Hochschule Wismar
University of Applied Sciences
Technology, Business and Design
Fakultät für Ingenieurwissenschaften
Bereich Elektrotechnik & Informatik

Master-Thesis

Schwachstellenanalyse von Funkprotokollen am Beispiel von Smart Home Anwendungen

Eingereicht am:	11. Oktober 2019
von:	Sebastian Pflaum geboren am in
Studiengang:	IT-Sicherheit und Forensik
Matrikel-Nr:	240646
Betreuer:	Prof. Dr.-Ing. habil. Andreas Ahrens
Zweitbetreuer:	Prof. Dr.-Ing. Antje Raab-Düsterhöft

Aufgabenstellung

Die vorliegende Arbeit beschäftigt sich mit der Schwachstellenanalyse von Funkprotokollen am Beispiel von Smart Home Anwendungen. Das Thema der Arbeit grenzt die Protokolle entsprechend auf die Prokolle ein, die in Smart Home Geräten zur Anwendung kommen. Da eine Analyse aller Smart Home Funkprotokolle den Rahmen dieser Ausarbeitung überschreiten würde, wurde sich auf die drei am häufigsten genutzten Protokolle, nämlich ZigBee, Z-Wave und Bluetooth, konzentriert. Die Arbeit gibt hierbei zunächst ein Grundverständnis in die Thematik. Für jedes Funkprotokoll folgt eine Darstellung der Funktionsweise, die in eine Analyse der Sicherheitsmaßnahmen übergeht. Basierend auf diesen werden dann Bedrohungen durch die Definition von möglichen Angriffsvektoren dargestellt. Abschließend folgt jeweils eine Abschlussanalyse. Darüber hinaus wird für das Funkprotokoll der ZigBee Spezifikation eine Testumgebung aufgebaut, die der Durchführung eines Angriffs und der Analyse des Funkprotokolls dient.

Abstract

The present thesis deals with the weak point analysis of radio protocols exemplified by Smart Home applications. The topic of this study limits the used protocols to those used in smart home devices. As an analysis of all Smart Home radio protocols would exceed the scope of this paper, the focus was on the three most commonly used protocols, ZigBee, Z-Wave and Bluetooth. The paper first gives a basic understanding of the topic. For each radio protocol, a description of its functionality is given, which is followed by an analysis of the security measures. Based on these, threats are then represented by the definition of possible attack vectors. Finally, a concluding analysis follows. In addition, a test environment is set up for the radio protocol of the ZigBee specification, which serves the execution of an attack and the analysis of the radio protocol.

Inhaltsverzeichnis

1 Einleitung **7**
1.1 Motivation . . . 7
1.2 Ziele der Arbeit . . . 8
1.3 Inhaltlicher Aufbau der Arbeit . . . 9

2 Grundlagen **10**
2.1 Das Internet der Dinge . . . 10
2.1.1 Einführung und Definition . . . 10
2.1.2 Grundlegende Architektur . . . 13
2.1.2.1 Perception Layer . . . 16
2.1.2.2 Network Layer . . . 16
2.1.2.3 Application Layer . . . 16
2.1.3 Sicherheitsherausforderungen und -schwachstellen . . . 17
2.1.3.1 Perception Layer . . . 17
2.1.3.2 Network Layer . . . 18
2.1.3.3 Application Layer . . . 20
2.2 Das ISO/OSI- und TCP/IP-Referenzmodell . . . 21
2.3 Netzwerktopologien . . . 24
2.4 Schwachstellenanalyse und Penetrationstests . . . 26
2.4.1 Was ist ein Penetrationstest? . . . 27
2.4.2 Schwachstellenanalyse versus Penetrationstest . . . 29
2.5 Bedrohungsmodellierung . . . 30
2.5.1 Bestimmung der Angriffsvektoren . . . 31
2.5.2 „DREAD“-Modell . . . 33

3 Vorgehensweise **35**

4 Schwachstellenanalyse - Vorbereitung **38**

5 Schwachstellenanalyse - ZigBee **40**
5.1 Informationsbeschaffung . . . 40
5.1.1 Einführung in das Funkprotokoll . . . 40
5.1.2 Funktionsweise / Grundlagen des Funkprotokolls . . . 41
5.1.2.1 ZigBee Protokollstack . . . 41
5.1.2.2 Logische Geräte nach der ZDO Definition . . . 44
5.1.2.3 Netzwerktopologien . . . 45
5.1.2.4 Applikationsprofile . . . 47
5.1.3 Sicherheitsmaßnahmen . . . 47
5.1.3.1 Sicherheitsmodell . . . 48
5.1.3.2 Sicherheitsannahmen . . . 49

5.1.3.3 Sicherheitsarchitektur 50
5.1.3.4 AES-CCM* 53
5.1.3.5 Sicherheitsschlüssel 53
5.1.3.6 Secure Network Join Prozedur 56
5.1.3.7 Lebensdauer der Sicherheitsschlüssel 57
5.1.3.8 Aktualisierung des Network Key 58
5.1.3.9 End Device Aging Mechanismus 59
5.1.4 Schwachstellen / Bestimmung der Angriffsvektoren 59
5.1.4.1 Key Sniffing 60
5.1.4.2 Replay Angriff 61
5.1.4.3 Insecure Rejoin 62
5.1.4.4 Hijacking 64
5.2 Bewertung der Informationen / Risikoanalyse 66
5.3 Aktive Eindringversuche 68
5.4 Abschlussanalyse 71

6 Schwachstellenanalyse - Z-Wave 73
6.1 Informationsbeschaffung 73
6.1.1 Einführung in das Funkprotokoll 73
6.1.2 Funktionsweise / Grundlagen des Funkprotokolls 74
6.1.2.1 Z-Wave Protokollstack 74
6.1.2.2 Home ID / Node ID 77
6.1.2.3 Komponenten 78
6.1.2.4 Routing 79
6.1.2.5 Beaming 80
6.1.3 Sicherheitsmaßnahmen 80
6.1.3.1 Sicherheitsklassen und Network Keys 81
6.1.3.2 Schlüsselaustausch 81
6.1.3.3 Aktiver Schutz vor Angriffen 82
6.1.4 Schwachstellen / Bestimmung der Angriffsvektoren 82
6.1.4.1 Reconnaissance 83
6.1.4.2 Z-Shave 83
6.2 Bewertung der Informationen / Risikoanalyse 87
6.3 Abschlussanalyse 88

7 Schwachstellenanalyse - Bluetooth 89
7.1 Informationsbeschaffung 89
7.1.1 Einführung in das Funkprotokoll 89
7.1.2 Funktionsweise / Grundlagen des Funkprotokolls 91
7.1.2.1 Bluetooth Protokollstack 91
7.1.2.2 Netzwerktopologien 93
7.1.2.3 Frequenzbereich/-spektrum 95
7.1.2.4 Bluetooth Pakete 97
7.1.3 Sicherheitsmaßnahmen 100
7.1.3.1 Sicherheitsarchitektur 101
7.1.3.2 Sicherheitsmanager 102

7.1.3.3 Bluetooth Device Address 102
7.1.3.4 Sicherheitsmodi BR/EDR 103
7.1.3.5 Sicherheitsstufen 105
7.1.3.6 Sicherheitsmodi LE 105
7.1.3.7 Schlüsselarten 106
7.1.3.8 Pairing und Link Key Generierung 107
7.1.4 Schwachstellen / Bestimmung der Angriffsvektoren 111
7.1.4.1 SSP Association Model „Just Works“ 112
7.1.4.2 Authentication Requests 112
7.1.4.3 Jamming 113
7.2 Bewertung der Informationen / Risikoanalyse 114
7.3 Abschlussanalyse 117

8 Schlussbemerkung 118

Literaturverzeichnis 119

Abbildungsverzeichnis 134

Listings 136

Tabellenverzeichnis 137

Glossar 138

Abkürzungsverzeichnis 140

A Penetrationstests 145
A.1 Ziele von Penetrationstests 145
A.2 Ansätze für Penetrationstests 146
A.3 Klassifizierung von Penetrationstests 148
A.3.1 Informationsbasis 149
A.3.2 Aggressivität 149
A.3.3 Umfang 150
A.3.4 Vorgehensweise 151
A.3.5 Technik 151
A.3.6 Ausgangspunkt 152
A.4 Methodologie zur Durchführung von Schwachstellenanalysen und Penetrationstests 153
A.4.1 Anforderungen an die Methodologie 153
A.4.2 Schlüsselkonzept der Schwachstellenanalyse 154
A.4.3 Phasen eines Penetrationstests 155

B Einrichtung des Penetrationstestlabors 158
B.1 Hardware zur Funkanalyse 158
B.1.1 APIMote (for ZigBee sniffing and transmission) 158
B.1.2 CC2531 USB Dongle 159
B.1.3 AVR Raven 159

B.2 Betriebssystem und Angriffswerkzeuge zur Funkanalyse 160
B.2.1 AttifyOS1.3 160
B.2.2 Killerbee / Attify ZigBee Framework 160

Selbstständigkeitserklärung **162**

1 Einleitung

1.1 Motivation

Smart Home und das Internet der Dinge (IdD), auch als Internet of Things (IoT) bezeichnet, sind unwiderruflich miteinander verbunden. Das IdD ist eines der beliebtesten Trends im Moment, der eine große Anzahl traditioneller Industrien einbezieht. Eine Umfrage über die Nutzung von über das Internet steuerbaren Geräte aus dem Jahr 2017 verdeutlicht diesen Trend. Demnach steuern mehr als 50 Prozent der 986 Befragten, im Alter von 18 bis 59 Jahren, ihren Fernseher über das Internet. Schon auf den nachfolgenden Plätzen werden Geräte wie Glühbirnen oder Steckdosen, aber auch Überwachungskameras genannt. (vgl. Abbildung 1.1) Die weltweit steigende Anzahl der angeschlossenen Sensoren führt zu immer mehr Daten. Diese Sensoren und Aktoren sind das, was wir im Allgemeinen IdD nennen. Die Anwendung des IdD im Rahmen eines privaten Haushalts ist das, was allgemein als Smart Home bezeichnet wird. Hierbei sind die Möglichkeiten, Prozesse im häuslichen Umfeld zu automatisieren nahezu unbegrenzt. Doch wie sieht es mit der Sicherheit der eingesetzten Geräte aus? „Besitzer von Smart Home Geräten gehen häufig unvorsichtig mit der Sicherheit der Geräte um [...]. Dabei sind sie häufig gar nicht das Ziel, sondern lediglich der Weg hinein.“ [1] Denn ob Funksteckdosen, Glühbirnen, Heizkörperthermostate oder Fernseher - smarte Geräte wissen viel über deren Nutzer. So können diese nicht nur Auskunft über deren Interessen oder Konsumgewohnheiten geben, sondern erzählen auch, wann ihre Besitzer nicht zu Hause sind. Werden die Geräte länger nicht genutzt, so lässt sich erahnen, dass deren Besitzer sich möglicherweise gerade im Urlaub befinden. Neben diesen Informationen, welche für einen klassischen Einbruch relevant wären, können Smart Home Geräte auch als Einfallstor ins Heimnetzwerk genutzt werden. Vernetzte Rechner werden so ein leichtes Ziel um diese mit Erpressungssoftware, sogenannte Ransomware, zu infizieren oder aber auch das gesamte Netzwerk in ein Bot-Netzwerk zu integrieren. Eine weitere Angriffsfläche bieten Smart Home Geräte durch die Nutzung einer drahtlosen physischen Kommunikation. Diese ermöglicht es Angreifern, die Kommunikation leichter abzufangen. Zusammen mit dem IdD führt dies zu beispiellosen Möglichkeiten für

Angreifer, vertrauliche Informationen preiszugeben und Daten zu manipulieren. Es ist somit entscheidend, effiziente und effektive Methoden zu finden, um solchen Angriffen entgegenzuwirken. Anderenfalls würden alle Vorteile, die das IdD vor allem in Bezug auf Smart Home bietet, verlorengehen.

