner Unterstützung für die serverseitige Schlüsselgenerierung ist es daher besser für ressourcenbeschränkte Geräte geeignet, denen die erforderliche Rechenleistung fehlt. Anstatt einen "flag day" CA-Rollover zu verwenden, wird das in CMP eingeführte Konzept der "transition period" verwendet.

Zusammenfassend lässt sich sagen, dass EST entwickelt wurde, um der neue De-facto-Standard für die Bereitstellung von Zertifikaten zu werden, nachdem SCEP aufgrund seiner Einfachheit weithin angenommen wurde, aber aufgrund der laufenden Überarbeitungen seiner unvollendeten Spezifikation keine ausreichende Funktionalität und Sicherheit bietet. Die Einfachheit und der ausgereifte Funktionsumfang machten EST zur ersten Wahl für eine mögliche Implementierung im Rahmen des Demonstrators des Projekts.

3.4.2 Sicheren Identitäten

Die Ende-zu-Ende-Sicherheit in der industriellen Kommunikation ist eines der Forschungsgebiete im Rahmen von Industrie 4.0 [100]. Die Notwendigkeit einer unternehmensübergreifenden Kommunikation für den automatisierten Informationsaustausch hat gezeigt, wie wichtig es ist, vertrauenswürdige Kommunikationsverbindungen zu ermöglichen. Um eine vertrauenswürdige Kommunikation zu ermöglichen, sind detaillierte Untersuchungen der Sicherheitsanforderungen, der sicheren Identitäten [101] und deren Anwendbarkeit, erforderlich. Jede Einheit, die versucht, eine sichere Kommunikation aufzunehmen, muss mit einer sicheren Identität verknüpft werden. Eine sichere Identität ist eine einzigartige Identität mit zusätzlichen Sicherheitseigenschaften, wie sie von einem Secure Element (TPM, SmartCard) bereitgestellt werden, um eine vertrauenswürdige Authentifizierung der Entität [101] zu ermöglichen.

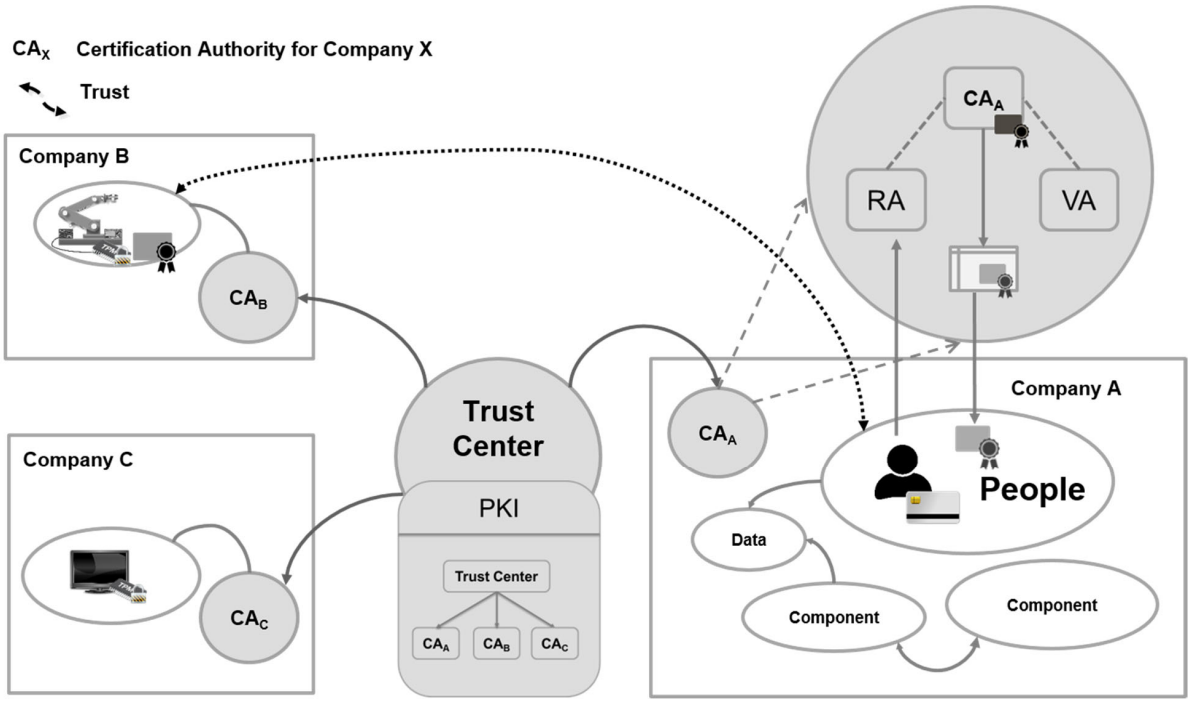


Abbildung 18: Sichere Ende-zu-Ende-Kommunikation innerhalb und über Unternehmensgrenzen hinweg [100]

Abbildung 18 leitet sich aus einer Veröffentlichung der Plattform Industrie 4.0 [100] ab. Das Beispielszenario in Abbildung 18 zeigt drei verschiedene Unternehmen, die über ein Trust Center Vertrauensbeziehungen aufbauen. Es gibt verschiedene Einheiten, die an jeder dieser Kommunikationsverbindungen innerhalb eines Unternehmens beteiligt sind, und sie können Komponenten, Personen usw. sein. Jedes Unternehmen, das versucht, eine sichere Ende-zu-Ende-Kommunikation aufzubauen, muss sich mit einer sicheren Identität authentifizieren. Abhängig von der gewählten sicheren Identität (z.B. ein privater Schlüssel mit einem zugehörigen X509 v3-Zertifikat) [102] ist der Träger für die Identität ein geeignetes Secure Element (z.B. eine SmartCard). Der in Abbildung 18 dargestellte Vertrauenslink zeigt einen Benutzer von Firma A, der sich mit einer Maschine in Firma B unter Nutzung sicherer Identitäten verbindet.

3.4.3 Attributzertifikat

Die in [103] beschriebene Privilege Management Infrastructure (PMI) definiert die am Autorisierungs-Framework beteiligten Einheiten. Ihre Anwendungsfälle im Rahmen von Industrie 4.0 werden in [104] erläutert. Ein PMI definiert ein Framework für die Zugriffskontrolle, das zur Verwaltung der Berechtigungen Attributzertifikat (engl. „Attribute Certificate“ (AC)) nutzt, die von sogenannten Attribute Authorties (AA) ausgestellt werden. Ein AC ist ein X.509-Zertifikat der Version 2, beinhaltet also keinen öffentlichen Schlüssel [105]. Jeder Benutzer benötigt für die Authentifizierung und Autorisierung ein eigenes Public-Key-Zertifikat (engl. „Public-Key Certificate“ (PKC)) und ein AC (Abbildung 19). Der Vorteil bei der Nutzung eines zusätzlichen ACs für die Berechtigungsvergabe liegt in darin, dass Berechtigungen relativ kurzlebig sein können und ein AC leichter, aufgrund des fehlenden Schlüssels, auszutauschen ist als ein PKC. Als Analogie lässt sich das PKC mit einem Reisepass und das AC mit einem Visum, welches eine temporäre Einreiseberechtigung ermöglicht, vergleichen [105]. Im Rahmen dieser Arbeit enthält das AC einen Attributtyp "Rolle", der die dem PKC-Inhaber zugewiesene Rolle bestimmt. Daher wird es auch als Rollenzuordnungs-AC bezeichnet.

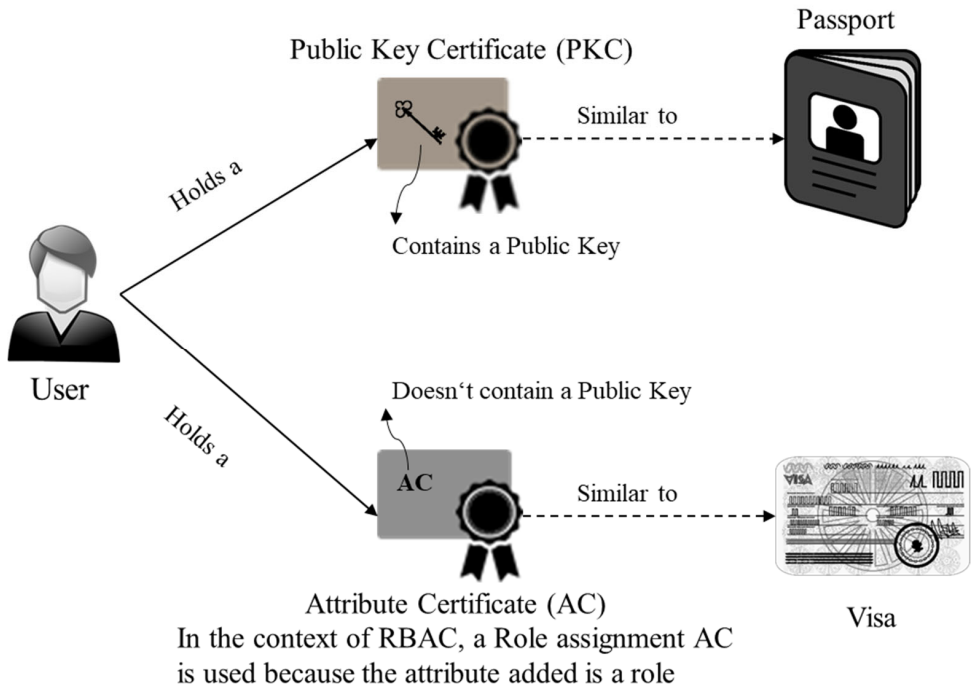


Abbildung 19: Eine Analogie für die AC-Nutzung [24]

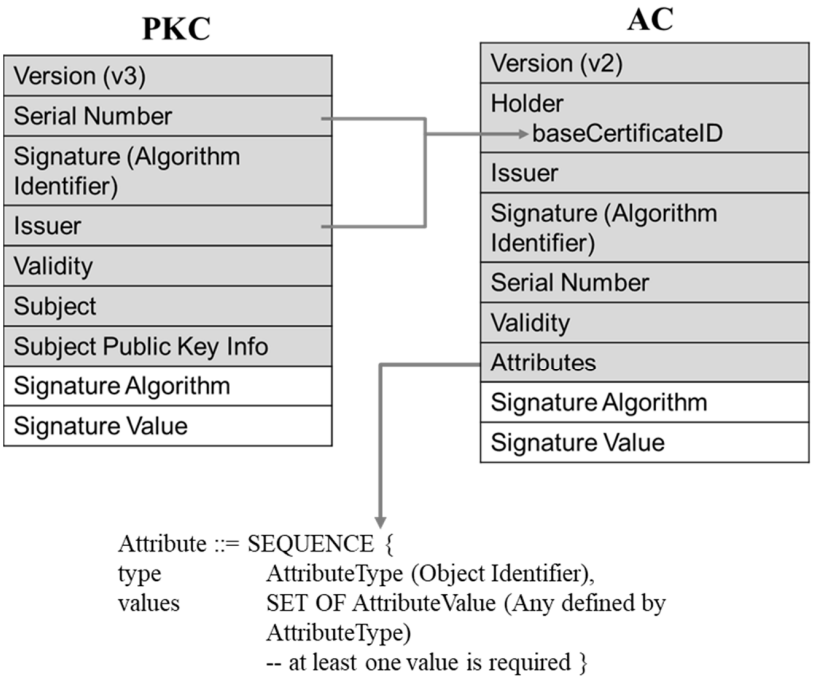


Abbildung 20: PKC und Attribut Zertifikate [24]

Abbildung 20 zeigt die Felder in einem PKC-ähnlichen Format, welche die Seriennummer und dem Emittenten, der den "Halter" einer AC definiert beschreibt. Obwohl es verschiedene Optionen für das Feld "Halter" einer AC [105] gibt, sollte es die baseCertificateID für das Halterfeld verwenden, wenn die Authentifizierung auf X509 PKC basiert. Das Feld "Attribute" eines AC bestimmt die verschiedenen Merkmale, die einer Entität zugeordnet sind. Die Attributtyp- und Wertepaare bestimmen den Zweck einer AC. Der Attributtyp wird über einen Object Identifier (OID) definiert und die Werte werden typabhängig definiert.