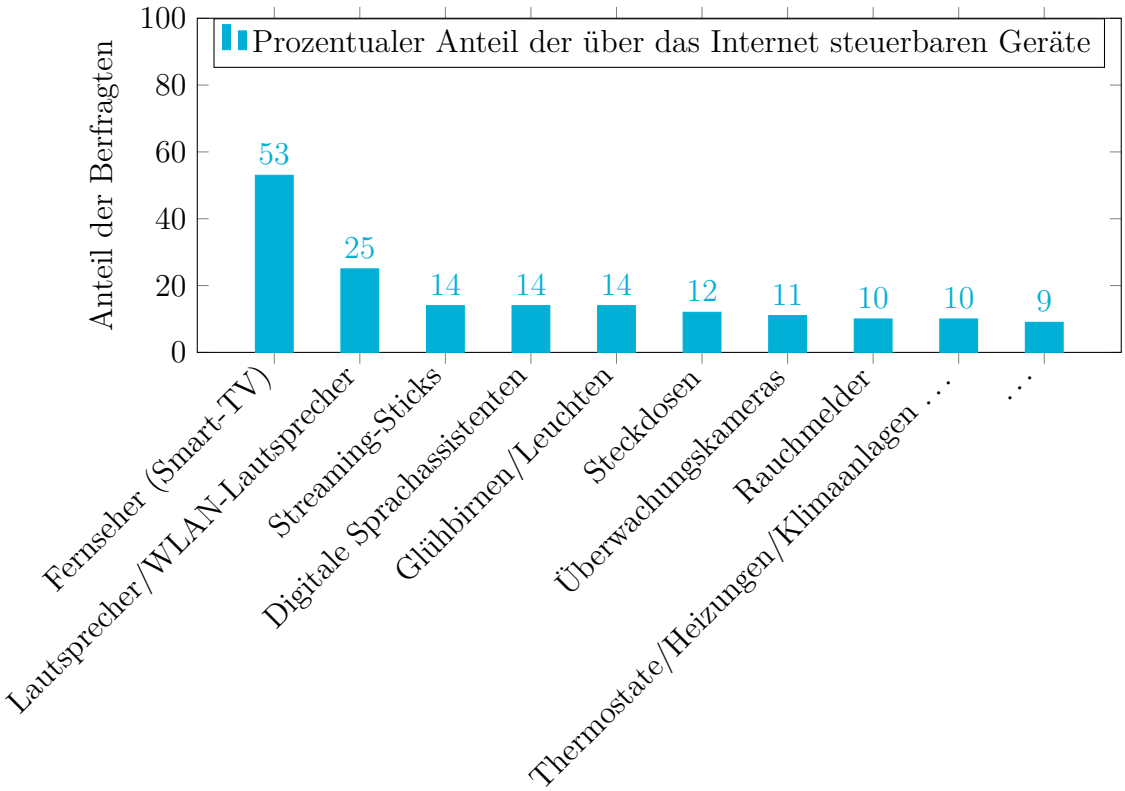

Abbildung 1.1: Vernetzte Geräte im Haushalt in Deutschland 2017, gekürzte Fassung (vgl. [2])

1.2 Ziele der Arbeit

Durch die Ausbreitung der Anwendung des IdD in intelligenten Technologien, wird das Angebot der genutzten Protokolle immer unübersichtlicher. Gravierender hierbei sind jedoch die schwerwiegenden Sicherheitsmängel dieser Protokolle, da die schnelle Markteinführung als ein Schlüsselfaktor zu sehen ist, deren Umsetzung mit einem weniger gründlichen Sicherheitsdesign und -test verbunden ist. Dies gilt insbesondere für den Bereich des Smart Home, wo der verbraucherorientierte Markt schnelle und kostengünstige Lösungen verlangt. Wie bereits in Kapitel 1.1 erwähnt, gibt es verschiedene Angriffsflächen. Um möglichen Angriffen entgegenzuwirken, ist zunächst eine gründliche Einarbeitung in das jeweilige Funkprotokoll und eine Sicher-

heitsanalyse der bestehenden Technologien vonnöten, um mögliche Schwachstellen zu ermitteln. Eine Analyse der Sicherheit des IdD wäre jedoch riesig und würde den Rahmen der Master-Thesis sprengen. Aus diesem Grund konzentriert sich die Master-Thesis im Speziellen auf die Schwachstellenanalyse einiger der derzeit am Markt anzutreffenden drahtlosen IdD-Protokolle für Smart Home.

1.3 Inhaltlicher Aufbau der Arbeit

Die Master-Thesis gliedert sich in acht Kapitel. Das erste Kapitel dient der Einleitung und beinhaltet neben der Motivation für das Thema und die Zielsetzung auch den inhaltlichen Aufbau der Arbeit. Das zweite Kapitel beschäftigt sich im Generellen mit der Schaffung von Grundlagen und einer gemeinsamen Definition der Sachverhalte. Den Beginn bildet hierbei die Einführung in das IdD. Dies soll dem Leser ein Grundverständnis des IdD vermitteln und das Gebiet des Smart Home innerhalb des IdD einordnen. Weiterhin wird im Bereich des IdD die grundlegende Architektur erörtert und Sicherheitsherausforderungen und -schwachstellen dieser dargelegt. Neben einer Einführung in das Open Systems Interconnection (OSI)-Schichten- und Transmission Control Protocol (TCP)/Internet Protocol (IP)-Modell werden auch die unterschiedlichen Netzwerktopologien betrachtet. Den Schluss des Grundlagen-Kapitels bildet eine Definition des Begriffs Penetrationstest und dessen Einordnung in den Bereich der Schwachstellenanalyse. Zum Schluss wird noch die Vorgehensweise der Bedrohungsmodellierung aufgegriffen und erläutert. Das vierte Kapitel strukturiert die restliche Arbeit, indem es die Vorgehensweise der Thesis beschreibt und festlegt. Ab Kapitel 5 widmet sich diese Ausarbeitung dann komplett der Schwachstellenanalyse der Funkprotokolle. Den Beginn macht hier ZigBee, gefolgt von Z-Wave und Bluetooth. Jedes Protokoll wird hierbei in einem eigenen Kapitel dargestellt. Den Schluss dieser Master-Thesis bildet eine abschließende Zusammenfassung in Form einer Schlussbemerkung. Es sei jedoch darauf hingewiesen, dass sich im Anhang dieser Arbeit weitergehende Informationen befinden, die in die Ausarbeitung mit einbezogen wurden.

2 Grundlagen

Eine Abhandlung mit dem Thema Schwachstellenanalyse – in unserem konkreten Fall von Funkprotokollen am Beispiel von Smart Home Anwendungen – befindet sich im Themenfeld der IT-Sicherheit. Als Ausgangsbasis für diese Arbeit ist ein theoretisches Hintergrundwissen vonnöten. Dieses dient einerseits als Grundlage für die im Hauptteil dieser Arbeit, der Schwachstellenanalyse (vgl. Kapitel 4), angeführten Darstellungen, zum anderen besteht das Problem, dass es – wie auch in anderen wissenschaftlichen Bereichen – unterschiedliche Ansätze gibt, einen Fachbegriff zu definieren. Dieser Teil soll daher in erster Linie zu einem gemeinsamen Verständnis beitragen.

2.1 Das Internet der Dinge

Das Internet der Dinge (IdD) ist ein interdisziplinäres Paradigma, in dem viele der uns umgebenden Objekte vernetzt und mit dem Internet verbunden werden, um neue Dienstleistungen anzubieten und die Produktivität zu steigern. Wie bereits in Kapitel 1.1 erwähnt, sind die Begriffe Smart Home und IdD unwiderruflich miteinander verbunden. Der nachfolgende Teil soll eine kurze Einführung und Definition in die Thematik, die zugrunde liegende Architektur, Sicherheitsherausforderungen und -schwachstellen sowie Sicherheitsmechanismen bieten.

2.1.1 Einführung und Definition

Angesichts der Publicity, die sich um das IdD entwickelt hat, ist es nicht außergewöhnlich, dass es ebenso viele Ansätze gibt, einen Terminus für diesen zu definieren. In der Literatur findet sich daher keine offizielle oder eindeutige Definition, die von der internationalen Gemeinschaft der Nutzer akzeptiert ist (vgl. [3]). Tatsächlich versuchen sich neben Akademikern und Forschern auch Praktiker, Innovatoren und weitere unterschiedliche Gruppen darin, den Begriff des IdD zu definieren. Kevin Ashton, langjähriger Mitarbeiter und Mitbegründer des Auto-ID Centers am Massachusetts Institute of Technology (MIT), gilt als Erfinder des Begriffs „Internet of Things“. In einem Interview mit Tim Cole im Februar 2018 beantwortet Ashton die

Frage, was seine Definiton des IdD im Jahr 1999 bedeutete und ob diese heute noch gültig sei, mit folgender Antwort (vgl. [4]): „It meant using the Internet to empower computers to sense the world for themselves. It still does.“ Eine weitere Definition liefern Haller, Karnouskos und Schroth in einem Konferenzbeitrag zum First Future Internet Symposium in Wien im September 2008 (vgl. [5, 3]): „A world where physical objects are seamlessly integrated into the information network, and where the physical objects can become active participants in business processes. Services are available to interact with these „smart objects“ over the Internet, query their state and any information associated with them, taking into account security and privacy issues.“ Vergleicht man die Definitionen von damals mit den unterschiedlichen Definitionen bis heute, so haben sie alle Eines gemeinsam: In der ersten Version des Internets ging es um Daten, die von Menschen geschaffen wurden, während es in der folgenden Version um Daten geht, die von Dingen geschaffen werden. Stellvertretend für alle in der Literatur genannten Definitionen, kann das IdD wie folgt definiert werden (vgl. Abbildung 2.1):

Definition: Internet der Dinge

An open and comprehensive network of intelligent objects that have the capacity to auto–organize, share information, data and resources, reacting and acting in face situations and changes in the environment.

Abbildung 2.1: Definition Internet der Dinge (vgl. [6, 2])

Nach der vorausgegangen Definition (vgl. Abbildung 2.1) handelt es sich beim IdD also um ein offenes und umfangreiches Netzwerk intelligenter Objekte, die in der Lage sind, sich automatisch zu organisieren, Informationen, Daten und Ressourcen auszutauschen und entsprechend zu reagieren und agieren, wenn es um Situationen oder Veränderungen in ihrer Umgebung geht. Doch woraus ist das IdD entstanden? Hier können unter anderem die jüngsten und fortwährenden Fortschritte in Technologien wie der drahtlosen Kommunikation, stromsparenden Prozessoren, eingebetteten Sensoren und Aktoren, Radio Frequency Identification (RFID), Cloud Computing und vielem mehr genannt werden. Obwohl nicht alle diese Technologien für jede einzelne IdD- bzw. Smart Home Anwendung benötigt werden, erleichtern sie doch deren Verbreitung, indem sie eine grundlegende Voraussetzung schaffen. So kann beispielsweise RFID zur kostengünstigen Identifizierung von Objekten eingesetzt werden. Cloud Computing ermöglicht die Berechnung und Auslagerung von Diensten an lokale oder globale Server und bietet zusätzliche Ressourcen für die Verarbeitung

großer Datenmengen oder komplexerer Vorgänge. Die wesentliche Eigenschaft des IdD ist zweifellos aber der hohe Einfluss auf viele Aspekte des täglichen Lebens und des Verhaltens potentieller Nutzer. Aus der Sicht eines Privatanwenders werden die offensichtlichsten Auswirkungen der Einführung von Smart Home bzw. des IdD sowohl im Arbeits- als auch im Privatbereich sichtbar. Gebäudeautomation, Assisted Living, E-Health, Enhanced Learning sind in diesem Zusammenhang nur einige Beispiele für mögliche Anwendungsszenarien, in denen das neue Paradigma in naher Zukunft eine zentrale Rolle spielen wird (vgl. [7]). Aus Sicht der Geschäftsanwender respektive der Industrie ergeben sich vor allem in Bereichen wie der Automatisierung und industriellen Fertigung aber auch in der Logistik, dem Geschäfts-/Prozessmanagement, der Stromversorgung sowie dem intelligenten Transport von Personen und Gütern, Konsequenzen (vgl. [8]).

Ausgehend von den obigen Überlegungen ist es somit nicht verwunderlich, dass das IdD vom National Intelligence Council (NIC) der Vereinigten Staaten von Amerika in die Liste der sechs "Disruptive Civil Technologies" mit potenziellen Auswirkungen auf die nationale Macht aufgenommen wurde. Die dort im Jahr 2008 gelisteten sechs zivilen Technologien bieten das Potenzial, die Macht der USA in den nächsten fünfzehn Jahren, beginnend ab dem Jahr 2010, zu stärken oder zu schwächen (vgl. [9]). Bis 2025 könnten sich so, laut NIC, Internetknoten in Alltagsgegenständen, wie Lebensmittelverpackungen, Möbel, Papierdokumente und mehr befinden. Die heutigen Entwicklungen zeigen zukünftige Chancen und Risiken auf, die sich ergeben, wenn Menschen selbst die gängigsten Geräte und Objekte fern bedienen, lokalisieren und überwachen können. Die populäre Nachfrage, kombiniert mit technologischen Fortschritten, könnte die großflächige Verbreitung des IdD fördern, was, wie das heutige Internet, einen unschätzbaren Beitrag zur wirtschaftlichen Entwicklung und zur militärischen Leistungsfähigkeit leisten könnte. Darüber hinaus werden potenzielle Risiken identifiziert, die sich aus der weitreichenden Verbreitung solcher Technologien ergeben. In diesem Zusammenhang wird auch betont, dass das IdD, wenn Alltagsgegenstände zu einem Informationssicherheitsrisiko werden, eine weitaus größere Verbreitung derartiger Risiken ermöglicht, als dies bisher mit dem Internet der Fall war (vgl. [9, V]).

Weiterhin gibt es noch viele technologische und soziale Herausforderungen, die gelöst werden müssen, bevor die Idee des IdD allgemein Akzeptanz findet. Im Mittelpunkt steht hierbei die vollständige Interoperabilität der miteinander vernetzten Geräte. Durch autonomes Anpassen und Verhalten sollen diese ein immer höheres Maß an Intelligenz bei gleichzeitiger Wahrung von Vertrauen, Privatsphäre und Sicherheit

erhalten (vgl. [10]). Darüber hinaus bringt die IdD-Idee einige neue Probleme in Hinblick auf die verschiedenen Möglichkeiten der Vernetzung mit sich. Faktisch zeichnet sich das IdD durch geringe Ressourcen sowohl in Bezug auf die Berechnung als auch auf die Energiekapazität aus. Dementsprechend müssen die angebotenen Lösungen neben den offensichtlichen Skalierungsproblemen auch der Ressourceneffizienz besondere Aufmerksamkeit schenken (vgl. [11, 9]).

2.1.2 Grundlegende Architektur

Bei der Recherche nach der grundlegenden Architektur des IdD findet man ebenso viele unterschiedliche Referenzmodelle wie es Definitionen gibt. Verschiedene Standardisierungsgruppen wie beispielsweise oneM2M, International Telecommunication Union (ITU) - Telecommunication Standardization Sector (ITU-T), Institute of Electrical and Electronics Engineers (IEEE) Standards Association (IEEE-SA), aber auch Foren und Konferenzen wie das IoT World Forum (IoTWF) oder die International Conference on Advanced Computer Theory and Engineering (ICACTE) haben Referenzmodelle für IdD-Architekturen entwickelt (vgl. [12, 22], [13, 6 ff.], [14, 10 f.], [15, 3 ff.], [16, V5-484 f.]).

Das IoTWF, eine von der Wirtschaft geförderte jährliche Veranstaltung mit Vertretern aus Wirtschaft, Politik und Wissenschaft, hat sich zum Ziel gesetzt, die Marktakzeptanz des IdD zu fördern (vgl. [17]). Deren Architekturausschuss, bestehend aus Branchenführern wie Cisco, IBM, Intel und SAP, veröffentlichte im November 2014 ein IdD-Referenzmodell (vgl. [18]). Ziel dieses Modells ist es, klare Definitionen und Beschreibungen zu liefern, die sich präzise auf Elemente und Funktionen von IdD-Systemen und -Anwendungen anwenden lassen. Dabei soll es in Form eines gemeinsamen Frameworks der Branche helfen, IdD-Implementierungen zu beschleunigen und die Zusammenarbeit und Entwicklung replizierbarer Bereitstellungsmodelle fördern. Abbildung 2.2 illustriert das vom Architekturausschuss der IoTWF entwickelte IdD-Referenzmodell, das in sieben Ebenen unterteilt ist. Jede Schicht liefert hierbei zusätzliche Informationen für die Definition einer gemeinsamen Terminologie. Durch die Standardisierung dieser kann ein global akzeptiertes Referenzmodell geschaffen werden, in dem beschreiben ist, wie Aufgaben auf den einzelnen Ebenen bearbeitet werden sollen, um die Simplizität sicherzustellen, eine hohe Skalierbarkeit zu ermöglichen und die Supportfähigkeit zu gewährleisten (vgl. [15]).

Ein weiteres Referenzmodell, welches durch die ITU-T in ihrer Recommendation Y.2060 definiert wurde, ist in Abbildung 2.3 dargestellt. Im Gegensatz zu den meisten anderen IdD-Referenzmodellen und architekturbasierten Modellen in der Litera-

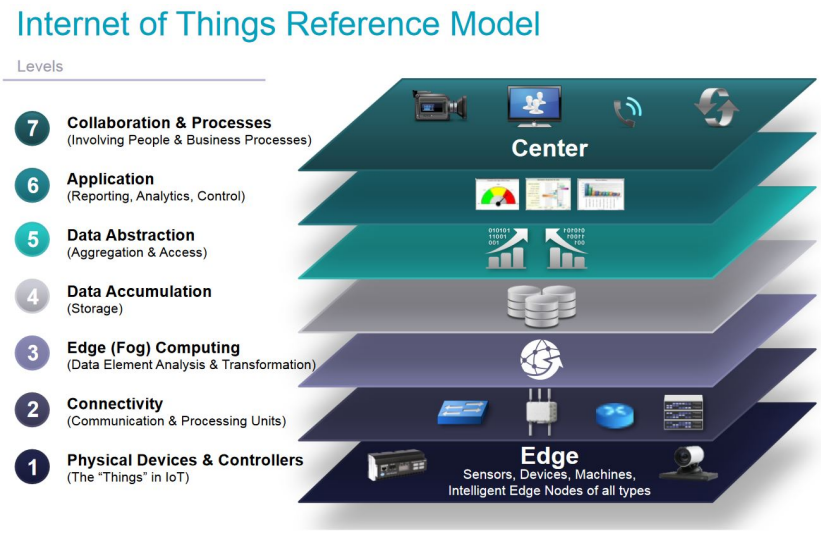

Abbildung 2.2: IdD-Referenzmodell nach dem IoTWF Architekturausschuss (vgl. [15, 3])

tur geht das ITU-T-Modell ausführlich auf die tatsächlichen physikalischen Komponenten des IdD-Ökosystems ein. Diese Vorgehensweise ist sinnvoll, da sie die Elemente im IdD-Ökosystem sichtbar macht, die miteinander vernetzt, integriert, verwaltet und den Anwendungen zur Verfügung gestellt werden müssen. Diese detaillierte Spezifikation des Ökosystems bestimmt die Anforderungen an die IdD-Fähigkeit (vgl. [13]).

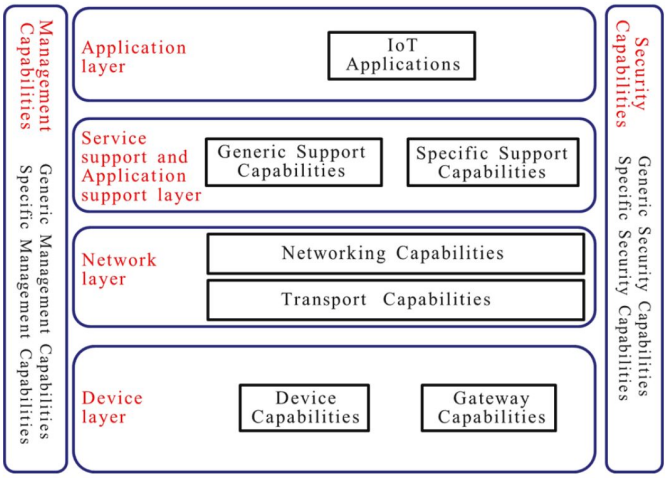

Abbildung 2.3: IdD-Referenzmodell nach der ITU-T Recommendation Y.2060 (vgl. [13, 7])

Vergleicht man das Referenzmodell des IoTWF mit dem der ITU-T Recommendation Y.2060, so zeigt sich, dass es sich eher mit der Frage der Entwicklung von Anwendungen, Middleware und Supportfunktionen für ein unternehmensweites IdD beschäftigt und damit eine sinnvolle Ergänzung zum ITU-T-Referenzmodell darstellt. Darüber hinaus konzentriert sich das Modell der ITU-T eher auf die Geräte- und Gateway-Ebene unter Einbeziehung einer eher grob gehalten Darstellung der oberen Schichten, zu denen die Anwendungsschicht gezählt werden kann. Generell kann festgestellt werden, dass es bei der ITU-T Recommendation Y.206x-Serie in erster Linie um die Definition eines Rahmens zur Unterstützung der Entwicklung von Standards im Hinblick auf die Interoperabilität von IdD-Geräten geht (vgl. [19]).

Die zuvor ausgeführten Erläuterungen zu den Referenzmodellen der IoTWF und der ITU-T sollten stellvertretend einen kurzen, oberflächlichen Überblick bieten und die Vielfältigkeit der unterschiedlichen Architekturmodelle darstellen. Die detaillierte Erörterung der einzelnen Ebenen, Funktionen und Absichten sowie ein detaillierter Vergleich beider Modelle würden den Rahmen dieser Ausarbeitung überschreiten und stellt auch nicht deren Kern dar. Nachfolgend soll daher nur die „High-Level-Architektur“ des IdD im Detail besprochen werden, die allgemein akzeptiert ist. Diese wurde unter anderem auf der dritten ICACTE im Jahr 2010 vorgestellt.

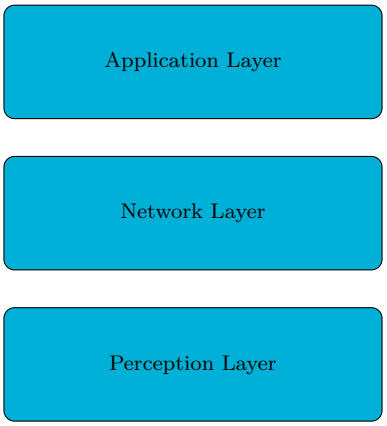

Abbildung 2.4: IdD-„High-Level-Architektur“ (abgeleitet anhand von [16, V5-484])

In Abbildung 2.4 ist die „High-Level-Architektur“ dargestellt, welche drei Ebenen umfasst. Um den spezifischen Anforderungen des IdD gerecht zu werden, bietet diese Architektur einen Rahmen, durch den verschiedene Ansätze implementiert werden können. Nachfolgend werden die drei Ebenen, von unten nach oben, kurz dargestellt und erläutert (vgl. [16, V5-484 f.]).

2.1.2.1 Perception Layer

Perception heißt ins Deutsche übersetzt „Wahrnehmung“. Dieser Begriff beschreibt im Wesentlichen die Hauptaufgabe dieser Schicht, nämlich die Wahrnehmung der physikalischen Eigenschaften der Dinge um uns herum, die Teil des IdD sind. Der Perception Layer wird auch als Device Layer bezeichnet, der aus physikalischen Objekten und Sensoren besteht (vgl. [20, 341]). Je nach Sensortyp kann er Informationen über Standort, Temperatur, Ausrichtung, Bewegung, Vibration, Beschleunigung, Feuchtigkeit, chemische Veränderungen in der Luft usw. liefern. Der Wahrnehmungsprozess selbst basiert auf verschiedenen Sensortechnologien wie Radio Frequency Identification (RFID), Global Positioning System (GPS) aber auch anderen Sensoren. Des Weiteren ist sie für die Umwandlung der Informationen in digitale Signale zuständig, die für die Netzwerkübertragung besser geeignet sind.