Eine Stelle, die Attributzertifikate ausstellt (AA), wird in [105] als AC-Emittent bezeichnet. Nach [105] darf eine AA nicht gleichzeitig ein PKC-Emittent sein, um die Verwechslung von Seriennummern und Widerrufen zu vermeiden. Abbildung 21 zeigt eine Root Certification Authority (CA), eine untergeordnete CA und eine AA, wobei die AA nur die ACs mit den notwendigen Informationen aus der zugehörigen PKC ausstellt [102].

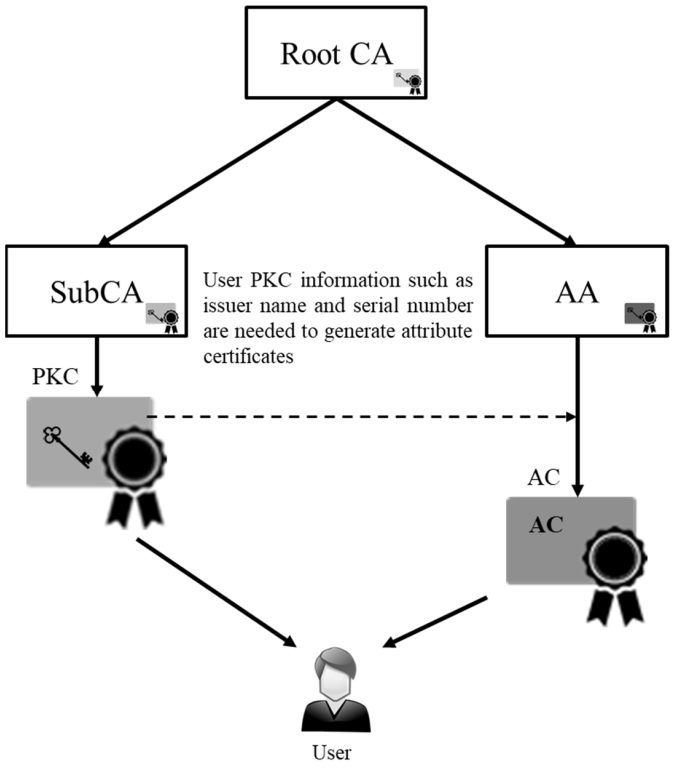


Abbildung 21: PKI und Attribute Authority [24]

Um die Verwendung von ACs zu demonstrieren, können sie mit Hilfe von Java-basierten Bibliotheken generiert werden. Anzumerken ist jedoch, dass das Open PMI-Projekt mit dem openssl-Tool die Attributzertifikate erzeugt [106]. Diese AC-Zertifikate können in einem Pull- oder Push-Modell verteilt werden.

3.4.3.1 Role Based Access Control (RBAC) vs Attribute Based Access Control (ABAC)

Die rollenbasierte Zugriffskontrolle (RBAC) gemäß [107] bestimmt die Berechtigungen, basierend auf den, den Benutzern zugeordneten Rollen sowie den Rollen zugeordneten Berechtigungen. Grundelemente eines RBAC-Modells sind Benutzer, Rollen, Operationen, Objekte und Berechtigungen.

Die Grundbegriffe sind in [107] wie folgt definiert:

- Benutzer: Ein Benutzer ist eine Person (kann auf Komponenten und Maschinen erweitert werden), die Zugriffsanforderungen initiiert.
- Rolle: Eine Rolle ist ein Titel, der den Aufgaben zugeordnet ist, die für eine Stellenfunktion im Rahmen eines Unternehmens beschrieben werden.
- Betrieb: Eine Operation ist eine ausführbare Funktion auf einem von einem Benutzer initiierten Objekt.
- Objekt: Ein Objekt ist eine Systemressource, die Zugriffsbeschränkungen erfordert.
- Erlaubnis: Es ist eine Berechtigung zum Ausführen einer Operation.

RBAC ist ein Sonderfall der attributbasierten Zugriffskontrolle (ABAC), wie in [108] beschrieben. Das ABAC-Modell ist ein Zugriffskontrollmodell, das auf den Attributen basiert, bei denen die Attribute aus Subjekten (Benutzern), Objekten und der Umgebung bestimmt werden. Im ABAC-Modell sind die Entscheidungen dynamisch unter Berücksichtigung der Attribute jeder Entität wie Benutzer, Objekt und Umgebung, was die Komplexität und Zeit erhöht, die benötigt wird, um ein solches Modell praktisch zu realisieren. RBAC, wobei die Rollenliste die Komplexität reduziert, aber im Voraus festgelegt ist [109].

3.4.3.2 ABAC in Industrie 4.0

Die neueste Veröffentlichung [110] der Plattform Industrie 4.0 stellt die Bedeutung der Zugangskontrolle (Access Control) im Kontext von Industrie 4.0 und die wichtigsten Ansätze zur Umsetzung dar. Wie bereits erwähnt, wird das in Industrie 4.0 häufig vorgeschlagene OPC UA Framework für die in [110] beschriebenen Anwendungsfälle berücksichtigt. Einige der wichtigsten Highlights sind die Unterscheidung zwischen ABAC und RBAC, die Granularität von ABAC im Rahmen von Industrie 4.0 und Asset Administration Shell (AAS) sowie die vorausgesagte Anwendbarkeit für eine Zugriffskontrolle im Rahmen von OPC UA. RBAC bleibt jedoch im ABAC-Modell integriert, da die Attribute des ABAC-Modells auch einem Zwischenrollenattribut [110] zugeordnet werden können. Ein vergleichender und kombinierter Ansatz von RBAC und ABAC ist notwendig, um bestehende RBAC-kompatible Zugangskontrollsysteme auf ABAC auszudehnen.

3.4.4 Nutzeridentitäten (User Identities)

3.4.4.1 Sichere Identitäten – SmartCards

Abbildung 22 zeigt eine exemplarische PKI, die für ein Unternehmen 'A' (gemäß Abbildung 18) erstellt und gepflegt werden muss. Abbildung 22 zeigt verschiedene untergeordnete CAs, die an der Ausstellung von Zertifikaten für Benutzer und Maschinen beteiligt sind. Abbildung 22 zeigt

auch die Verwaltung von Benutzerzugriffsrechten, sichere Elemente als Qualifikationsträger und vertrauenswürdige Kommunikationsverbindungen innerhalb und außerhalb von Unternehmensgrenzen. Dies ist ein einfacher Anwendungsfall, um die Anwendbarkeit von PKI in einer industriellen Umgebung aufzuzeigen.

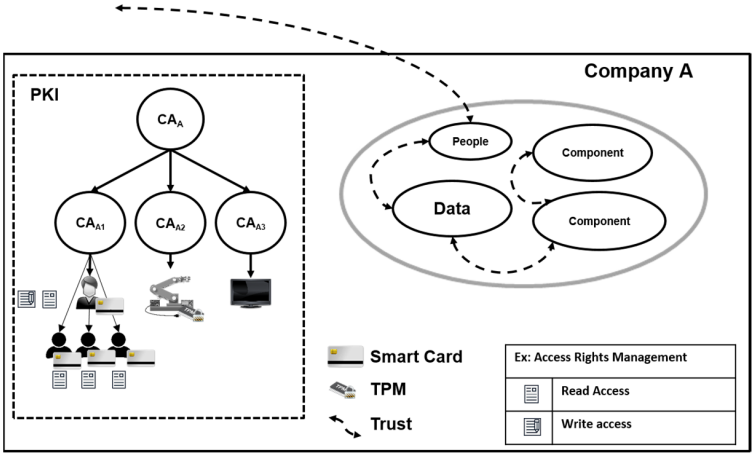


Abbildung 22: Beispiel PKI – Firma A [20]

3.4.4.2 Nutzer-Autorisierung unter Verwendung von AC in OPC UA

Um die Benutzerauthentifizierung über PKCs und die Autorisierung über ACs für ausgewählte Middleware-Anwendungen zu realisieren, ist eine mehrstufige PKI aufgebaut, wie in Abbildung 23 dargestellt. Die RootCA hat drei unterlagerte Sub_CAs, nämlich eine 'Sub_CA1' für die Ausgabe der OPC UA/MQTT Anwendungszertifikate und eine 'Sub_CA2' für die Ausgabe der Benutzer PKC-Zertifikate für den OPC UA Benutzer X509IdentityToken Illustration und eine 'AA' für die Ausgabe der Benutzerattributzertifikate. Das OPC UA Framework verwendet die Zertifikate der Anwendungsinstanz für eine Anwendung, um sich gegenüber einer anderen Anwendung zu authentifizieren. Und es bietet eine Benutzerauthentifizierungsoption zur Verwendung des X509IdentityToken, mit dem ein vom Benutzer bereitgestelltes X509 v3-Zertifikat übergeben wird [111]. Um die beispielhafte Möglichkeit einer Autorisierung mit ACs zu veranschaulichen, wurde eine solche für OPC UA in dem im Forschungsprojekt entwickelten Demonstrator umgesetzt. Ebenso kann die PKI für DDS-Domainteilnehmer erweitert werden. Das Keytool-Utility wurde verwendet, um die erforderlichen Schlüsselpaare und X509-Zertifikate zu generieren. Die Keytool-Kommandos wurden in Java-Klassen organisiert, die als NetBeans-Projekt gebündelt wurden, um einfache Fehlerbehandlungen und ggf. zukünftig erforderliche Änderungen zu erleichtern.

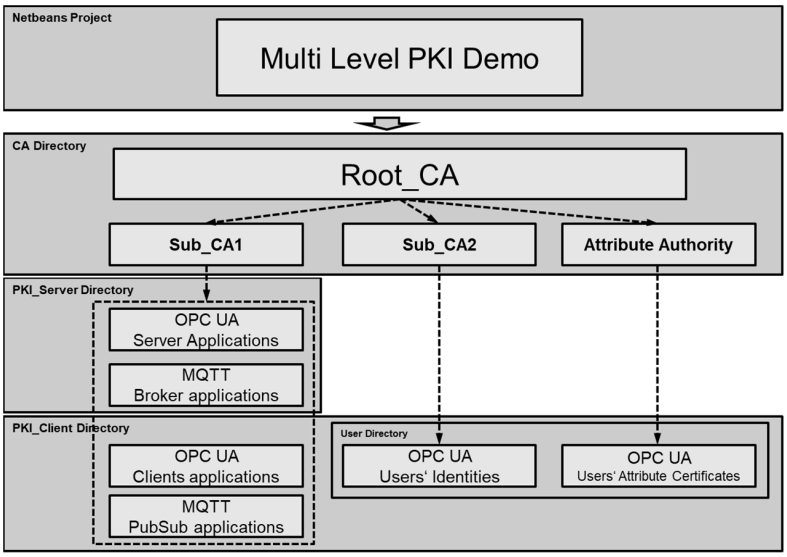


Abbildung 23: Multilevel PKI für den Demonstrator

Anschauliche Beispiele für die in diesem Abschnitt beschriebene Authentifizierung über sichere Identitäten und die Autorisierung über Attributzertifikate werden für die gewählten Middleware-Lösungen am Demonstrator implementiert in 4.

3.5 AP 5 Middleware

Auf Basis der in Kapitel 3.1 gewonnenen Übersicht über vorhandene Middleware-Lösungen, werden diese in diesem Kapitel unter Berücksichtigung von definierten Kriterien evaluiert. Anschließend werden Middleware-Lösungen und entsprechende Implementierungen, die eine sichere Kommunikation zwischen allen Komponenten und Systemen eines verteilten Automatisierungssystems ermöglichen, ausgewählt.

3.5.1 Evaluationskriterien

Bei der Wahl einer geeigneten Middleware-Lösung für Systeme in der I4.0 spielen diverse Faktoren eine Rolle. Aus diesem Grund werden im Folgenden zunächst generelle Anforderungen und anschließend Anforderungen an die IT-Sicherheit der Middleware definiert.

3.5.1.1 Generelle Anforderungen

Nachrichtenparadigmen beschreiben das Muster des gegenseitigen Nachrichtenaustauschs. Bei *Request-Response* (RR) werden von einem Client Ressourcen bei einem Server angefordert. Diese werden in einer Antwort an den Client weitergeleitet. Bei einem *Publish-Subscribe*-System (PS) können Server Ihre Ressourcen in sogenannte Topics unterteilen und einem Broker mitteilen. Interessierte Clients können Topics abonnieren. Neue Nachrichten innerhalb eines Topics werden vom Broker an alle Abonnenten weitergeleitet, ohne dass diese angefordert werden müssen. Bei einem *Point-to-Point*- Austausch (PP) wird zwischen einem Sender und einem Empfänger eine Message-Queue über einen Broker gebildet. Nachrichten werden vom Sender in die Queue eingereiht und solange Einbehalten bis sie von einem Empfänger konsumiert werden.