2.1.2.2 Network Layer

Der Network Layer, also die Netzwerkschicht, bildet das Gehirn des IdD und ist für die Übertragung und Verarbeitung von Informationen aus dem Perception Layer verantwortlich. Darüber hinaus ist sie für die Datenübertragung zum Application Layer durch verschiedene Netzwerktechnologien der drahtlosen und kabelgebundenen Kommunikation zuständig. Die wichtigsten Übertragungsmedien sind beispielsweise Mobilfunk, Wi-Fi, Bluetooth, ZigBee aber auch die Infrarot-Technologie (vgl. [20, 341]). Riesige Datenmengen werden über das Netzwerk übertragen. Daher ist es entscheidend, eine solide Middleware bereitzustellen, die diese enorme Datenmenge speichern und verarbeiten kann. Um dieses Ziel zu erreichen, ist Cloud Computing die primäre Technologie, welche in dieser Schicht zur Anwendung kommt. Diese Technologie bietet eine zuverlässige und dynamische Schnittstelle, über die Daten gespeichert und verarbeitet werden können. Im Rahmen dieser Ausarbeitung spielt die dargestellte Schicht eine zentrale Rolle, da hier die Funkkommunikation verwendet wird und somit dem Themenbereich der Funkprotokolle zugeordnet werden kann.

2.1.2.3 Application Layer

Der Application Layer, auch Anwendungsschicht genannt, ist für die Bereitstellung anwendungsspezifischer Dienste für den Benutzer verantwortlich. Sie definiert verschiedene Anwendungen, in denen das IdD eingesetzt werden kann. Hierunter fallen beispielsweise Anwendungen wie Smart Home, Smart Production, Smart Energy,

Smart Transportation und Smart Health (vgl. [20, 341], [21, 3]). Hierbei bildet die Schicht das Frontend der gesamten IdD-Architektur, durch welche die IdD-Potenziale genutzt werden können.

2.1.3 Sicherheitsherausforderungen und -schwachstellen

Im Hinblick auf die in Kapitel 2.1.2 beschriebene „High-Level-Architektur“ gibt es auch Sicherheitsherausforderungen und -schwachstellen in den verschiedenen Ebenen. Diese Sicherheitsherausforderungen und -schwachstellen werden in diesem Kapitel kurz erläutert und bilden die Grundlage für das Kapitel über die Schwachstellenanalyse (vgl. Kapitel 4). Die Namen der Angriffe sind nicht ins Deutsche übersetzt worden, da zum einen oft keine klare Übersetzung vorliegt und zum anderen die englischen Begriffe in der Fachliteratur allgemein verwendet werden.

2.1.3.1 Perception Layer

Sensoren und -Gateways, Aktoren und RFID sind nur einige Beispiele für Komponenten des Perception Layers. Diese sind zumeist kostengünstige und in der Regel ressourcenbeschränkte Geräte in Bezug auf CPU und Speicher. Weitere Merkmale sind ein niedriger Stromverbrauch, minimale Wartung und die Fähigkeit, in rauen Umgebungen zu arbeiten. Im Perception Layer bilden Angriffe auf die physische Sicherheit der Sensorinfrastruktur die häufigste Angriffsmethode, da sich die verschiedenen Arten von Sensoren in der Regel über einen längeren Zeitraum an einem Ort befinden. Darüber hinaus stellen Azis und Haq in einem Artikel im International Journal of Computers Applications im März 2018, aber auch Andrea, Chrysostomou und Hadjichristofi in einem Konferenzbeitrag zum IEEE Symposium on Computers and Communication (ISCC) im Juli 2015, allgemein mögliche Angriffsszenarien auf den Perception Layer dar. Im Folgenden werden einige dieser möglichen Angriffe kurz erläutert (vgl. [21, 4 ff.], [22, 32]):

- *Node Tampering*
  Beim „Node Tampering“ beschädigt der Angreifer einen Sensorknoten, indem er die Hardware des Knotens ganz oder teilweise physisch ersetzt. Es ist auch möglich, dass der Angreifer den Knoten elektronisch manipuliert, so dass er Zugang zu ihm erhält und sensible Informationen wie gemeinsame kryptografische Schlüssel, falls vorhanden, lesen oder ändern kann.

- *Malicious Node Injection*
  Bei diesem Angriff fügt der Angreifer einen falschen oder bösartigen Knoten

zwischen die eigentlichen Knoten des IdD-Netzwerks ein. Dies ermöglicht dem Angreifer den Zugriff und die Kontrolle über den Datenstrom des Netzwerks und gibt ihm die Möglichkeit, die Übertragung der Daten zwischen den Knoten zu lesen, zu kontrollieren und gegebenenfalls zu stören. Diese Art von Angriff ist auch bekannt unter dem Begriff „Man-in-the-Middle Attack“.

- *Malicious Code Injection*
  Unter „Malicious Code Injection“ versteht man die Kompromittierung eines Knoten, indem diesem physisch ein bösartiger Code injiziert wird, der ihm Zugang zum IdD-System verschaffen kann. Ziel ist es, Zugang zum Netzwerk zu erhalten und die Nichtverfügbarkeit von Netzwerkdiensten zu erreichen.

- *Node Jamming in Wireless Sensor Networks (WSNs)*
  Bei diesem Angriff stört der Angreifer die Funkfrequenzen, auf denen die drahtlosen Sensorknoten arbeiten. Dies unterbricht die Signalübertragung und verhindert so, dass die Konten miteinander kommunizieren können. Gelingt es ihm, einen wichtigen Hauptknoten zu stören, so kann er den gesamten bereitgestellten IdD-Dienst blockieren (vgl. [23]).

- *Sleep Deprivation Attack*
  In einem IdD-Netzwerk werden weit entfernte Sensoren in der Regel mit austauschbaren Batterien betrieben. Um Energie zu sparen, sind diese Knoten so programmiert, dass sie bei Nichtgebrauch in den „Schlafmodus“ wechseln. Bei dieser Art von Angriff leitet der Angreifer falsche Anfragen an den Knoten weiter, um zu verhindern, dass er in den Ruhemodus wechselt und den Dienst aufgrund von Stromausfällen beendet.

- *Replay Attack*
  Der Replay Angriff, auch bekannt als Playback Angriff, ist ein Angriff, bei dem ein Eindringling die Kommunikation zwischen Sender und Empfänger belauscht. Er nutzt die empfangenen Informationen um diese erneut an das Opfer zu senden. Da es sich um unveränderte, verschlüsselte Daten handelt, sind die Informationen entsprechend authentifiziert und das Opfer behandelt diese somit als eine korrekte Anfrage. Die Anfrage, welche ursprünglich zuvor schon ausgeführt wurde, wird erneut ausgeführt (vgl. [24]).

2.1.3.2 Network Layer

Der Network Layer, die Kommunikationsschicht des IdD, nutzt drahtlose und vernetzte Infrastrukturen zur Kommunikation (vgl. Kapitel 2.1.2.2). Diese sind sowohl

für passive als auch für aktive Angriffe anfällig. Passive Angriffe werden verwendet um Informationen zu erhalten, indem der Angreifer den ein- und ausgehenden Datenverkehr verfolgt und Pakete mit geeigneten Tools für den Zugriff auf sensible Daten entschlüsselt. Die meisten der aktiven Angriffe sind beispielsweise Denial of Service (DoS), IP-Spoofing oder Man-In-the-Middle Angriffe. Die Authentifizierung und Zuverlässigkeit der auf der Netzwerkschicht transportierten Daten ist daher die primäre Sicherheitsanforderung dieser Schicht. Im folgenden Kapitel werden einige der möglichen Angriffe kurz beschrieben (vgl. [21, 4 ff.], [22, 32]):

- *DoS Attack*
  Während eines DoS-Angriffs überflutet der Angreifer das IdD-Netzwerk mit mehr Daten, als es verarbeiten kann. Das Netzwerk bricht zusammen und der Dienst wird entsprechend gestört.

- *Man-in-the-Middle Attack*
  Bei dieser Art von Angriff gelingt es dem Angreifer, sich zwischen zwei Sensorknoten zu klinken und so auf eingeschränkte Daten zuzugreifen. Durch das Überwachen, Abfangen und Steuern der Kommunikation zwischen den Knoten verletzt der Angreifer die Privatsphäre der beiden Knoten. Im Gegensatz zur „Malicious Node Injection“ muss der Angreifer nicht unbedingt physisch vor Ort sein, um diesen Angriff erfolgreich durchzuführen, sondern er verlässt sich ausschließlich auf die Netzwerkkommunikationsprotokolle des IdD-Systems.

- *Sinkhole Attack*
  Bei einem „Sinkhole“-Angriff ist das Ziel des Angreifers, den gesamten Datenverkehr von benachbarten Knoten auf einen einzelnen, ausgewählten Knoten zu ziehen. Dieser ausgewählte Knoten kann beispielsweise er selbst sein, um so weitere Angriffe vorzunehmen. Erreicht wird dies, indem der Knoten für die Umgebung attraktiv gemacht wird. Dies kann zum Beispiel durch „Quality of Service“ erreicht werden. Kann der Angreifer einen störungsfreien Kanal mit geringen Latenzzeiten zur Verfügung stellen, so wird er von den Protokollen entsprechend verwendet. Solche Protokolle propagieren derartige Kanäle oft an benachbarte Knoten, wodurch nun die Pakete im „Sinkhole“ landen. Dies führt letztendlich zum Verlust der Pakete und einem DoS (vgl. [25]).

- *Sybil Attack*
  Bei einem „Sybil Attack“ beansprucht ein einzelner Knoten, der sogenannte „Sybil“ Knoten, unrechtmäßig die Identitäten einer großen Anzahl von Knoten

und imitiert diese. Diese Art des Angriffs führt dazu, dass falsche Informationen von den benachbarten Knoten akzeptiert werden. Im schlimmsten Fall kann ein Angreifer eine beliebige Anzahl zusätzlicher Knotenidentitäten mit nur einem physikalischen Gerät erzeugen (vgl. [26]).

- *Traffic Analysis Attacks*
  Mittels Sniffing kann der Angreifer vertrauliche Information ausspionieren. Grundsätzlich versucht ein Angreifer bei fast allen Angriffen zunächst, einige Netzwerkinformation über das Angriffsziel zu erhalten, bevor er seinen eigentlichen Angriff ausführt. Dies geschieht beispielsweise unter anderem mit Hilfe von Port-Scanning- oder Packet-Sniffing-Anwendungen.

2.1.3.3 Application Layer

Der Application Layer definiert alle Anwendungen, welche die IdD-Technologie nutzen oder in denen das IdD eingesetzt wird. Wie bereits erwähnt, gehört zu diesen Anwendungen auch das Smart Home. Diese Anwendungen bieten spezifische Dienste, die je nach Art der Anwendung und abhängig von den verwendeten Sensoren erfassten Informationen variieren können. Es gibt viele Probleme in der Anwendungsschicht, bei denen die Sicherheit das Hauptthema ist. Softwareangriffe sind die Hauptursache für Sicherheitsschwachstellen in jedem computergestützten System. Insbesondere wenn das IdD verwendet wird, um ein intelligentes Zuhause zu schaffen, führt es zu vielen Bedrohungen und Schwachstellen sowohl von innen als auch von außen. Um eine hohe Sicherheit in einem IdD-basierten Smart Home zu gewährleisten, ist eines der Hauptprobleme, dass die in Smart Homes verwendeten Geräte eine schwache Rechenleistung und einen geringen Speicherbedarf aufweisen. Einige allgemeine Bedrohungen auf dieser Ebene werden im Folgenden aufgeführt und erläutert (vgl. [21, 4 ff.], [22, 32]):

- *Malicious Code Attack*
  Bei dieser Art von Angriff injiziert der Angreifer bösartige Codes wie Spyware, Würmer, Viren und Trojaner in das IdD-System und -Netzwerk, um beispielsweise Daten zu ändern oder den Endbenutzern legitime Dienste zu verweigern.

- *Phishing Attack*
  Spezielle Software, die unwissentlich vom Benutzer aktiviert oder installiert wurde, ermöglicht es dem Angreifer, Anmeldeinformationen und andere wichtige Authentifizierungsinformationen zu sammeln. Diese können dann für den autorisierten Zugriff auf das IdD-System und -Netzwerk verwendet werden.

- *DoS / Distributed Denial of Service (DDoS)*
  Ein Angreifer kann DoS- oder verteilte DoS-Angriffe (DDoS) auf das betroffene IdD-Netzwerk über die Anwendungsschicht ausführen, die alle Benutzer im Netzwerk betrifft. Diese Art von Angriff kann auch berechtigte Benutzer daran hindern, auf die Anwendungsschicht zuzugreifen.

2.2 Das ISO/OSI- und TCP/IP-Referenzmodell

Als wichtige Grundlage für die Analyse von Funkprotokollen empfiehlt es sich, mit dem OSI-Referenzmodell der International Organization for Standardization (ISO) beziehungsweise der ITU-T zu beginnen. Ziel hierbei ist es, ein besseres Verständnis für die Implementierungen der Funkprotokolle, welche ab Kapitel 4 näher betrachtet werden, zu vermitteln. Das OSI-Schichtenmodell (vgl. Abbildung 2.5) bietet eine Standardarchitektur zur Definition der Netzwerkkommunikation.

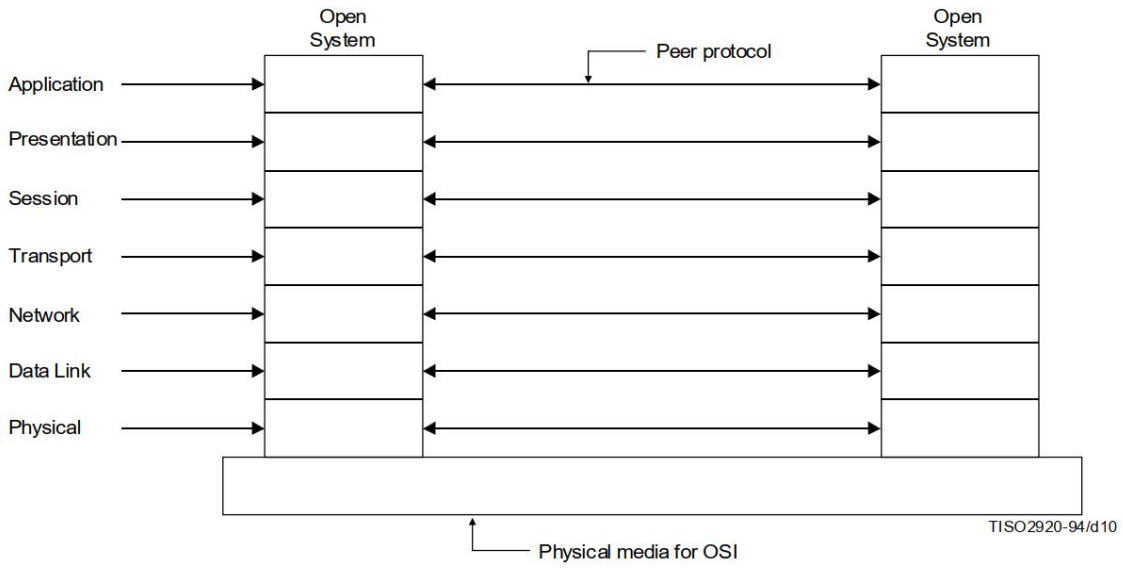

Abbildung 2.5: OSI-Referenzmodell nach der ITU-T Recommendation X.200 (vgl. [27, 28])

Nachfolgend werden die in Abbildung 2.5 dargestellten Schichten kurz beschrieben (vgl. [27, 49 ff.], [28]):

- *Schicht 1 - Physical Layer (Bitübertragungsschicht)*
  Bietet die mechanischen, elektrischen, funktionellen und verfahrenstechnischen Mittel, um die physikalische Verbindung für die Bitübertragung zu realisieren.

Die Daten werden Bit für Bit, als Bitstrom, übertragen. Fehlerbehandlung sowie die Unterscheidung zwischen Nutzdaten und Steuerinformationen findet hier nicht statt. Der Physical Layer weißt entsprechend auch keinen eigenen Header auf.

- *Schicht 2 - Data Link Layer (Sicherungsschicht)*
  Da der Physical Layer die Daten struktur- und inhaltsneutral überträgt, ist es Aufgabe des Data Link Layers, hier Strukturen zu schaffen. Die zu übertragenden Bits werden hier als logischer Bit-Verbund definiert, dem Datenrahmen beziehungsweise Data Frame. Dieser besteht aus mehreren Feldern, die Kontrollinformationen bereitstellen aber auch einem Feld für die Nutzdaten. Mittels der Frames steuert die Schicht die Übertragungsfehler, die zwischen zwei benachbarten Knoten auftreten können.

- *Schicht 3 - Network Layer (Vermittlungsschicht)*
  Der Network Layer hat die Aufgabe Pakete vom Data Link Layer zu empfangen beziehungsweise an diesen zu übertragen. Die Hauptaufgabe des Network Layers besteht jedoch darin, eine logische, hierarchische Adressstruktur bereitzustellen und mittels Routing den Weg durch das Netzwerk zu bestimmen, so dass Pakete von einem Sender den gewünschten Empfänger erreichen können.

- *Schicht 4 - Transport Layer (Transportschicht)*
  Aufgabe des Transport Layers ist es, die Daten aus dem Session Layer in kleinere Einheiten, den Segmenten, aufzuteilen und diese an den Network Layer unter der Maßgabe, dass alle Daten den Empfänger erreichen, weiterzuleiten. Der Transport Layer bildet das Bindeglied zwischen den oberen und unteren Schichten. Dabei werden die unteren Schichten (Network Layer, Data Link Layer und Physical Layer) für die Datenübertragung von Komponente zu Komponente genutzt, um Netzwerkdienste respektive Übertragungsdienste bereitzustellen. Die oberen Schichten (Session Layer, Presentation Layer und Application Layer) verantworten hingegen die Datenübertragung innerhalb jeder Komponente, um Anwendungsdienste zu ermöglichen. Alle im Transport Layer definierten Protokolle haben eine End-to-End-Verbindung. Hierdurch können jeweils ein Dienst auf Sender- und Empfängerseite über Steuerinformationen miteinander kommunizieren, wobei es keine Rolle spielt, wie viele Netze zwischen den Komponenten liegen.

- *Schicht 5 - Session Layer (Sitzungsschicht)*
  Der Zweck des Session Layer ist es, Kommunikationsanforderungen zwischen

den Endsystemen zu organisieren und zu verwalten. Für die Verbindung und den Datenaustausch notwendige Steuerungs- und Kontrollmechanismen sind entsprechend implementiert. Weiterhin handelt es sich bei den Sitzungen auf dem Session Layer um Prozess-zu-Prozess-Verbindungen.

- *Schicht 6 - Presentation Layer (Darstellungsschicht)*
  Der Presentation Layer ermöglicht eine gemeinsame Darstellung der zwischen den Anwendungen übertragenen Daten. Dadurch werden die Anwendungen von der Problematik der „gemeinsamen“ Darstellung von Informationen befreit und erhalten somit eine syntaktische Unabhängigkeit. Weiterhin stellt sie sicher, dass der Informationsgehalt der Daten des Application Layer während der Übertragung erhalten bleibt.

- *Schicht 7 - Application Layer (Anwendungsschicht)*
  Der Application Layer ist die Schicht, auf der die Endbenutzeranwendungen implementiert sind. Hierbei stellt sie Funktionen für die Anwendungen zur Verfügung, stellt die Verbindung zu den unteren Schichten her und regelt die Dateieingabe und -ausgabe.

Einen weiteren Ansatz bildet das im Request for Comments (RFC) 1122 der Internet Engineering Task Force (IETF) beschriebene TCP/IP-Referenzmodell, das häufig für die Internetkommunikation verwendet wird, da es das OSI-Schichtenmodell in einer vereinfachten Sicht mit nur vier Schichten darstellt (vgl. [29]). Abbildung 2.6 zeigt die Schichten des TCP/IP-Referenzmodell in Gegenüberstellung mit den Schichten des OSI-Referenzmodells (vgl. [30, 11]). Nachfolgend werden die vier Schichten des TCP/IP-Referenzmodells kurz beschrieben (vgl. [30, 13 ff.]):

- *Schicht 1 - Link Layer*
  Wie in Abbildung 2.6 ersichtlich, fasst der Link Layer den Physical Layer und den Data Link Layer des OSI-Referenzmodells zusammen. Dies drückt sich auch in seinen Aufgaben aus. Zum einen hat er die Aufgabe, die Bitübertragung über ein physikalisches Medium sicherzustellen, zum anderen ist er auch für das Framing, also die Einkapselung der Protokolle des Internet Layers in Frames, verantwortlich.

- *Schicht 2 - Internet Layer*
  Der Internet Layer wird in der Literatur auch, analog zum OSI-Referenzmodell, als Network Layer bezeichnet (vgl. [31, 8]). Die Aufgaben erstrecken sich über das Empfangen und Versenden von Daten bis hin zum Routing der Pakete.

TCP/IP-Modell:	OSI-Modell:
Application Layer	Application Layer Presentation Layer Session Layer
Transport Layer	Transport Layer
Internet Layer	Network Layer
Link Layer *(auch: Network Access Layer)*	Data Link Layer Physical Layer

Abbildung 2.6: TCP/IP-Referenzmodell nach dem IETF RFC 1122 im Vergleich mit dem OSI-Referenzmodell (vgl. [30, 11])

- *Schicht 3 - Transport Layer*
  Mittels dem Transport Layer werden die durch den Internet Layer zum Ziel beförderten Daten an die zugehörige Anwendung geliefert. Jede dieser Verbindungen wird durch einen Quell- und eine Zielport spezifiziert. Hier kommt auch das Multiplexing zum Einsatz, dass multiple Verbindungen ermöglicht.
- *Schicht 4 - Application Layer*
  Die letzte Schicht des TCP/IP-Referenzmodells, der Application Layer, fasst die verbleibenden drei Schichten des OSI-Referenzmodells zusammen und wird von den einzelnen Netzwerkprogrammen und -diensten genutzt.

Im Bereich des IdD gibt es, wie bereits in den Grundlagen in Kapitel 2.1.2 angeschnitten, unzählige Modelle. Im Anschluss an die vorhergehenden Ausführungen zum OSI- und TCP/IP-Referenzmodell konzentriert sich diese Ausarbeitung auf die „High-Level-Architektur“ des IdD (vgl. Abbildung 2.4). Entsprechende Gegenüberstellungen zu den Modellen der Funkprotokolle werden in den einzelnen Kapiteln der Schwachstellenanalyse aufgeführt.

2.3 Netzwerktopologien

Ein weiterer wichtiger Punkt, den es zu berücksichtigen gilt, ist die Art und Weise, wie die Geräte innerhalb der drahtlosen Netzwerke miteinander verbunden sind. Diese Netzwerke stellen das Rückgrat des IdD dar, da Sensordaten zuverlässig und

so zeitnah wie möglich, d.h. mit minimaler Latenzzeit, zur Verfügung stehen müssen. Die Form der Kommunikation dieser Daten ist daher entscheidend. Es gibt verschiedene Topologien für den Aufbau solcher drahtloser Netzwerke. Ihr Verständnis ist der Schlüssel zum Aufbau eines erfolgreichen Systems. Nachfolgend werden die Topologien, in Verbindung mit Abbildung 2.7, kurz dargestellt (vgl. [32], [33]):

- *Point-To-Point*
  Die Point-To-Point Topologie, auch als „Line“ Topologie bezeichnet, ist die einfachste der Netzwerktopologien. Hierbei sind zwei Systeme jeweils über ein einziges Netzwerkmedium miteinander verbunden. Vorteil dieser Topologie ist das einfache und dauerhafte Herstellen der Verbindung zwischen den beiden Endpunkten. Da keine zusätzliche Hardware zur Verbindung notwendig ist, stellen nur die Endpunkte an sich potenzielle Fehlerquellen dar.

- *Token Ring*
  Bei der Token Ring Topologie hat jede Netzwerkstation jeweils eine Verbindung zu jedem Nachbarn. Die Kollisionsvermeidung erfolgt mittels einem speziellen Drei-Byte-Rahmen, dem Token. Da eine Station ohne das Token nicht senden kann, gibt es entsprechend keine Kollisionsdomäne. Jeder Knoten kann zu einem potenziellen Fehlerpunkt werden, wenn er in dem Moment ausfällt, während er den Token hält. Dieses geht dann verloren und muss, mit Verzögerungen, neu generiert werden.