Echtzeitfähigkeit (EZ) ist für Anwendungen mit zeitkritischen Übertragungsfenstern relevant. In diesem Beitrag wird daher zwischen harter, weicher und keiner Echtzeitfähigkeit unterschieden.

Harte Echtzeitfähigkeit bedeutet, dass Deadlines immer eingehalten werden. Bei weicher Echtzeitfähigkeit liegt die durchschnittliche Übertagungszeit innerhalb der Deadline aber vereinzelte Überschreitungen sind tolerabel. Bei keiner Echtzeitfähigkeit sind keinerlei Garantien für die rechtzeitige Zustellung von Nachrichten gegeben.

Quality-of-Service (QoS): QoS-Optionen werden benötigt, um die von Anwendungen vorgeschriebene Dienstgüte der Kommunikation einzuhalten. Da in diesem Beitrag nur Protokolle auf der Anwendungsschicht untersucht werden, sind nur zusätzliche QoS-Optionen relevant, die die grundlegenden QoS-Funktionen oberhalb der Transportschicht, zum Beispiel von TCP/IP, ergänzen. Dabei wird zwischen umfassenden, elementaren und keinen QoS-Optionen unterschieden. Zu elementaren QoS-Funktionen zählen einfache Zustellungsbestätigungen oder die Behandlung von Zeitüberschreitungen und Übertragungsfehlern. Bei umfassenden QoS-Funktionen ist ein umfangreiches Portfolio an Optionen gegeben, um die Härte der Echtzeitfähigkeit und den Grad der Zuverlässigkeit in Form von Redundanzen oder verschiedener Wiederübertragungsarten festzulegen.

Ressourcenbeschränkte Plattformen wie eingebettete Systeme verfügen oft nur über geringe Speicherkapazität und Rechenleistung. Zudem bestehen teilweise Beschränkungen in Bezug auf die zulässige Energieaufnahme. Middleware-Ansätze müssen für ein ressourcenarmes Arbeitsumfeld optimiert sein, um unter derartigen Bedingungen die gewünschte Funktionalität zu bieten.

Verfügbare Implementierungen sind notwendig, um ein Middleware-Konzept in einem System zu integrieren. Nicht alle Ansätze wurden bisher in einer funktionierenden Implementierung umgesetzt. Die Verfügbarkeit von Open-Source und/oder kommerzieller Implementierungen ist daher eine grundsätzliche Voraussetzung bei der Auswahl geeigneter Lösungen.

3.5.1.2 Anforderungen an die IT-Sicherheit

Die IT-Sicherheitsziele der Vertraulichkeit, Authentizität und Autorisation werden nachfolgend als Anforderungen an die Informationssicherheit von Middleware-Systemen definiert.

Ende-zu-Ende (E2E) Verschlüsselung ist für einen vertraulichen Datenaustausch essentiell. In Abhängigkeit von dem verwendeten Verschlüsselungsverfahren und einem zwischen den Kommunikationspartnern etablierten geheimen Schlüssel werden alle Nachrichten vom Sender verschlüsselt und vom Empfänger wieder entschlüsselt. Dafür werden entweder öffentliche Standards wie *Transport Layer Security* (TLS) oder *Datagram TLS* (DTLS) in den Protokollstack integriert oder es wird auf proprietäre Verschlüsselungslösungen zurückgegriffen.

Authentifizierung dient zum Identitätsnachweis eines Kommunikationspartners und garantiert, dass empfangene Daten von der authentifizierten Instanz stammen. Dazu dienen im simpelsten Fall eine Nutzername- und Passwort-Abfrage. Zu zuverlässigeren Lösungen zählen Zertifikate oder Token, die von einer vertrauenswürdigen dritten Instanz ausgestellt und dadurch verifizierbar sind. Die bei der Ende-zu-Ende-Verschlüsselung vorausgehende gegenseitige Authentifizierung der Kommunikationspartner auf der Transportschicht wird in diesem Punkt nicht betrachtet. Der Fokus liegt auf den Mechanismen zur Authentifizierung einzelner Benutzer und Clients auf der Anwendungsschicht, um es Applikationen zu ermöglichen, den Datenzugriff individuell anzupassen.

Rollenbasierte Zugriffskontrolle verwendet eine Client-Authentifizierung, um die Zugriffsrechte auf bestimmte Ressourcen zu beschränken. Abhängig von der zugewiesenen Rolle eines authentifizierten Clients kann beispielsweise der Schreib- oder Lesezugriff auf ausgewählte Datensätze erlaubt oder verweigert werden. Dies ist insbesondere bei Framework-Protokollen relevant, die eigene Datenmodelle spezifizieren und damit auch eine Zugriffskontrolle über die eigenen Sicherheitsmodule integrieren können.

3.5.2 Auswahl der Middleware-Lösungen

In Abbildung 24 ist ein strukturiertes Modell der evaluierten Middleware-Protokolle aus Kapitel 3.1.1 abgebildet. Es hebt die Abstraktionsstufen der zwei Protokollklassen aus Kapitel 3.1.1 hervor und bildet die Abhängigkeiten der Protokolle untereinander ab. Weiterhin wird deutlich, dass der Großteil der Middleware-Lösungen für die Verschlüsselung des Datenkanals auf den TLS/DTLS Standard zurückgreift. Die Ergebnisse der Analyse der Anforderungen sind in Tabelle 6 zusammenfassend dargestellt. Anders als in Abbildung 24 unterscheidet sich die Auswahl der Sicherheitsfunktionen für die Authentifikation von Clients und Benutzern stark. Zwar decken sich die grundlegenden Authentifizierungsmechanismen wie X.509 Zertifikate oder Kerberos Tickets, doch ist die Implementierung der Sicherheitsinfrastruktur bei allen Framework-Protokollen Middleware-spezifisch. Die Verwendung von standardisierten Authentifizierungslösungen, wie SASL oder WS-Trust, ist nur bei einigen Messaging-Protokollen wiederzufinden. In den vertikal-orientierten Netzwerken in der I4.0 ist die gleichzeitige Verwendung mehrerer Middleware-Lösungen zu erwarten. Die Nutzung Middleware-spezifischer Sicherheitsfunktionen erschwert dabei jedoch die Verwaltung und Sicherung des Kommunikationsnetzes bei Einsatz verschiedener Protokolle.

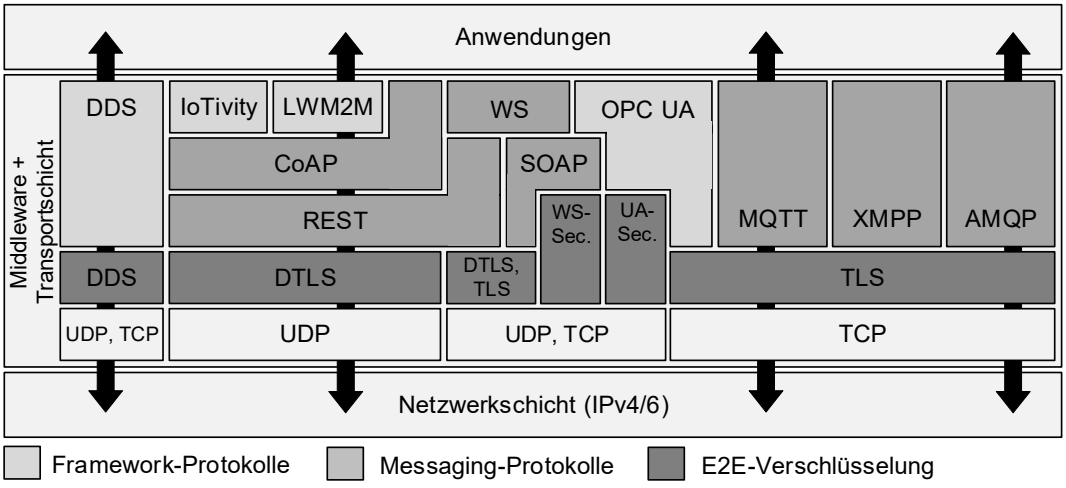


Abbildung 24: Strukturiertes Modell der evaluierten Middleware-Protokolle

Tabelle 6: Übersicht aller evaluierten Middleware-Protokolle

Protokoll		Organisation	Protokolltyp	Transport-protokoll	Open-Source	Echtzeit	Zusätzliche QoS	Beschränkte Systeme	Nachrichten-paradigma	Sicherheit		
										E2E- Verschl.	Authent.	Zugriffs-Kontrolle
WS	**SOAP**	W3C	MP	TCP,UDP	✓	✗	✓	✗	RR	TLS,WS-Sec.	WS-Trust	✓
	REST	W3C	MP	TCP,UDP	✓	✗	✗	✗	RR	TLS, DTLS	✗	✗
CoAP		IETF	MP	REST/UDP	✓	✗	○	✓	RR	DTLS	✗	✗
MQTT		OASIS	MP	TCP	✓	○	✓	✓	PS	TLS	TLS	✓
AMQP		OASIS	MP	TCP	✓	✗	✓	✗	RR, PS, PP	TLS	SASL	✓
XMPP		IETF	MP	TCP	✓	○	✗	✗	PS	TLS	SASL	✓
IoTivity		OCF	FP	CoAP/UDP	✓	✗	○	✓	RR	DTLS	P	✓
LWM2M		OMA	FP	CoAP/UDP	✓	✗	○	✓	RR	DTLS	P	✓
OPC UA		OPCF	FP	SOAP/TCP UDP	✓	✗	○	✓	RR,PS	TLS,P, WS-Sec.	P	✓
DDS		OMG	FP	TCP, UDP	✓	✓	✓	✓	PS	P	P	✓

✓ = Unterstützt/Harte EZ/Umfangreiche QoS | ✗ = Nicht unterstützt/Keine EZ/Keine QoS | ○= Weiche EZ/Beschränkte QoS

MP = Messaging-Protokoll | FP = Framework-Protokoll | P = Proprietär

Die endgültige Auswahl für die weitere Bearbeitung innerhalb des Projektes erfolgte unter Berücksichtigung der Anforderungen und der Voruntersuchung, außerdem wurde die allgemeine Akzeptanz in der Industrie bestimmter Protokolle beachtet. Es wurden die Protokolle DDS, MQTT und OPC UA (Tabelle 6 Gelb) ausgewählt.

3.5.3 Auswahl der Implementierungen

Die Analyse der Protokolle basiert auf den jeweiligen Spezifikationen. Die Spezifikation einer Middleware gibt lediglich vor, welche Funktionen theoretisch unterstützt werden und wie das generelle Konzept aussieht. Die letztendliche Umsetzung der Funktionen und der Sicherheitsfeatures ist implementierungsabhängig. Da die Standardisierung der Middleware-Protokolle schon vor vielen Jahren erfolgt ist, haben sich eine Vielzahl, sowohl kommerzieller, als auch frei verfügbarer (Open Source), Implementierungen am Markt etablieren können. Auf Grundlage der durchgeführten Analyse der Protokolleigenschaften wurden in einem folgenden Schritt am Markt verfügbare Implementierungen der einzelnen Middleware-Protokolle untersucht, um die Implementierungen für den geplanten Demonstrator auswählen zu können. Hierbei lag der Fokus auf der Umsetzung

der spezifizierten Sicherheits-Features in den einzelnen Protokollen. Es wurden auf Basis einer vorangehenden Studie [112] die folgenden Implementierungen an Hand eines projektspezifischen Kriterienkataloges ausgewählt:

- Mosquitto für MQTT [113]
- MILO für OPC UA [114]
- CoreDX DDS von Twin Oaks für DDS [115]

In Tabelle 7 werden die IT-Sicherheits-Features der Implementierungen dargestellt.

Tabelle 7: Übersicht der Sicherheits-Features der ausgewählten Implementierungen

<table>
<tr><th>OSI-Schicht</th><th>Sicherheitsaspekt</th><th>Mosquitto MQTT</th><th colspan="2">CoreDX DDS</th><th colspan="3">MILO OPC UA</th></tr>
<tr><td rowspan="6">Anwendungsschicht</td><td rowspan="4">Authentifizierung</td><td rowspan="4">Benutzername/Passwort & Plugin</td><td colspan="2" rowspan="4">X.509</td><td rowspan="4">Nutzung von OPC UA Secure Channel ermöglicht:</td><td colspan="2">Anonym</td></tr>
<tr><td colspan="2">Benutzername/Passwort</td></tr>
<tr><td colspan="2">X.509IdentityToken</td></tr>
<tr><td colspan="2">IssuedIdentityToken</td></tr>
<tr><td rowspan="2">Autorisierung</td><td rowspan="2">Access Control List</td><td colspan="2" rowspan="2">Permissons Document</td><td colspan="3">“Information Modelling: configure access level / user access level” (Stand 2015)</td></tr>
<tr><td colspan="3">“Attribute based access control: Role object” (Stand 2017)</td></tr>
<tr><td>Vertraulichkeit</td><td>Abhängig von der Krypto-Bibliothek</td><td colspan="2">AES-GCM</td><td rowspan="2">Abhängig von der Security-Policy</td><td colspan="2">z. B. AES, RSA, usw.</td></tr>
<tr><td>Integrität</td><td>Abhängig von der Krypto-Bibliothek</td><td colspan="2">AES-GMAC</td><td colspan="2">z. B. SHA256, RSA, usw.</td></tr>
<tr><td rowspan="3">Transportschicht</td><td>Authentifizierung</td><td rowspan="3">X.509 (TLS)</td><td rowspan="3">X.509 (D/TLS)</td><td>×</td><td rowspan="3">X.509 (TLS)</td><td rowspan="3">OPC UA Secure Channel</td><td rowspan="3">WS - Security</td></tr>
<tr><td>Vertraulichkeit</td><td>AES-GCM</td></tr>
<tr><td>Integrität</td><td>AES-GMAC</td></tr>
</table>

3.6 AP 6 Echtzeit-Ethernet

Im Projektantrag wurde auch das Thema Echtzeit-Ethernet adressiert. Hierbei sollte ein besonderer Fokus auf dem in Deutschland weit verbreiteten PROFINET Protokoll liegen. Zeitgleich zum laufenden Forschungsprojekt wurde in der PROFIBUS Nutzerorganisation in der Arbeitsgruppe SB PG10 (Security) ein Projekt zur Spezifikation eines Security Layers für PROFINET gestartet. Da es keinen Sinn machte, parallel dazu im Forschungsprojekt IT_SIVA eine konkurrierende Architektur zu entwickeln, wurde beschlossen, dass Herr Prof. Dr. Niemann in dem Projekt der PNO mitarbeitet und die Anforderungen aus Sicht des Forschungsprojektes IT_SIVA einbringt. Im Rahmen der PNO-Arbeit wurde ein Whitepaper mit den grundlegenden Anforderungen und Konzepte erstellt [116]. Darüber hinaus hat Prof. Dr. Niemann die Konzepte auf einer IEEE-Konferenz in Helsinki vorgestellt [18] Weitere grundlegende Überlegungen zum Thema Echtzeit-Ethernet finden sich auch im Schlussbericht des Forschungsprojektes SEC_PRO [117] [118].

Aus den beschriebenen Gründen wird daher auf eine detaillierte Darstellung des Themas Echtzeit-Ethernet an dieser Stelle verzichtet.

3.7 AP 7 Verwaltungswerkzeug

Die beiden verschiedenen Kernpunkte, die für das Arbeitspaket erreicht werden sollen, sind die Entwicklung eines Rollenverwaltungswerkzeugs zur Verwaltung der Zugriffsrechte und eines Visualisierungs- und Überwachungswerkzeugs zum Visualisierung des aktuellen Zustands der Geräteverbindungsmodi/-typen und des aktuellen Security-Status je nach Wahl der Middleware. In dem „Zusatzdokument zu AP7[4]“ werden die einzelnen Tools illustriert.

3.7.1 AP7.1 Rechte- und Rollenverwaltung

Der Ansatz zum Entwurf des Tools und zur Implementierung des Zugriffskontroll-Frameworks mit in RBAC integriertem ABAC werden in den folgenden Abschnitten beschrieben.

3.7.1.1 Role Administration Tool – Anforderungen

Abbildung 25 zeigt die wichtigsten Entwurfsmodule für das Werkzeug. Die konzeptionelle Idee wird im Folgenden in Stichpunkten weiter ausgeführt:

- Die beiden Hauptmodule dienen der Verwaltung von Rollen / Benutzern und der Verwaltung von Anmeldeinformationen (Generierung von ACs).
- Verwaltung von Rollen und Benutzern:
    - Die Liste der Rollen muss verwaltet werden. Dies bedeutet, dass eine genau definierte Rolle und ihre Beschreibung, die für den Anwendungsfall erforderlich ist, mit dem Tool erstellt, gelöscht oder geändert werden kann.
    - Die Liste der Benutzer und ihrer zugehörigen Identitäten muss verwaltet werden. Zur Veranschaulichung kann eine Benutzeridentität mithilfe eines vorhandenen X509-Zertifikats erstellt werden, oder es kann ein neues X509-Zertifikat erstellt werden, das vom Sub_CA2 signiert wurde, oder es kann eine sichere Identität mithilfe der SmartCard erstellt werden. In diesem Fall wird die SmartCard später für die Benutzerauthentifizierung mit dem Demonstrator verwendet.

[4] Abrufbar unter www.fe-zvei.org

- Verwaltung der Berechtigungsnachweise, die zum Zuordnen einer Rollenzuweisung zwischen Benutzer und Rolle erforderlich sind:
    - Es gibt 3 Untermodule, um die verschiedenen vorhandenen Funktionen oder Optionen für die Zugriffssteuerung in der ausgewählten Middleware zu vereinfachen und zu berücksichtigen.
    - Im Falle von OPC UA werden mit diesem Tool die in AP4 beschriebenen Konzepte demonstriert. Attributzertifikate werden mit einem Rollenattribut generiert, indem der vorhandene Benutzer einer vorhandenen Rolle zugeordnet wird, die zuvor mit dem Tool erstellt wurde:
        - OPC UA applicationURI identifiziert die Anwendung eindeutig und wird daher als Attribut verwendet, um die benutzerspezifische Rollenzuweisung in Bezug auf die Ressourcen zu bestimmen.
        - Die AC-Vorlage kann angezeigt und geändert werden.
- Bei MQTT-Publisher- oder Abonnentenanwendungen kann der Name des Maklerthemas als Attribut verwendet werden, um die Zuordnung zwischen dem Benutzer (kann eine Pub / Sub-Anwendung sein) und der Rolle zu bestimmen.

Für DDS-Domain-Teilnehmer kann die eindeutige Kennung ermittelt und auf ähnliche Weise zur Erstellung einer Rollenzuweisung zwischen Rolle und Benutzer verwendet werden.

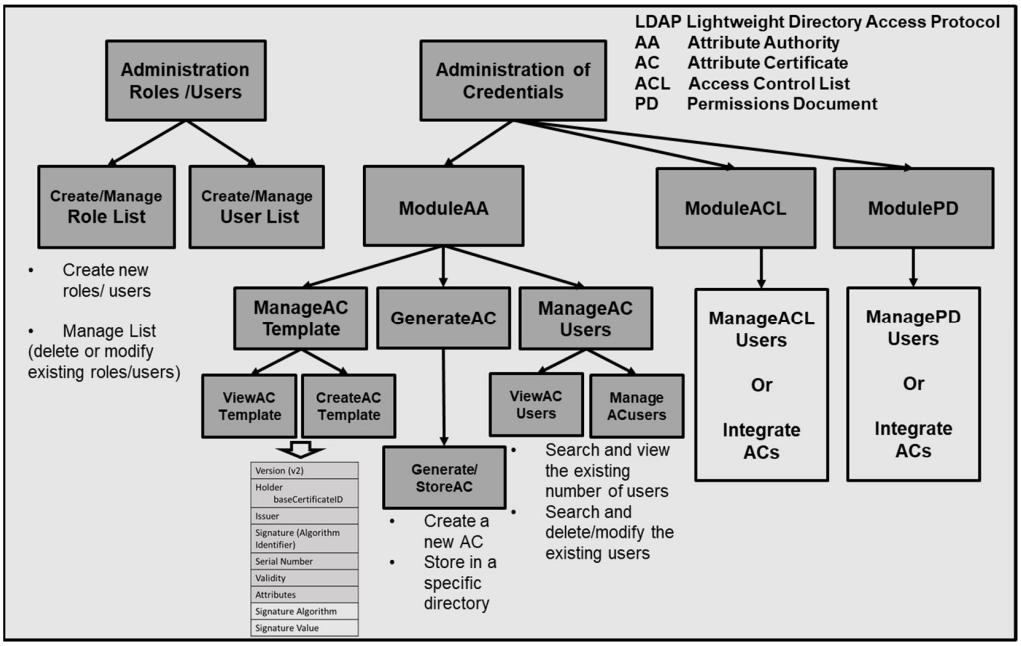


Abbildung 25: Role Administration Tool – Wesentliche Module

Design/ Implementierung

Das Design und die Implementierung werden in diesem Abschnitt erläutert. Die Benutzeroberfläche (UI) wurde mit Java-Swing-Komponenten entwickelt und das Backend zum Push und Pull von Identitäts- und Attributzertifikaten war der Apache DS LDAP-Server. Der LDAP-Server muss über das erforderliche Schema verfügen, um Einträge zu erstellen und die Attribut-Einträge nach Bedarf zu speichern. Abbildung 26 zeigt die beiden Umgebungen, in denen der Einsatz des Tools zu berücksichtigen ist, eine davon ist die Anwendungsentwicklung und die andere die Verwaltung von Zugriffsrechten. Auf der Seite der Anwendungsentwicklung der Fabrik wird jede Anwendung

und ihre jeweiligen Ressourcen (object1, object2, etc., siehe Abbildung 26) bestimmt, die wiederum die Access Control List (ACL) zur Einschränkung des Zugriffs bestimmt. Auf der Administrationsseite der Fabrik wird die Liste der von der Entwicklungsseite bestimmten Rollen gepflegt. Die Listenbenutzer mit den jeweiligen eindeutigen Identitäten werden verwaltet und das Tool hilft bei der Erstellung der Rollenzuordnung zwischen Benutzeridentifikationsattribut, AC mit spezifischer Rolle und der jeweiligen Anwendung. Im Falle des IT SIVA-Demonstrators werden OPC UA-Anwendungen zur Veranschaulichung derselben verwendet.

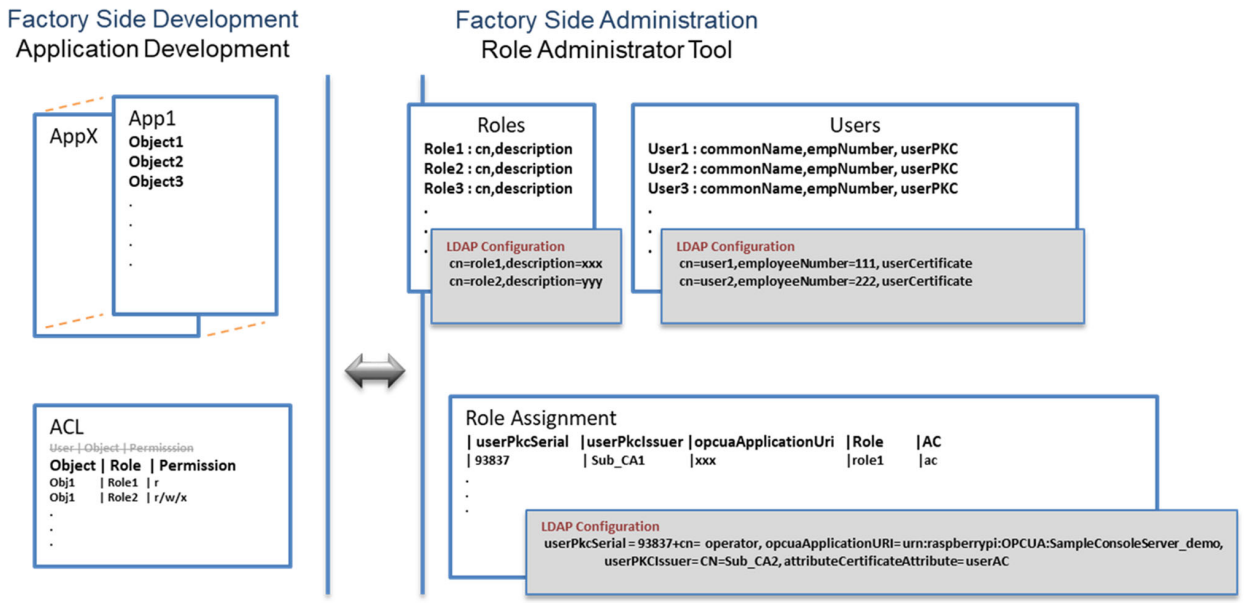


Abbildung 26: Einsatzumgebungen des Role Administration Tools

3.7.2 AP7.1 Visualisierung der aktuellen IT-Sicherheitslage im Netzwerk

Um den aktuellen Zustand des Gesamtsystems, sowie genauere Informationen über die Kommunikation der Netzwerkteilnehmer zu visualisieren, wurde ein Monitoring-Tool entwickelt. Im Folgenden wird auf die Komponenten, die Funktionsweise und die Wartung eingegangen.