- *Star*
  Die Star Topologie entsteht, wenn die Point-To-Point Topologie mit einem zentralen Hub, Switch oder Router verwendet wird, an den jeder Netzwerkknoten angeschlossen ist. Hierdurch ist jede Netzwerkstation indirekt mit jedem anderen Knoten verbunden, der mit demselben Hub verbunden ist. Als „Single Point of Failure“ ist hier der Hub zu nennen. Fällt dieser aus, ist keine weitere Kommunikation möglich.

- *Mesh*
  Wird der zentrale Hub eliminiert und jeder Knoten mit jedem der anderen Knoten über Point-To-Point Verbindungen verbunden, so erhält man eine Mesh Topologie. Unterschieden werden hierbei vollständig verbundene Mesh-Netzwerke, sogenannte „Fully Connected“ Topologien, und nur teilweise verbundene Mesh-Netzwerke. Bei einem vollständig verbundenen Mesh-Netzwerk ist jeder Knoten mit jedem anderen Knoten verbunden, wohingegen beim teilweise verbundenen Mesh-Netzwerk nur einige Knoten Verbindungen zu mehr als

einem Knoten aufweisen. Die Anzahl der Verbindungen bei vollständig verbundenen Mesh-Netzwerken wächst quadratisch mit der Anzahl der Knoten.

- *Tree*
  Die Tree Topologie kann mit einer Sammlung von Star Topologien verglichen werden, bei denen die Hubs miteinander verbunden sind. Die einzelnen Netzwerkknoten in der Tree Topologie werden als Leaves, also Blätter, bezeichnet. Wird die Verbindung zu einem Leave verloren, so ist nur dieser vom Netzwerk isoliert. Verliert einer der Hubs eine Verbindung, so kann ein ganzer Abschnitt an Leaves vom Netzwerk isoliert werden.

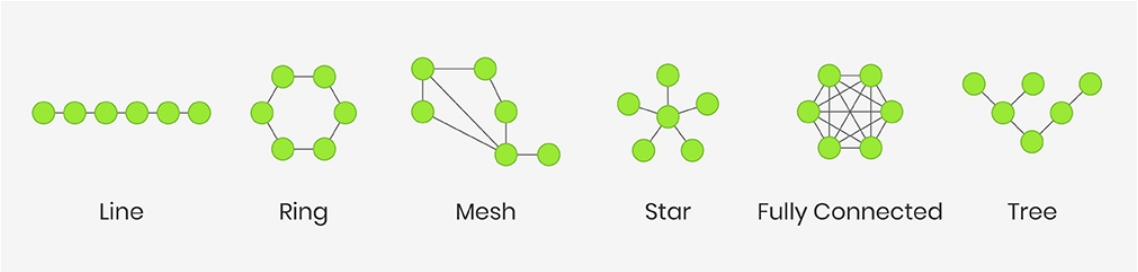

Abbildung 2.7: Netzwerktopologien im IdD-Umfeld (vgl. [33])

2.4 Schwachstellenanalyse und Penetrationstests

Das Potenzial der Informationstechnologie ist heute überall zu finden. Unternehmen erhöhen ihre Abhängigkeit von Informationstechnologien wie Cloud, IdD-Geräten, mobilen Geräten und sozialen Medien täglich und auch vor dem privaten Haushalt macht der Technologiezuwachs keinen Halt. Das Cyber-Risiko steigt hierdurch weiterhin in alarmierender Geschwindigkeit. Jeden Tag ist von Cyber-Angreifern zu hören, die in Computersysteme und Server eindringen und Alles, von Passwörtern bis hin zu Finanzinformationen, stehlen. So warnt beispielsweise auch die Telekom vor einer drastischen Zunahme der Cyper-Attacken. Demnach registriere sie pro Tag bis zu 46 Millionen Angriffe auf ihre Infrastruktur, so Dirk Backofen, Leiter Telekom Security / Senior Vice President (SVP) and Head of Telekom Security (vgl. [34]).

Doch wie machen sie das? Die Antwort ist einfach. Cyberkriminelle nutzen in der Regel Schwachstellen in Computersystemen, Netzwerken und Anwendungen aus, um das zu bekommen, was sie wollen. Diese Schwachstellen lassen sich auf mehrere Gründe zurückführen. Fehler im Design von Hard- und Software können geschäftskritische

Daten einem Risiko aussetzen. Dies gilt ebenso für mangelhaft konfigurierte IT-Systeme. Die so entstehenden Schwachstellen können durch Angreifer genutzt werden um in die Systeme einzudringen, an Informationen zu gelangen und diese in Sekundenschnelle zu stehlen. Ungesicherte Netzwerke machen es den Cyberkriminellen ebenfalls einfach, Systeme anzugreifen. Je komplexer die Architektur eines IT-Systems ist, desto größer ist auch die Chance, dass dieses System anfällig für Schwachstellen ist und somit angreifbar wird. Häufig können auch menschliche Fehler, wie unsachgemäße Entsorgung sensibler Dokumente, Programmierfehler, Bedrohungen durch Insider, gemeinsame Nutzung von Passwörtern usw. zu Sicherheitsverletzungen führen.

Die gute Nachricht ist jedoch, dass es einen Weg gibt, wie Sicherheitsschwachstellen in Computersystemen, Netzwerken und Anwendungen aufgedeckt werden können, bevor dies Cyberkriminelle schaffen. Dies kann durch einen iterativen Prozess erreicht werden, der als Penetrationstest oder einfach als Pen-Test bekannt ist.

2.4.1 Was ist ein Penetrationstest?

Das Wort Penetration kommt vom spätlateinischen Substantiv *penetrātio* und bedeutet übersetzt soviel wie „das Eindringen“ (vgl. [35]). Bringt man diesen Begriff in Verbindung mit der Informationstechnologie, so beschreibt ein Penetrationstest das Eindringen beziehungsweise den Versuch des Eindringens in ein IT-System. Technisch betrachtet handelt es sich um eine systematische Untersuchung des IT-Systems von innen oder außen, um mögliche Schwachstellen, welche von einem Angreifer genutzt werden können, ausfindig zu machen. Hierbei kann das IT-System eine beliebige Kombination aus Applikation, Host oder Netzwerk sein. Bei einem Penetrationstest handelt es sich also um eine Maßnahme die Resistenz aller IT-Infrastrukturkomponenten gegen Angriffe vor Eintreten eines Sicherheitsvorfalles zu bestimmen. Er repräsentiert damit also die Durchführung von Angriffen gegen ein IT-System.

Diese Durchführung soll reale Angriffe nachahmen und die Verhaltensweise von Angreifern aus deren Sichtweise simulieren. „Hierzu werden die Erfolgsaussichten eines vorsätzlichen Angriffs auf einen Informationsverbund oder ein einzelnes IT-System eingeschätzt und daraus notwendige ergänzende Sicherheitsmaßnahmen abgeleitet beziehungsweise die Wirksamkeit von bereits umgesetzten Sicherheitsmaßnahmen überprüft“ (vgl. [36, 4]). Eine klare Abgrenzung zu einem realen Angriff kann gezogen werden nach der Maßgabe, dass die Angriffe von dem Besitzer des zu untersuchenden Systems oder Netzwerkes initiiert oder in Auftrag gegeben werden. Da ein

Angriff Schäden anrichten und auch rechtliche Folgen mit sich bringen kann, müssen sämtliche Angriffe vom Initiator hinsichtlich der zu beachtenden gesetzlichen Rahmenbedingungen und Bedenken kontrolliert werden. Hier sind vor allem die sogenannten Hackerparagrafen zu nennen, welche seit dem 11.08.2007 in Kraft sind. Im §202a Strafgesetzbuch (StGB) (vgl. Abbildung 2.8) wird etwa das Ausspähen von Daten geregelt. Hierunter könnte nach entsprechender Auffassung beispielsweise das Knacken eines verschlüsselten WLANs fallen.

Strafgesetzbuch (StGB): §202a Ausspähen von Daten

1) Wer unbefugt sich oder einem anderen Zugang zu Daten, die nicht für ihn bestimmt und die gegen unberechtigten Zugang besonders gesichert sind, unter Überwindung der Zugangssicherung verschafft, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

Abbildung 2.8: StGB: §202a Ausspähen von Daten (vgl. [37])

Der Penetrationstest soll als Grundlage für Aussagen über die Sicherheit eines bestimmten Kontextes dienen. Als ein Leitbild für die Durchführung von Penetrationstests kann ein Zitat von Sun Tzu aus seinem Buch „The Art of War“ herangezogen werden, welches wie folgt lautet (vgl. [38, 30]):

„If you know the enemy and know yourself,
you need not fear the result of a hundred battles.“ (vgl. [39, 11])

Das Zitat besagt also, dass man einem möglichen Feind ohne Angst vor dem Ausgang eines Kampfes entgegentreten kann, wenn man ihn und sich selbst kennt. Wird dies in den Kontext eines Penetrationstests gestellt, so kann man mit dem Wissen über Vorgehensweisen vorzeitig verhindern, dass ein potenzieller Angreifer Schaden verursachen kann. Basierend auf den oben genannten Überlegungen kann ein Penetrationstest wie folgt definiert werden (vgl. Abbildung 2.9).

Definition: Penetrationstest

Penetrationstests sind der Versuch, Schwachstellen durch kontrollierte Angriffe auszunutzen, um festzustellen, ob unbefugter Zugriff oder andere schädliche Aktivitäten gegen ein Betrachtungsobjekt möglich sind. Die Angriffe werden in Übereinstimmung mit den gesetzlichen Rahmenbedingungen durchgeführt, nachdem sie von der für das Betrachtungsobjekt verantwortlichen Person in Auftrag gegeben oder initiiert wurden.

Abbildung 2.9: Definition Penetrationstest

2.4.2 Schwachstellenanalyse versus Penetrationstest

Der Schwerpunkt dieser Arbeit liegt in der Schwachstellenanalyse von Funkprotokollen am Beispiel von Smart Home Anwendungen. Wie in der Einleitung des Kapitels Grundlagen (vgl. Kapitel 2) bereits erwähnt, kann es bei der Verwendung von fachspezifischen Begriffen zu unterschiedlichen Auffassungen kommen. In Anlehnung an Dr. Daniel Hamburg sind „technische Sicherheitsanalysen und Penetrationstests [...] Schlüsselkomponenten einer nachhaltigen IT-Sicherheitsstrategie (vgl. [40])“. Worin liegt aber nun der tatsächliche Unterschied zwischen Schwachstellenanalyse und Penetrationstest?

Nach Luber und Schmitz stellt die Schwachstellenanalyse „einen Oberbegriff dar und kann Vulnerability- oder Security Scans sowie Penetrationstests beinhalten (vgl.[41]).“ Die Analyse von Schwachstellen stellt dabei einen Prozess der Definition, Identifizierung, Klassifizierung und Priorisierung von Schwachstellen in IT-Systemen, Anwendungen und Netzwerkinfrastrukturen dar und umfasst typischerweise auch die Verwendung von automatischen Testwerkzeugen, wie beispielsweise Netzwerksicherheitsscannern, deren Ergebnisse im Rahmen eines Abschlussberichtes dargelegt werden. Dabei wird die IT-Infrastruktur auf ihre Konformität, Effizienz und Effektivität hin überprüft, wobei die Ausnutzung und das Eindringen in die Infrastruktur oftmals nicht in Betracht gezogen wird. „Ein Penetrationstest hingegen ist kaum automatisiert und erfolgt nach ausführlicher, oft manueller Informationssammlung(vgl.[41]).“ Hierbei geht er in der Regel einen Schritt weiter, indem er mehr Wert darauf legt, Schwachstellen zu identifizieren und soviel Zugriff wie möglich auf das IT-System zu erlangen um diese dann effektiv ausnutzen zu können. Die Bewertung von Schwachstellen endet demnach unmittelbar vor der Kompromittierung des IT-Systems, während ein Penetrationstest das Ziel verfolgt, das

IT-System zu kompromittieren, um festzustellen, wie tief ein Angreifer vordringen kann und wie schwerwiegend ein Angriff sein könnte.