3.7.2.1 Komponenten und Funktionsweise

Zur Überwachung der Protokollteilnehmer wurde eine Monitoring Lösung auf Basis von Node.js entwickelt. Grundsätzlich besteht die Implementierung dabei aus drei Komponenten:

1. Backend Applikation
2. Client Applikation
3. Frontend Applikation

In Abbildung 27 wird der schematische Ablauf dargestellt.

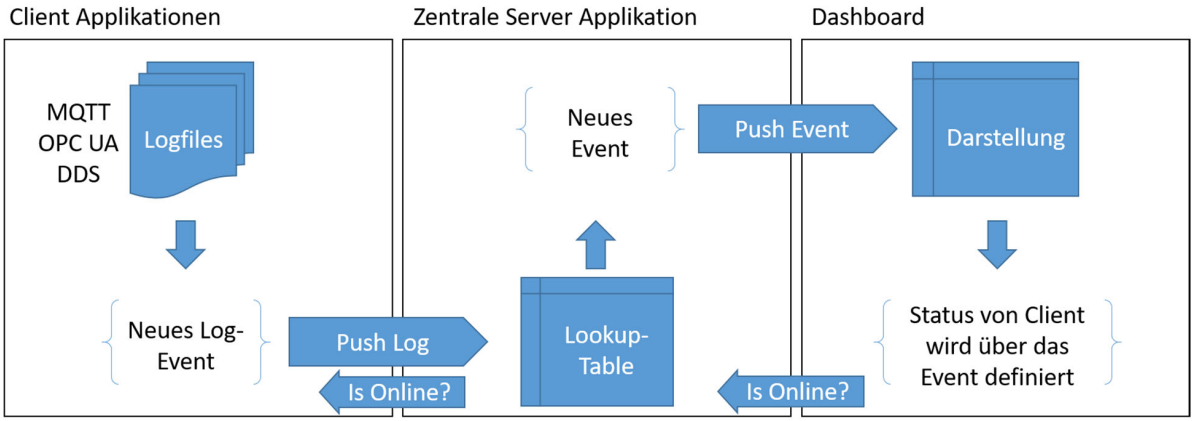


Abbildung 27: Monitoring-Tool

Die Client Applikation(en) sind in der Lage auf Änderungen an Log-Dateien in den angeschlossenen Devices zu reagieren. Für jede Client Applikation wird eine Konfigurationsdatei bereitgestellt, in der alle zu überwachenden und lokal existierenden Log-Dateien definiert und mit einer Log-ID versehen werden können. Client Applikationen können im Hintergrund auf einem beliebigen Ziel-System installiert und ausgeführt werden. Einmal gestartet übermitteln die Client Applikationen jede Änderung die an einer durch sie überwachten Log-Datei vorgenommen wird unmittelbar an die zentrale Backend Applikation. Die Backend Applikation fungiert im Wesentlichen als Vermittler zwischen Client Applikationen und Frontend Applikationen. Dafür verteilt die Backend Applikation die Information aller registrierten Änderungen sukzessive und unmittelbar nach deren Erhalt an alle angeschlossenen Frontend Applikationen. Die Frontend Applikationen zeigen die empfangene Information, also die konkrete Änderung an einer Log-Datei an. Um im Frontend zwischen den verschiedenen Client Applikationen bzw. Log-Dateien unterscheiden zu können, werden die Änderungen über die Log-ID kategorisiert. Die Log-ID ermöglicht also die Zuordnung von Log-Datei, Client und der empfangenen Änderung. Die Log-ID sollte darum möglichst eineindeutig sein und mit Bedacht gewählt werden. Bewährt haben sich Kombinationen aus Client Hostname und Log-File Produzent wie beispielsweise „*Raspberry-1-MQTT*“.

Die Übermittlung der Änderungen beziehungsweise die Kommunikationen zwischen allen Teilnehmenden Software-Komponenten ist mit WebSockets implementiert. Das Frontend kann sowohl über HTTP als auch über HTTPS bereitgestellt werden, da es sich um eine reguläre Singe-Page-Applikation handelt.

3.7.2.2 Weiterentwicklung und Wartung

Node.js ist eine sehr beliebte, schnelle und effiziente Plattform und JavaScript ist als Programmiersprache insbesondere für Anfänger eine gute Wahl. Somit ist in Bezug auf Weiterentwicklung und Wartung der Lösung sichergestellt, dass es keiner aufwändigen Einarbeitung bedarf. Architektonisch liefert das klassische Client-Server Prinzip eine sehr flexible Lösung, um je nach Bedarf Clients anschließen oder abschalten zu können. Natürlich kann, je nach Auslastung der Backend Applikation, eine vertikale Skalierung des Backend Systems notwendig sein. Grundsätzlich wird aber angenommen, dass die Lösung auch mit einem sehr hohen Aufkommen von Änderungen ausreichend zuverlässig funktionieren wird. Sollte das System bei großen Datenaufkommen zu langsam werden, wäre es außerdem möglich die Funktionalität der Client Applikationen und einen Client-seitigen Änderungsfilter zu erweitern. Hierbei stellt sich sowieso die Frage, in wie weit alle und jegliche Änderungen an die Frontend Applikationen kommuniziert werden müssen. In unserem Testaufbau konnten wir diesbezüglich jedoch keine Engpässe beobachten.

4 Demonstrator

Der Demonstrator wurde entwickelt, um die Ergebnisse der vorangegangenen Arbeitspakete zu veranschaulichen. Der sichere Verbindungsaufbau von OPC UA-, MQTT- und DDS-Anwendungen, der Online-Validierungsdienst für X509-Zertifikate (Online Certificate Validation Service, OCSP), die Integration der Rollenverwaltung, die Benutzerauthentifizierung über SmartCard und die Autorisierung über Attributzertifikate werden demonstriert. Das Design und die Implementierung werden in den folgenden Abschnitten erläutert.

4.1 Design des Demonstrators

Abbildung 28 zeigt das Design, bei dem das linke Ende drei verschiedene Sensoren zeigt, die mit drei verschiedenen Raspberry-Pi, Middleware/Framework-Serveranwendungen wie OPC UA, MQTT und DDS über Modbus oder serielle Verbindung verbunden sind.

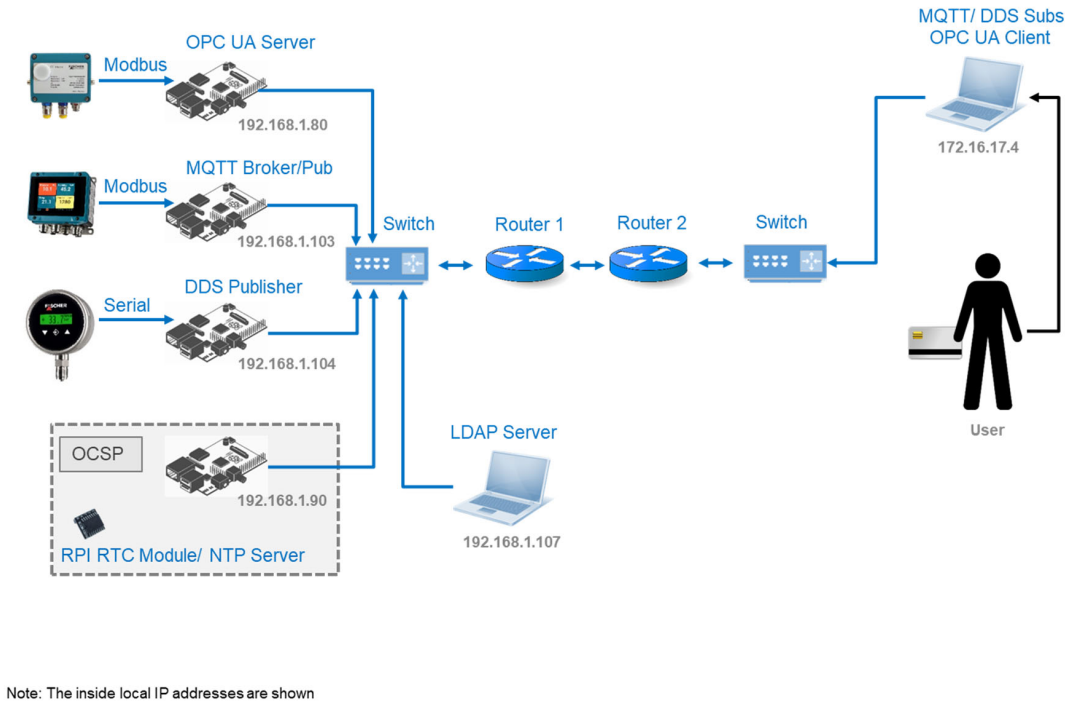


Abbildung 28: Demonstrator Aufbau

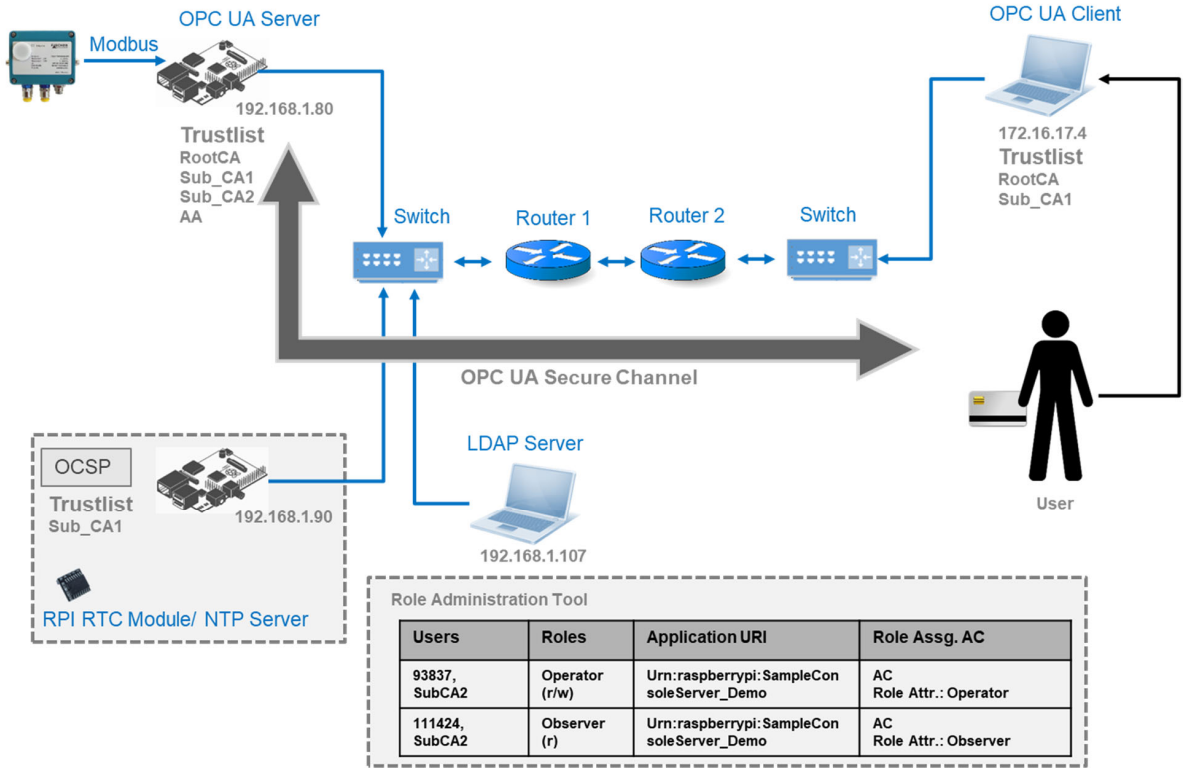
Das rechte Ende zeigt die Client-Anwendungen. Eine einfache Wide-Area-Network (WAN)-Einrichtung wird über zwei Router dargestellt. Die am linken Ende dargestellte Hardware stellte die lokale Einrichtung in einer Fabrik mit Zeitsynchronisation über eine Echtzeituhr (RTC) dar, die als NTP-Server fungiert und somit die Möglichkeit einer Online-Validierung des X509-Zertifikats ermöglicht. Wie in Abbildung 28 zu sehen ist, laufen das RTC-Hardwaremodul und der OCSP-Server in einem weiteren Raspberry-Pi, der mit dem gleichen Switch wie die Serveranwendungen verbunden ist

Um die ACs zu speichern und zu verteilen, werden die Verzeichnisdienste durch Verzeichniszugriffsprotokolle erleichtert. Das Permis-Projekt beschreibt die Anwendung eines Lightweight Directory Access Protocol (LDAP)-Servers zum Speichern und Abrufen von X.509-Zertifikaten [119], [120]. Die Anwendbarkeit des Verzeichnisdienstes ist angemessen, da er das Löschen

einer AC aus dem Verzeichnis direkt ermöglicht, anstatt einen zusätzlichen Widerrufsdienst zu aktivieren. Das LDAP-Protokoll ist ein Anwendungsschichtprotokoll über TCP. Sichere Verbindungen zum LDAP-Server können optional aktiviert werden. Die LDAP-Verzeichnisdienste eignen sich am besten für Szenarien mit "Single Write und Many Reads". Eine solche Anwendung eines LDAP-Servers wird in der Implementierung verwendet. Für diese Arbeit wird der Open-Source-LDAP-Server Apache DS verwendet [121]. Der Open-Source-LDAP-Client Apache Directory Studio wird verwendet, um LDAP-Operationen auf dem Server zu durchsuchen und anzuwenden [122]. Um einen Eintrag innerhalb eines LDAP-Servers zu erstellen, müssen das erforderliche Schema erstellt und die notwendigen Objektklassen- und Attributtypen nach Bedarf angelegt werden. Ein Beispiel für die Erstellung eines solchen Schemas ist in [123] beschrieben. Die ACs werden im LDAP-Server mit dem zuvor genannten Client-Studio gespeichert. Die gespeicherten ACs werden über das Pull-Modell verteilt. Jeder Client, der versucht, auf den LDAP-Server zuzugreifen, muss den Standort des Servers, auf dem der Dienst ausgeführt wird, und die Anmeldeinformationen (falls vorhanden) kennen. Für diese Arbeit wird eine einfache Benutzername-/Passwort-Authentifizierung verwendet, um eine Verbindung über TCP zwischen den LDAP-Client/Server-Anwendungen herzustellen. Auch der LDAP-Server, der die Zertifikate verwaltet, läuft auf einem Laptop, der mit dem gleichen Switch wie Serveranwendungen verbunden ist.

So wird der vollständige sichere Verbindungsaufbau zwischen Remote-Anwendungen unter Verwendung der in AP4 diskutierten Sicherheitsinfrastrukturkonzepte demonstriert.

4.2 Middleware



Users	Roles	Application URI	Role Assg. AC
93837, SubCA2	Operator (r/w)	Urn:raspberrypi:SampleConsoleServer_Demo	AC Role Attr.: Operator
111424, SubCA2	Observer (r)	Urn:raspberrypi:SampleConsoleServer_Demo	AC Role Attr.: Observer

Abbildung 29: OPC UA sicherer Verbindungsaufbau

Abbildung 29: OPC UA sicherer Verbindungsaufbau

zeigt die OPC UA Anwendung. Der OPC UA Server sammelt über Modbus Daten von einem Differenzdruck-Sensor vom Typ DE 43 [124] ein. Dieser Sensor kann neben der Druckdifferenz auch die Temperatur des Mediums erfassen. Der OPC UA Adressraum wurde erstellt, um die Informationen des Sensors und die dynamischen Werte des Sensors aufzunehmen. Die Zugriffskontrollkonfigurationen werden in Bezug auf die Rolle der dynamischen Knoten zur Demonstration der Anwendungsfälle für Attributzertifikate vorgenommen.

Die PKI schafft die notwendige Infrastruktur für sichere Identitäten, die für die Client/Server-Anwendungen konfiguriert werden. Die Certificate Trust List (CTL) wird von OPC UA IEC 62541 Part 4 Services [111] genannt, um die vertrauenswürdigen Zertifikate innerhalb der OPC UA-Anwendungen zu konfigurieren. Zur Veranschaulichung sind die X509-Zertifikate, denen die OPC UA-Anwendungen vertrauen, in Abbildung 29 dargestellt. Der OPC UA Server muss der Attribute Authority vertrauen, auch um die Validierung der ACs zu erleichtern. Der OCSP-Server muss mit Schlüsselpaaren und Zertifikaten von Sub_CA1 ausgestattet sein, um die OCSP-Anfrage zu validieren und die OCSP-Antwort zu signieren. Die in Abbildung 29 dargestellte Tabelle ist eine exemplarische Konfiguration, um das Werkzeug zusammen mit der Einrichtung zu demonstrieren. Beispielsweise identifizieren Seriennummer und Ausstellername (93837, SubCA2) Details eines X509-Zertifikats eines Benutzers den Benutzer eindeutig und es ist das Attribut, mit dem ein Rollenattribut (AC - Role Attr.: Operator, ein AC-haltiges Rollenattribut) einer OPC UA-Anwendung zugeordnet wird, die von der AnwendungURI eindeutig identifiziert wird (urn:raspberrypi:SampleConsoleServer_Demo).

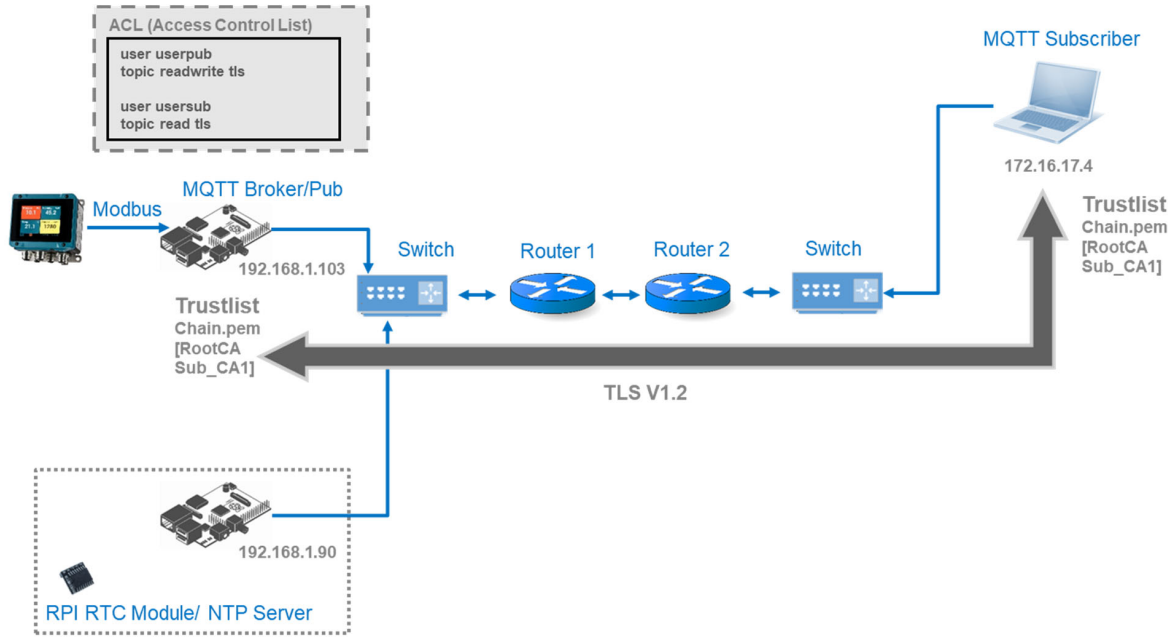


Abbildung 30: MQTT sicherer Verbindungsaufbau

Abbildung 30 zeigt den sicheren Verbindungsaufbau zwischen dem MQTT Broker und den Teilnehmeranwendungen. Der MQTT-Broker sammelt die Daten eines EA 15 [125] Fischer-Sensor,

der über zwei Kanäle zur Messung von Feuchte und Druck verfügt, ein. Der Publisher veröffentlicht die Daten zu dem entsprechenden Topic, das auf dem Broker gehostet wird.

Die MQTT Broker-, Publisher- und Abonnentenanwendungen sind mit sicheren Identitäten und vertrauenswürdigen X509-Zertifikaten konfiguriert. Die TLS Version 1.2 wird innerhalb der MQTT Broker-Konfiguration konfiguriert, um die sichere Verbindung zu Publisher- und Subscriber-Anwendungen zu ermöglichen. Im Falle einer mehrstufigen PKI muss die Vertrauenskette innerhalb der Broker-Konfiguration konfiguriert werden, wie in Abbildung 30 dargestellt. Der Broker kann auch mit einer Zugriffskontrollliste (ACL) konfiguriert werden, z.B. wie in Abbildung 30 dargestellt, kann eine einfache Textdatei erstellt werden, die veranschaulicht, dass ein Benutzername "user-pub" Lese-/Schreibzugriff auf das Thema "tls" haben kann. Ebenso kann ein Benutzer mit dem Namen "usersub" Lesezugriff auf das Thema tls haben.

DDS-Anwendungen wurden auch mit Security-Plugins getestet und die in AP4 gezeigte PKI konnte auf DDS-Anwendungen erweitert werden. Die Implementierungen mit akademischer Lizenz von 'TWINOAKS' wurden zum Testen von DDS-Anwendungen und deren Sicherheits-Plugins verwendet.

5 Schlusswort

Das Projekt konnte insgesamt erfolgreich abgeschlossen werden. An den Standorten Hannover und Lemgo stehen je ein Demonstrator zur Verfügung, an denen sich die implementierte Software vorführen lässt. Mit dem hier vorgelegten Abschlussbericht erfolgt zudem eine umfassende Beschreibung der entwickelten Lösung. Für interessierte Unternehmen steht zudem eine Reihe weiterer Dokumente zur Verfügung, die die Projektinhalte tiefergehend vermitteln. Diese können, genau wie die im Projekt entstandene Software, von allen interessierten Unternehmen direkt bei den Forschungseinrichtungen zur Bereitstellung angefragt werden. Darüber hinaus können die profitieren Unternehmen insbesondere von folgenden Projektergebnissen profitieren:

- Ein umfassender Überblick sowie eine detaillierte technische Bewertung der betrachteten Middleware-Protokolle gemäß Tabelle 4
- Ein Vorschlag für ein Referenzarchitekturmodell für eine Middleware-Kommunikation.
- Eine detaillierte Beschreibung und Auswahl in Frage kommender Protokollstack-Implementierungen.
- Eine Muster-Implementierung für eine protokollneutrale Public-Key-Infrastruktur
- Zwei Demonstratoren zum Nachweis der Funktionalität

Das Projektteam möchte sich an dieser Stelle bei allen Mitgliedern des projektbegleitenden Ausschusses für das Feedback und die anregenden Diskussionen bedanken.

6 Literaturverzeichnis

References

[1] Rysavy, O., Rab, J. u. Sveda, M. 2013 Federated Conference on. 2013, S. 1435–1440. *Improving security in SCADA systems through firewall policy analysis. Computer Science and Information Systems (FedCSIS)*.

[2] McKay, M. 2012 IEEE-IAS/PCA Cement Industry Technical Conference. 2012, S. 1–15. *Best practices in automation security*. https://ieeexplore.ieee.org/document/6215678.

[3] Kurscheid, J. 2013. *International ETG-Congress 2013. Symposium 1: security in critical infrastructures today : 5-6 Nov. 2013*. IEEE, Piscataway, NJ.

[4] Bundesamt für Sicherheit in der Informationstechnik. 2013. *ICS-Security-Kompendium*. https://www.bsi.bund.de/DE/Themen/ICS/Empfehlungen/ICS/empfehlungen_node.html.

[5] Brandle, M. and Naedele, M. 2008. Security for Process Control Systems: An Overview. *IEEE Secur. Privacy* 6, 6, 24–29.

[6] IEC TS 62443-1-1:2009. *Industrial communication networks - Network and system security - Part 1-1: Terminology, concepts and models*.

[7] Stouffer, K., Pillitteri, V., Lightman, S., Abrams, M., and Hahn, A. 2013. *SP800-82: Guide to Industrial Control Systems (ICS) Security*.