Auch in den durch die Common Criteria for Information Technology Security Evaluation (CC) beschriebenen Zielen zur Analyse von Schwachstellen wird der Unterschied zwischen Schwachstellenanalyse und Penetrationstest in ähnlicher Weise dargestellt (vgl. [42, 185 ff.]):

- „A vulnerability analysis is performed by the evaluator to ascertain the presence of potential vulnerabilities.“
- „The evaluator performs penetration testing, to confirm that the potential vulnerabilities cannot be exploited in the operational environment for the Target of Evaluation (TOE). Penetration testing is performed by the evaluator assuming an attack potential of Basic.“

Das nachfolgende Beispiel soll den prinzipiellen Unterschied einer Schwachstellenanalyse zu einem Penetrationstest näher veranschaulichen. Die Analyse von Schwachstellen ist wie die Betrachtung einer Tür mit der Überlegung, ob diese ver- oder entriegelt ist. Dies könnte es jemandem erlauben, sich unbefugten Zugang zu verschaffen. Ein Penetrationstest hingegen versucht tatsächlich die Tür zu öffnen und zu erkunden, wohin diese führt. Er versucht zu ergründen, ob sich weitere Möglichkeiten ergeben, nachdem er durch die Tür vorgedrungen ist. Ein Penetrationstest ist somit ein zuverlässiger Indikator für die Schwächen von Netzwerken oder IT-Systemen. Penetrationstests sind von Natur aus invasiver, während die Schwachstellenanalyse hingegen vergleichsweise weniger invasiv ist und die System- oder Netzwerkdienste potenziell nicht beeinträchtigt. Daher hat der Penetrationstest mehr Potenzial, System- oder Netzwerkdienste zu stören und Schwachstellen zu identifizieren.

Zusammengefasst lässt sich feststellen, dass die Schwachstellenanalyse ein wichtiges Instrument für die proaktive IT-Sicherheit ist. Den nächsten Schritt hierbei bilden, als empirischer Teil, die Penetrationstests.

2.5 Bedrohungsmodellierung

Betrachtet man beispielsweise die Softwareentwicklung oder die System- und Netzwerkbereiche, so ist man in der Regel mit den möglichen Angriffsflächen und -vektoren dieser Bereiche vertraut. Der Begriff Angriffsfläche oder Angriffsvektor bezeichnet hierbei die Vielzahl an Möglichkeiten, mit denen ein System potenziell bedroht und kompromittiert werden kann (vgl. [43]). Prinzipiell ist die Wahrscheinlichkeit

eines Kompromisses umso höher, je mehr Angriffsvektoren ein Gerät aufweist. Dabei bilden die Angriffsvektoren in erster Linie die Grundlage für Bedrohungen, die das Risiko beinhalten, das Gerät mit dem Ziel unbeabsichtigter Handlungen negativ zu beeinflussen. Der Prozess der Bedrohungsmodellierung soll dem entgegenwirken. Dieser Auffassung ist auch Microsoft und beschreibt in einem Artikel zum Thema der Sicherheitsarchitektur im Bereich des IdD, dass die „Sicherheit [...] mit einem Bedrohungsmodell“ beginnt (vgl. [44]). Ziel der Bedrohungsmodellierung ist somit, alle verschiedenen Einstiegspunkte darzustellen, die ein Angreifer potenziell missbrauchen könnte. Dieser Schritt stellt einen der wichtigsten Schritte innerhalb der Schwachstellenanalyse respektive des Penetrationstests dar und erstreckt sich über die Phasen der Informationsbeschaffung sowie deren Bewertung und der Risikoanalyse hinweg (vgl. Abbildung 3.1).

2.5.1 Bestimmung der Angriffsvektoren

Wie zu beginn des Kapitels 2.5 bereits erwähnt, bilden Angriffsvektoren die Basis der Bedrohungsmodellierung. Doch wie lassen sich diese näher bestimmen? Das folgende Kapitel soll dieses Thema näher beleuchten und für ein gemeinsames Verständnis sorgen.

Zu Beginn der Definition der Angriffsvektoren sollten alle möglichen Informationsquellen durchsucht und Informationen gesammelt werden. Potenzielle Informationsquellen hierbei können Gerätedokumentationen und Handbücher, Online-Ressourcen und Artikel über das Gerät sowie alle verfügbaren Inhalte und vorherige Recherchen über das jeweilige Gerät sein. Dabei sind auch die verschiedenen Systemkomponenten wie CPU, Architekturtyp, Firmware-Update-Prozess und vor allem, im Falle dieser Master-Thesis, die verwendeten Kommunikationsprotokolle zu beachten. Betrachtet man IdD-Lösungen aus einem höheren Level, so lässt sich die gesamte Architektur grob in drei Kategorien unterteilen:

1. Embedded Device
2. Firm- und Software
3. Funkkommunikation

Das Ziel bei der Definition von Angriffsvektoren ist es, Funktionalitäten zu definieren und potenzielle Bedrohungen in Abhängigkeit von der jeweiligen Architekturkategorie einzuordnen. Diese Ausarbeitung bezieht sich hierbei jedoch lediglich auf die

Kategorie der Funkkommunikation. Diese bildet die Grundlage für die Kommunikation zwischen unterschiedlichen Geräten. Sofern ein Gerät einen Datenaustausch über drahtlose Übertragungsmedien ausführt, kann seine drahtlose Verbindung unter Umständen überwacht, entschlüsselt, wiederholt, verfälscht, gestohlen oder sogar angegriffen oder kontrolliert werden. Die gebräuchlichsten Funkprotokolle, die in IdD-Geräten verwendet werden, sind Mobilfunk, Wi-Fi, Bluetooth Low Energy (BLE), ZigBee, Z-Wave und viele mehr (vgl. [45]). Dementsprechend kann es notwendig sein, dass eine spezielle Hardware für die Analyse der Funkkommunikation erforderlich wird. Hierauf wird noch näher im nachfolgenden Kapitel B zur Einrichtung Penetrationstestlabors eingegangen. Neben den verschiedenen Hardwarekomponenten sind hier auch die notwendigen Softwarekomponenten aufgeführt, die für die Sicherheitsbewertung des Funkprotokolls der ZigBee Spezifikation erforderlich sind. In Abhängigkeit der jeweiligen Funkkomponente die untersucht wird, gibt es unterschiedliche Arten von Schwachstellen. Anhand einer Recherche, zu den am häufigsten auftretenden Schwachstellen in Funkprotokollen, lässt sich folgende Aufstellung erstellen (vgl. [46], [47], [48]):

- Replay Angriff
- Man-in-the-Middle Angriff
- Radio Frequency Jamming
- Denial of Service
- Fehlende Verschlüsselung
- Packet Sniffing
- Eavesdropping

Auf die vorher genannten Schwachstellen, welche nur einen Auszug möglicher Angriffe darstellen, wird in den Kapiteln ab Kapitel 4 im Zuge der Schwachstellenanalyse, im gegebenen Fall, noch näher eingegangen.

Beschäftigt man sich mit dem Thema der Bedrohungsmodellierung, so ist es, im Rahmen der Erstellung von Angriffsvektoren der unterschiedlichen Funkkommunikationen, sinnvoll sich auch mit den folgenden Fragestellungen zu befassen:

- Mit welcher Frequenz arbeitet das Gerät?
- Gibt es ähnliche Geräte, die im gleichen Frequenzbereich arbeiten wie dieses Gerät?

- Wie viele Geräte kann jede Komponente gleichzeitig bedienen?
- Welche Komponente initiiert den Authentifizierungs- und Kopplungsmechanismus?
- Wie sieht der Kopplungsmechanismus aus?
- Welche Protokolle werden von den verschiedenen Komponenten verwendet?
- Handelt es sich um freie oder proprietäre Protokolle?

Diese Punkte stellen jedoch lediglich einen Auszug aus den Überlegungen dar, die bei der Analyse der Funkkommunikationen respektive der Funkprotokolle für die Geräte im Bereich des Smart Home berücksichtigt werden sollten, um mögliche Angriffsvektoren zu bestimmen.

2.5.2 „DREAD“-Modell

Wurden Bedrohungen für die Funkkommunikation respektive der Funkprotokolle mittels der Definition von Angriffsvektoren definiert, so sollten diese entsprechend bewertet werden, damit erkennbar wird, welche Probleme gravierend sind beziehungsweise vernachlässigt werden können. Hierfür werden die nachfolgend aufgelisteten Fragestellungen genutzt, die jeweils einer Bewertungskategorie zugeordnet sind (vgl. [49], [50, 224 f.]):

- **Damage Potential:** Wie groß ist der Schaden, wenn er ausgenutzt wird?
- **Reproducibility:** Wie einfach ist es, den Angriff zu reproduzieren?
- **Exploitability:** Wie einfach ist es, anzugreifen?
- **Affected Users:** Wie viele Benutzer sind in etwa betroffen?
- **Discoverability:** Wie einfach ist es, die Schwachstelle zu finden?

Das „DREAD“-Modell stellt ein Risikobewertungssystem im Bereich von 1-3 zur Verfügung. Hierbei repräsentiert die *1 ein geringes Risiko (Low)*, *2 ein mittleres Risiko (Medium)* und *3 ein hohes Risiko (High)*. In Tabelle 2.1 ist eine typische Bewertungstabelle dargestellt, die bei der Priorisierung von Bedrohungen verwenden werden kann. Hierzu beantwortet man, für jede Bedrohung einzeln, die oben aufgelisteten Fragen unter Verwendung der Bewertungstabelle und zählt anschließend die Punkte zusammen. Das Ergebnis kann dabei im Bereich von 5-15 liegen. Hierbei wird ein Ergebnis von *5-7 als geringes Risiko (Low)*, *8-11 als mittleres Risiko (Medium)* und *12-15 als hohes Risiko (High)* eingestuft. Wenn eine Bedrohung als hoch

eingestuft wird, stellt sie ein erhebliches Risiko dar und muss so schnell wie möglich behoben werden. Mittlere Bedrohungen müssen angegangen werden, allerdings mit geringerer Dringlichkeit. Niedrige Bedrohungen können hingegen eher vernachlässigt werden.

	High (3)	**Medium (2)**	**Low (1)**
***D**amage Potential*	Angreifer erlangt Administratorrechte	Vertrauliche Informationen werden bekannt	Triviale Informationen werden bekannt
***R**eproducibility*	Stets reproduzierbar	Reproduzierbar mittels Zeitfenster und „Race Condition“	Schwer reproduzierbar, auch bei genauer Kenntnis der Schwachstelle
***E**xploitability*	Angriff ist dem Anfänger/Laien möglich	Erfahrener Angreifer könnte den Angriff durchführen	Erfordert einen extrem erfahrenen Spezialisten und ein fundiertes Wissen
***A**ffected Users*	Alle Benutzer betroffen	Einige Benutzer betroffen	sehr geringer Prozentsatz von Benutzern betroffen
***D**iscoverability*	Veröffentlichte Informationen erklären den Angriff	Schwachstelle nicht sofort ersichtlich	Schwachstelle ist verborgen und es ist unwahrscheinlich, dass Angreifer das Schadenspotenzial herausfinden

Tabelle 2.1: Bewertungstabelle des „DREAD“-Modells zur Priorisierung von Bedrohungen (vgl. [49])

3 Vorgehensweise

Im Rahmen der Ausarbeitung dieser Masterarbeit soll, wie bereits erwähnt, eine Schwachstellenanalyse von Funkprotokollen sowie eine genauere Untersuchung des Funkprotokolls der ZigBee Spezifikation in Verbindung mit aktiven Eindringversuchen durchgeführt werden. In diesem Zusammenhang ist ein einheitliches Verfahren zu evaluieren, um einerseits der Schwachstellenanalyse gerecht zu werden und andererseits aktive Eindringversuche durch Penetrationstests in diesen Rahmen einzuführen (vgl. Kapitel A.4.2).