[8] Tofino Security. 2012. *Using ANSI/ISA-99 Standards to Improve Control System Security*.

[9] Heiss, S. u. Hausmann, S. In: 9. Magdeburger Maschinenbau-Tage. 2009. *VPN-Implementierungen auf Endgeräten der Automatisierungstechnik*. https://www.th-owl.de/init/en/veroeffentlichungen/publikationen/a/filteron/4/author.html.

[10] Hausmann, S., Brand, J.-C., Miske, A. u. Heiss, S. Laufzeit: 2008 - 2011 = Secure networks (VPN) in automation, 2012. *Sichere Kommunikationsnetze (VPN) in der Automatisierungstechnik. SKAT; Abschlussbericht*.

[11] Kagermann, H., Wahlster, W., and Helbig, J. 2013. *Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0. Abschlussbericht des Arbeitskreises Industrie 4.0. Deutschlands Zukunft als Produktionsstandort sichern.* https://www.bmbf.de/files/Umsetzungsempfehlungen_Industrie4_0.pdf.

[12] Lee, E. A. 2008 11th IEEE International Symposium on. 2008, S. 363–369. *Cyber Physical Systems: Design Challenges*. https://ieeexplore.ieee.org/document/4519604.

[13] Houyou, A. M., Huth, H.-P., Kloukinas, C., Trsek, H., and Rotondi, D. 2012 IEEE 17th Conference on. 2012, S. 1–7. *Agile manufacturing: General challenges and an IoT@Work perspective*.

[14] Imtiaz, J. and Jasperneite, J. 2013 11th IEEE International Conference on. 2013, S. 500–505. *Scalability of OPC-UA down to the chip level enables "Internet of Things"*.

[15] DKE Deutsche Kommission Elektrotechnik, Elektronik Informationstechnik in DIN und VDE. 2013. *Die deutsche Normungs-Roadmap Industrie 4.0,*. https://www.din.de/blob/95954/97b71e1907b0176494b67d8d6d392c54/aktualisierte-roadmap-i40-data.pdf.

[16] Runde, M., Hausmann, S., Tebbe, C., Czybik, B., Niemann, K.-H., Heiss, S., Jasperneite, J., Hochschule Hannover, and Hochschule Ostwestfalen-Lippe. 2014. *SEC_PRO : sichere Produktion mit verteilten Automatisierungssystemen*.

[17] PROFIBUS Nutzerorganisation e.V. *„Security Erweiterungen für PROFINET“: PI White Paper für PROFINET*. https://www.profibus.com/download/pi-white-paper-security-extensions-for-profinet/.

[18] Niemann, K.-H. 22 - 25 July, 2019. “IT-Security extensions for PROFINET”. *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)* (22 - 25 July, 2019), 407–412.

[19] Friesen, M., Karthikeyan, G., Steinke, K., Niemann, K.-H., Heiss, S., and Wisniewski, L. Automation 2018. *„Sichere Middleware-Lösungen für die Industrie 4.0 - Eine IT-Sicherheitsanalyse aktueller Kommunikationsansätze“*. https://www.researchgate.net/publication/323656205_Sichere_Middleware-Losungen_fur_die_Industrie_40_-_Eine_IT-Sicherheitsanalyse_aktueller_Kommunikationsansatze.

[20] Karthikeyan, G. and Heiss, S. ETFA 2018. *PKI and User Access Rights Management for OPC UA based Applications*. https://ieeexplore.ieee.org/document/8502603.

[21] Friesen, M., Karthikeyan, G., Heiss, S., Wisniewski, L., and Trsek, H. KommA 2018. *“A comparative evaluation of security mechanisms in DDS, TLS and DTLS”*. https://www.researchgate.net/publication/329035319_A_comparative_evaluation_of_security_mechanisms_in_DDS_TLS_and_DTLS.

[22] Steinke, K., Tebbje, S., and Niemann, K.-H. AALE 2019. *„Änderung von IT-Security-Anforderungen durch den Wandel der Automatisierungsstrukturen im Kontext von Industrie 4.0“*. https://www.vde-verlag.de/proceedings-de/564860014.html.

[23] Tebbje, S., Niemann, K.-H., Friesen, M., Karthikeyan, G., Heiss, S., Trsek, H., Jänicke, L., and Meyer, C. Automation 2019. *Entwicklung einer IT-Sicherheitsinfrastruktur für verteilte Automatisierungssysteme*. https://www.researchgate.net/publication/334330574_Entwicklung_einer_IT-Sicherheitsinfrastruktur_fur_verteilte_Automatisierungssysteme_-_Integraton_verschiedener_Middleware-Losungen_in_eine_durch_Attributzertifikate_erweiterte_Public-Key-Infrastruktur.

[24] Karthikeyan, Gajasri; Heiss, Stefan (2019): *Enhancing Authorization Mechanisms using Attribute Certificates for OPC UA based Applications* https://ieeexplore.ieee.org/document/8972148 .

[25] Maxim Friesen. Projektarbeit, Technische Hochschule OWL, 2019 (nicht veröffentlicht). *“Evaluation of Certificate Provisioning Protocols for Public Key Infrastructures in Distributed Systems”.*

[26] Kai Steinke. Masterarbeit, Hochschule Hannover, 2018 (nicht veröffentlicht). *„Entwurf einer IT-Security Referenzarchitektur für Industrie 4.0 basierte Systeme“*.

[27] Tebbje, S. Masterarbeit, Hochschule Hannover, 2019 (nicht veröffentlicht). *„Evaluierung eines einheitlichen Sicherheitskonzeptes für die Middleware-Protokolle MQTT und DDS“*.

[28] Mahnke, W., Leitner, S.-H., and Damm, M. 2009. *OPC Unified Architecture*. Springer, Berlin.

[29] Plattform Industrie 4.0 Bundesministerium für Wirtschaft und Energie (BMWi). 2017. *Sichere Kommunikation für Industrie 4.0*. https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/hm-2018-sichere-kommunikation.pdf?__blob=publicationFile&v=4.

[30] PC Control. 2012. *Full-load test stand ensures smooth commissioning of offshore wind turbines*. https://www.pc-control.net/pdf/special_wind_2012/solutions/pcc_special_wind_2012_areva_e.pdf. Accessed 30 January 2018.

[31] OPC Foundation. *OPC UA*. https://opcfoundation.org/about/opc-technologies/opc-ua/. Accessed 17 September 2017.

[32] DIN IEC 62541-5:2015. *OPC Unified Architecture - Teil 5: Informationsmodell 2015*.

[33] OPC Foundation. *OPC Foundation Announces support of Publish / Subscribe for OPC UA*. https://opcfoundation.org/news/opc-foundation-news/opc-foundation-announces-support-of-publish-subscribe-for-opc-ua/. Accessed 21 September 2017.

[34] BR Automation. *Echtzeitfähige OPC UA*. https://www.br-automation.com/en/about-us/monthly-newsletter/latest-news/real-time-capable-opc-ua/. Accessed 21 September 2017.
[35] OPC Foundation. *Unified Architecture Teil 14: PubSub*. https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-14-pubsub. Accessed 21 September 2017.
[36] DIN IEC 62541-2:2016. *OPC Unified Architecture - Part 2 Security Model*. https://www.vde-verlag.de/iec-normen/223904/iec-tr-62541-2-2016.html.
[37] DIN IEC 62541-4:2015. *OPC Unified Architecture - Teil 4: Dienstleistungen*.
[38] D. Cooper, S. Santesson, S. farrell, S. Boeyen, R. Housley, and W. Polk. 2008. *RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. https://tools.ietf.org/html/rfc5280.
[39] S. Sanesson, M. Myers, and R. Ankney. 2013. *RFC 6960: X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*. https://tools.ietf.org/html/rfc6960.
[40] M. Nystrom and B. Kaliski. 2000. *RFC 2986: PKCS #10: Certification Request Syntax Spezifikation Version 1.7*. https://tools.ietf.org/html/rfc2986.
[41] DIN IEC 62541-3:2015. *OPC Unified Architecture - Teil 3: Adressraummodell*.
[42] K. Herron. *Eclipse Milo*. https://projects.eclipse.org/projects/iot.milo. Accessed 16 August 2017.
[43] OPC Unified Architecture. *open62541*. https://github.com/open62541/open62541. Accessed 16 August 2017.
[44] OPC Unified Architecture. *LGPL Pure Python OPC-UA Client und Server*. https://github.com/FreeOpcUa/python-opcua. Accessed 16 August 2017.
[45] OPC Unified Architecture. *Prosys OPC*. https://www.prosysopc.com. Accessed 16 August 2017.
[46] Softing AG. https://industrial.softing.com/en/news/news-details/article//softing-industrial-offers-commercial-license-for-opc-ua-net-standard-stack.html. Accessed 6 November 2017.
[47] Andrew Banks (IBM), Rahul Gubta (IBM). 2014. *MQTT Version 3.1.1*. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/csprd02/mqtt-v3.1.1-csprd02.html.
[48] Rahul Gubta. 2014. *5 Things to Know About MQTT*. https://www.ibm.com/developerworks/community/blogs/5things/entry/5_things_to_know_about_mqtt_the_protocol_for_internet_of_things?lang=en. Accessed 8 September 2017.
[49] Hillar, G. C. 2017. *MQTT essentials. A lightweight IoT protocol : the preferred IoT publish-subscribe lightweight messaging protocol*. Packt Publishing, Birmingham, UK.
[50] B. Boyd, J. Gauci, M. Robertson, N. van Duy, and R. Gupta. *Building Real-time Mobile Solutions with MQTT and IBM MessageSight*. http://www.redbooks.ibm.com/redbooks/pdfs/sg248228.pdf. Accessed 7 September 2017.
[51] Thomas Bayer. *MQTT, Das M2M und IoT Protokoll*. https://www.predic8.de/mqtt.htm. Accessed 4 March 2016.
[52] IBM Knowledge Center. *MQTT Security*. https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_7.5.0/com.ibm.mm.tc.doc/tc00150_.htm. Accessed 7 September 2017.
[53] Zamfir, S., Balan, T., Iliescu, I., and Sandu, F. 2016. *A security analysis on standard IoT protocols*.
[54] Eclipse Foundation. *Eclipse Mosquitto - An open source MQTT broker*. https://mosquitto.org/.
[55] HiveMQ. *Enterprise MQTT Broker*. http://www.hivemq.com. Accessed 21 September 2017.
[56] Eclipse Foundation. *Eclipse Paho*. http://www.eclipse.org/paho. Accessed 21 September 2017.
[57] Object Management Group (OMG). 2015. *Data Distribution Service Specification Version 1.4*. https://www.omg.org/spec/DDS/About-DDS/.

[58] Twin Oaks Computing, I. *What can DDS do for You?* http://www.omg.org/hot-topics/documents/dds/CoreDX_DDS_Why_Use_DDS.pdf. Accessed 4 September 2017.
[59] Gilchrist, A. 2016. *Industry 4.0*. Apress, Berkeley, CA.
[60] Real Time Innnovations (RTI). *DDS Standard*. https://www.rti.com/products/dds/omg-dds-standard. Accessed 4 September 2017.
[61] DDS Foundation. *What is DDS?* http://portals.omg.org/dds/what-is-dds-3/. Accessed 12 September 2017.
[62] Gerardo Pardo-Castellote, Bert Farabaugh, Rick Warren. 2005. *An Introduction to DDS and Data- Centric Communications*. https://studylib.net/doc/11266839/an-introduction-to-dds-and-data-centric-communications-ge…
[63] Paolo Bellavista, Antonio Corradi, Luca Foschini, Alessandro Pernafini. 2013. *Data Distribution Service (DDS): A performance comparison of OpenSplice and RTI implementations*. https://www.semanticscholar.org/paper/Data-Distribution-Service-(DDS)%3A-A-performance-of-Bellavista-Corradi/168b1a33aebf090350718c0af31150d5df692da2.
[64] DDS Foundation. *Why Choose DDS?* http://portals.omg.org/dds/why-choose-dds/. Accessed 8 September 2017.
[65] Prismtech. *Was sind die Vorteile von DDS?* http://www.prismtech.com/ vortex/technologies/what-are-benefits-dds. Accessed 7 September 2017.
[66] OpenDDS. *Nutzung der QoS-Richtlinien*. http://opendds.org/about/qosusages.htm. Accessed 14 September 2017.
[67] DDS Foundation. *Technical Benefits*. http://portals.omg.org/dds/key-technical-benefits. Accessed 5 September 2017.
[68] Prismtech. *Vortex Lite Webcast vorstellen*. https://blog.prismtech.com/2015/01/21/introducing-vortex-lite-webcast/. Accessed 24 September 2017.
[69] DDS Foundation. *"Where Can I Get DDS?"*. https://www.dds-foundation.org/where-can-i-get-dds/. Accessed 8 September 2017.
[70] Tebbje, S., Steinke, K., Niemann, K.-H., Friesen, M., Karthikeyan, G., and Heiss, S. Internes Dokument, Hochschule Hannover, Technische Hochschule OWL, 2018 (nicht veröffentlicht). *Description and evaluation of middleware solutions*.
[71] DIN SPEC 91345. *Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0)*.
[72] Heidel, R., Hoffmeister, M., Hankel, M., and Döbrich, U. 2017. *Industrie4.0 Basiswissen RAMI4.0. Referenzarchitekturmodell mit Industrie4.0-Komponente*. VDE Verlag GmbH; Beuth Verlag GmbH, Berlin, Wien, Zürich.
[73] M. Hankel. 2015. *Das Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0)*. https://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/zvei-faktenblatt-rami.pdf?__blob=publicationFile&v=3.
[74] S-W. Lin, B. Miller, J. Durand, G. Bleakley, A. Chigani, R. Martin, B. Murphy, and M. Crawford. 2017. *Industrial Internet Consortium. The Industrial Internet of Things Volume G1: Reference Architecture*. https://www.iiconsortium.org/IIC_PUB_G1_V1.80_2017-01-31.pdf.
[75] ISO/IEC/IEEE 42010:2011. *Systems and software engineering – Architecture description*. https://www.iso.org/standard/50508.html.
[76] M. Heidrich and J. Luo. 2016. *Industrial Internet of Things: Referenzarchitektur für die Kommunikation*. https://www.iks.fraunhofer.de/content/dam/esk/dokumente/Whitepaper_IoT_dt_April16.pdf.
[77] BR Automation. *Real-time capable OPC UA*. https://www.br-automation.com/en/about-us/newsletter/latest-news/real-time-capable-opc-ua/.
[78] K.-H. Niemann. 2017. *IT-Security-Konzepte für die Prozessindustrie: Anforderungen im Kontext von Industrie 4.0*. http://media.di-ver-lag.de/atp/atp_07-08_2014_Niemann.pdf.

[79] R. Neugebauer, M. Jarke, and K. Thoma. 2014. *Strategie- und Positionspapier Cyber-Sicherheit 2020: Herausforderungen für die IT-Sicherheitsforschung*. https://www.iese.fraunhofer.de/content/dam/iese/de/dokumente/Fraunhofer-Strategie-und_Positionspapier_Cyber-Sicherheit2020.pdf.

[80] IEC 62443-1-1:2009. *Industrielle Kommunikationsnetze-IT-Sicherheit für Netze und Systeme-Teil 1-1: Terminilogie, Konzepte und Modelle*.

[81] IEC 62443-3-3:2015-06. *Industrielle Kommunikationsnetze-IT-Sicherheit für Netze und Systeme-Teil 3-3-: Systemanforderungen zur IT-Sicherheit und Security-Level*.

[82] IEC 62443-4-2:2017-10. *Industrielle Kommunikationsnetze-IT-Sicherheit für Netze und Systeme-Teil 4-2: Anforderungen an Komponenten industrieller Automatisierungssysteme*.

[83] Klettner, C., Tauchnitz, T., Epple, U., Nothdurft, L., Diedrich, C., Schröder, T., Goßmann, D., Banerjee, S., Krauß, M., Latrou, C., and Urbas, L. 2017. Namur Open Architecture. *atp* 59, 01-02, 17.

[84] NAMUR-Interessengemeinschaft Automatisierungstechnik der Prozessindustrie e.V. 2015. *NAMUR-Empfehlung NE 153: „Automation Security 2020: Design, Implementierung und Betrieb industrieller Automatisierungssysteme“*.

[85] VDI/VDE 2182 Blatt 1: Informationssicherheit. 2011. *in der industriellen Automatisierung-Allgemeines Vorgehensmodell*.

[86] NAMUR-Interessengemeinschaft Automatisierungstechnik der Prozessindustrie e.V. 2016. *Position paper: An Ethernet communication system for the process industry*.

[87] BSI-Bundesamt für Sicherheit in der Informationstechnik. 2016. *Anforderungsdokument Cloud Computing (C5): Kriterien zur Berurteilung der Informationssicherheit von Cloud-Diensten*. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Broschueren/Anforderungskatalog-Cloud_Computing-C5.pdf?__blob=publicationFile&v=3.

[88] ZVEI-Zentralverband Elektrotechnik- und Elektronikindustrie e.V. 2017. *Orientierungsleitfaden für Hersteller zur IEC 62443*. https://www.zvei.org/fileadmin/u-ser_upload/Presse_und_Medien/Publikationen/2017/April/Orientierungsleitfa-den_fuer_Hersteller_IEC_62443/Orientierungsleitfaden_fuer_Hersteller_IEC_62443.pdf.

[89] BSI-Bundesamt für Sicherheit in der Informationstechnik. 2016. *Sichere Nutzung von Cloud-Diensten: Schritt für Schritt von der Strategie bis zum Vertragsende*. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Broschueren/Sichere_Nutzung_Cloud_Dienste.pdf?__blob=publicationFile&v=10.

[90] D.F. Ferraiolo and D.R. Kuhn. 1992. Role-Based Access Controls. In *15th National Computer Security Conference*.

[91] Tsolkas, A. and Schmidt, K. 2017. *Rollen und Berechtigungskonzepte. Identity- und Access-Management im Unternehmen*. Edition <kes>. Springer Vieweg, Wiesbaden.

[92] OPC Foundation. *Spcification Part 2: Security Model*. https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-2-security-model/. Accessed 5 February 2018.

[93] R. Barnes, J. Hoffman-Andrews, D. McCarney, and J. Kasten. 2019. *RFC 8555: Automatic Certificate Management Environment (ACME)*. https://tools.ietf.org/html/rfc8555.

[94] J. Young and A. Honore. 2016. *Simple Certificate Enrollment Protocol Overview*. https://www.cisco.com/c/en/us/support/docs/security-vpn/public-key-infrastructure-pki/116167-technote-scep-00.html. Accessed 7 August 2019.

[95] C. Adams, S. Farrell, T. Kause and T. Mononen. 2005. *RFC 4210: Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)*. https://tools.ietf.org/html/rfc4210.

[96] J. Schaad and M. Myers. 2008. *RFC 5272: Certificate Management over CMS (CMC)*. https://tools.ietf.org/html/rfc5272.

[97] M. Pritikin, P. Yee and D. Harkins. 2013. *RFC 7030: Enrollment over Secure Transport*. https://tools.ietf.org/html/rfc7030.

[98] B. Kaliski. 1998. *RFC 2315: PKCS #7: Cryptographic Message Syntax Version 1.5*. https://tools.ietf.org/html/rfc2315.

[99] M. Myers and C. Adams. 1999. *RFC 2511: Internet X.509 Certificate Request Message Format*. https://tools.ietf.org/html/rfc2511.

[100] Federal Ministry for Economic Affairs and Energy (BMWi). 2016. *Technical Overview: Secure Identities*. https://www.plattform-i40.de/I40/Redaktion/EN/Downloads/Publikation/secure-identities.pdf?__blob=publicationFile&v=7.

[101] Federal Ministry for Economic Affairs and Energy BMWi. 2016. *Technical Overview: Secure cross-company communication*. https://www.plattform-i40.de/I40/Redaktion/EN/Downloads/Publikation/secure-cross-company-communication.pdf?__blob=publicationFile&v=5.

[102] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley and W. Polk. 2008. *RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. https://tools.ietf.org/html/rfc5280.

[103] International Telecommunication Union. 2016. *X.509: Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate framework*. https://www.itu.int/rec/T-REC-X.509.

[104] Wallis, K., Kemmer, F., Jastremskoj, E., and Reich, C. 2017. *Adaption of a Privilege Management Infrastructure (PMI) Approach to Industry 4.0*. https://ieeexplore.ieee.org/document/8113778.

[105] S. Farrell, R. Housley and S. Turner. 2010. *RFC 5755: An Internet Attribute Certificate Profile for Authorization*. https://tools.ietf.org/html/rfc5755.

[106] J. A. M. Montes. *Open PMI project*.

[107] American National Standards Institute, Inc. 2004. *INCITS 359-2012 (R2017): Information technology - Role Based Access Control*. https://webstore.ansi.org/Standards/INCITS/INCITS3592012R2017?source=blog.

[108] V. C. Hu, D. Ferraiolo, R. Kuhn and A. Schnitzer. 2017. *SP 800-162: Guide to Attribute Based Access Control (ABAC) Definition and Considerations*. https://csrc.nist.gov/publications/detail/sp/800-162/final.

[109] S. Verma, M. Singh and S. Kumar. 2012. *Comparative analysis of Role Base and Attribute Base Access Control Model in Semantic Web*. https://pdfs.semanticscholar.org/7750/f0bffaff9c3bb66fa7b8dfef2f46daf0525e.pdf.

[110] Platform Industrie 4.0. 2018. *Zugriffssteuerung fuer Industrie 4.0- Komponenten zur Anwendung von Herstellern, Betreibern und Integratoren*. https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/zugriffssteuerung-industrie40-komponenten.html.

[111] OPC Foundation. 2017. *OPC Unified Architecture Specification- Part 4: Services*. https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-4-services.

[112] Tebbje, S. Masterarbeit, Hochschule Hannover, 2019 (nicht veröffentlicht). *„Evaluierung eines einheitlichen Sicherheitskonzeptes für die Middleware-Protokolle MQTT und DDS“*.

[113] Eclipse. *“Eclipse Mosquitto™ An open source MQTT broker”*. https://mosquitto.org/.

[114] Herron, K. *“Eclipse Milo - an open source implementation of OPC UA (IEC 62541) ”*. https://github.com/eclipse/milo.

[115] Twin Oaks Cimputing Inc. *“CoreDX DDS”*. http://www.twinoakscomputing.com/.

[116] PROFIBUS Nutzerorganisation. 2019. *Security Erweiterungen für PROFINET PI White Paper für PROFINET. Version 1.05,*. https://www.profibus.com/index.php?eID=dumpFile&t=f&f=87330&token=3047389d9d4da992386aac997c7141fe5f75a9b0.

[117] Runde, M., Hausmann, S., Tebbe, C., Czybik, B., Niemann, K.-H., Heiss, S., and Jasperneite, J. 2014. *SEC_PRO - Sichere Produktion mit verteilten Automatisierungssystemen. Schlussbericht für das FHprofUnt-Forschungsprojekt mit dem FKZ 1760A10 sowie 17060B10*. https://serwiss.bib.hs-hannover.de/frontdoor/index/index/docId/499.

[118] Runde, M. Dissertation, Magdeburg, 2014. *Echtzeitfähige Protokollerweiterung zum Schutz Ethernet-basierter Automatisierungskomponenten.* https://d-nb.info/1057913936/34.

[119] J. Sermersheim. 2006. *RFC 4511: Lightweight Directory Access Protocol (LDAP): The Protocol*. https://tools.ietf.org/html/rfc4511.

[120] Chadwick, D. W., Otenko, A., and Ball, E. 2003. *Role-based access control with X.509 attribute certificates*.

[121] Apache Software Foundation, “ApacheDS,” [Online]. Available: https://directory.apache.org/apacheds/. [Accessed 12 8 2019]. *ApacheDS*. https://directory.apache.org/apacheds. Accessed 12 August 2019.

[122] Apache Software Foundation. *Apache Directory Studio*. https://directory.apache.org/studio. Accessed 12 August 2019.

[123] S. Otenko, D. Chadwick and E. Thornton. *PERMIS Java API Cookbook*. http://sec.cs.kent.ac.uk/permis/documents/PERMISAPICookbook.html. Accessed 2 December 2018.

[124] Fischer Mess-und RegelTechnik. *DE43*. https://www.fischermesstechnik.de/de/produkte/details/de43. Accessed 12 August 2019.

[125] Fischer Mess-und RegelTechnik. *EA 15*. https://www.fischermesstechnik.de/de/produkte/details/ea15. Accessed 12 August 2019.