Als Ausgangspunkt hierfür dient das fünfstufige Verfahren für Penetrationstests des Bundesamt für Sicherheit in der Informationstechnik (BSI) (vgl. Abbildung A.3 sowie Kapitel A.4.3). Das Schlüsselkonzept der Schwachstellenanalyse (vgl. Abbildung A.2 sowie Kapitel A.4.2) gliedert sich dabei in die unterschiedlichen Phasen des Penetrationstests ein. Dabei korrespondieren die drei Phasen der Schwachstellenanalyse praktisch mit den ersten drei Phasen des Penetrationstests, weswegen sich dieses Vorgehensmodell auch für die Ausarbeitung und Strukturierung dieser Arbeit eignet. In diesem Zusammenhang ist zu berücksichtigen, dass nicht alle Testschritte aus der Penetrationsteststudie des BSI übernommen wurden. Vielmehr wurden diese speziell für die Analyse von Funkprotokollen ausgelegt und bilden somit die Struktur für die Kapitel der Schwachstellenanalyse ab Kapitel 4. In Abbildung 3.1 ist entsprechend der vorhergegangen Erläuterungen die Vorgehensweise der Schwachstellenanalyse für diese Ausarbeitung in Verbindung mit dem Prozess des Penetrationstest dargestellt. Den Kern der Schwachstellenanalyse stellen die Phasen der Informationsbeschaffung und deren Evaluierung dar, die durch die in Kapitel 2.5 dargestellte Definition zur Bestimmung von Angriffsvektoren und dem „DREAD“-Modell zur Bedrohungsbewertung unterstützt werden.

Bei der Schwachstellenanalyse der Funkprotokolle ohne eine tiefer gehende Analyse wird die Phase 4 der aktiven Eindringversuche nicht durchgeführt und es wird somit direkt in die Phase 5 der Abschlussanalyse übergegangen. Des Weiteren ist festzustellen, dass die Phase 1 der Vorbereitung einen allgemeingültigen Charakter ausweist und daher in Kapitel 4, als Einleitung zu den darauf folgenden Kapiteln,

lediglich einmal aufgeführt wird.

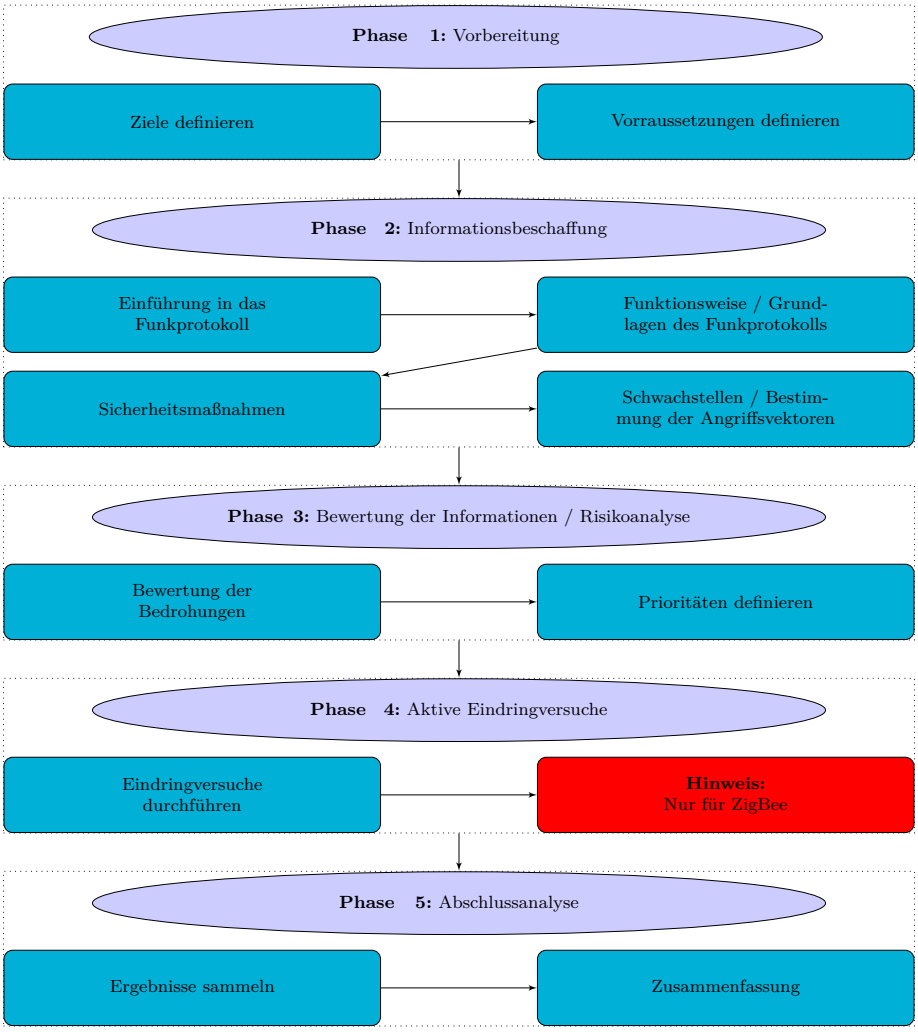

Abbildung 3.1: Schwachstellenanalyse in Verbindung mit dem Prozess des Penetrationstests (in Anlehnung an [51, 3], sowie [52, 47])

Im Rahmen der tiefer gehenden Analyse des Funkprotokolls der ZigBee Spezifikation mittels Penetrationstest ist eine spezifische Klassifizierung der Vorgehensweise erforderlich. Grundlage hierfür bilden die Ausführungen in Kapitel A.3 zur Klassifizierung von Penetrationstests. Nachfolgend soll die getroffene Auswahl für die Vorgehensweise kurz erläutert werden. In Tabelle 3.1 ist die Auswahl entsprechend zusammengefasst dargestellt. Hinsichtlich der Informationsbasis wurde sich für den Black-Box-Ansatz entschieden. Zwar lassen sich im Internet über das zu untersuchende Funkprotokoll viele Informationen, Architekturdarstellungen, Schwachstellen sowie tiefer gehende Informationen finden, allerdings stellt diese Informationsbasis nicht das Wissen eines Insider dar, was letztlich nicht für einen White-Box-An-

satz spricht. Für die Aggressivität und Intensität wurde ein aggressives Vorgehen gewählt mit dem Bestreben, alle potenziellen Schwachstellen auszunutzen. Hierbei beschränkt sich der Umfang auf ein fokussiertes Vorgehen, da im Fall dieser Master-Thesis nur die Schwachstellen von Funkprotokollen von Smart Home Anwendungen beziehungsweise IdD-Geräten untersucht werden, nicht aber das komplette System an sich. Im Gegensatz zur Darstellung in Kapitel A.3.2, in welcher darauf hingewiesen wird, dass eine aggressive und zugleich offensichtliche Vorgehensweise auf Grund der schellen Identifikation des Angriffs nicht besonders geeignet ist, wurde hier eine offensichtliche Vorgehensweise in Kombination mit einem aggressiven Angriff gewählt. Die Begründung hierzu liegt in der Tatsache, dass der Angriff in einem Testlabor durchgeführt wird und daher die Identifikation eines Angriffs nicht von Relevanz ist. Wie bereits in den vorhergehenden Kapiteln erwähnt, beschränkt sich die Master-Thesis auf die Schwachstellenanalyse der Funkprotokolle, welche in der Klassifizierung eines Penetrationstests im Bereich der Technik in die Einordnung als „sonstige Kommunikation“ fällt. Als letzter Punkt der Klassifizierung ist die Bestimmung des Ausgangspunktes festzulegen, von dem aus der Test initiiert wird. Im Falle dieser Ausarbeitung lässt sich dieser als ein „von außen“ durchzuführender Angriff beschreiben, da bei dem Test davon ausgegangen wird, keinerlei Zugriff auf die internen Strukturen der Umgebung zu besitzen.

Kriterium	**Wert**
1. Informationsbasis	Black-Box
2. Aggressivität	aggressiv
3. Umfang	fokussiert
4. Vorgehensweise	offensichtlich
5. Technik	sonstige Kommunikation
6. Ausgangspunkt	von außen

Tabelle 3.1: Klassifizierung der Vorgehensweise für die Phase der aktiven Eindringversuche (in Anlehnung an [52, 52])

4 Schwachstellenanalyse - Vorbereitung

In diesem Kapitel werden die Ziele und der Umfang der Schwachstellenanalyse definiert. Wie bereits in Kapitel 3 erwähnt, wird diese Phase nicht für jedes Funkprotokoll einzeln durchgeführt, sondern hat einen allgemeinen Charakter.

Zunächst sind die Ziele der Schwachstellenanalyse, die den Kern dieser Masterarbeit darstellen, zu definieren. Bereits in der Einleitung zu dieser Arbeit wurde dargestellt, dass das Ziel der Ausarbeitung darin besteht, die Schwachstellen der derzeit auf dem Markt verfügbaren IdD-Protokolle für die drahtlose Kommunikation im Smart Home Bereich zu analysieren (vgl. Kapitel 1.2). Der Umfang dieser Arbeit unterscheidet sich von den im Exposé für die Masterarbeit angegebenen Funkprotokollen dahingehend, dass nur drei der am häufigsten eingesetzten Protokolle analysiert werden. Die Ausarbeitung von neun Funkprotokollen, wie ursprünglich im Exposé angedacht, würde den Rahmen sprengen.

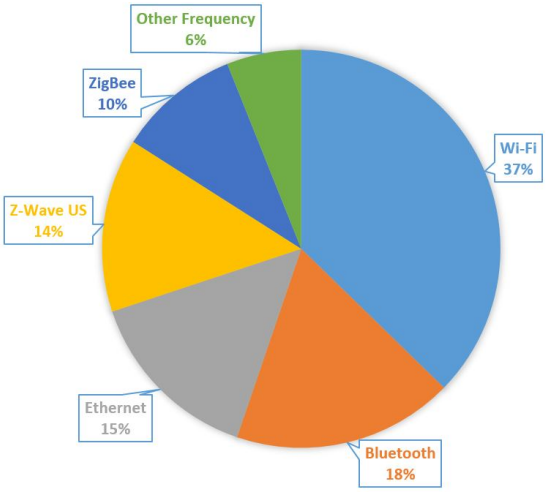

Abbildung 4.1: Statistik der genutzten Protokoll in Smart Home Geräten (eigene Darstellung, Datenquelle [53])

„Consumers are gradually upgrading to smart homes, and radio protocols such as Wi-Fi, Z-Wave, ZigBee, Thread and Bluetooth Low Energy are currently the preferred wireless protocols from a global perspective (vgl. [45]).“ So schreibt Chun Liew

2015 in einem Artikel über „The Smart Home radio protocols war“, den Krieg der Smart Home Funkprotokolle. Chun, Gründer von SmartHomeDB, einer Community für Smart Home Konsumenten, benennt in seinem Artikel die drahtlosen Protokolle ZigBee, Thread, Z-Wave, Wi-Fi und Bluetooth als die am häufigsten bevorzugt genutzten Protokolle weltweit. Vergleicht man dies mit einer aktuellen Statistik, die in Abbildung 4.1 dargestellt ist, so erkennt man, dass diese leicht von den Aussagen Liews abweicht. So stellen Wi-Fi, Bluetooth und Z-Wave aktuell die am meisten genutzten Protokolle der drahtlosen Kommunikation im Smart Home Bereich dar. Eine Schwachstellenanalyse zum Thema Wi-Fi, in Bezug auf die Standards Wi-Fi Protected Access (WPA) 2 und WPA 3, wurde bereits durch meinen Kommilitonen Ken Blendien durchgeführt und wird daher in dieser Ausarbeitung nicht betrachtet (vgl. [54]). Daher wird das nächste Funkprotokolle, ZigBee, in diese Ausarbeitung mit einbezogen. Zusammenfassend bilden also die drahtlosen Protokolle, ZigBee, Bluetooth und Z-Wave die Basis für die im Rahmen dieser Ausarbeitung zu untersuchenden Funkprotokolle. Das Kapitel über ZigBee stellt hierbei den größten Umfang dar, da neben der Schwachstellenanalyse auch ein Penetrationstest, in Form von aktiven Eindringversuchen, durchgeführt wird.

5 Schwachstellenanalyse - ZigBee

Dieses Kapitel behandelt die Schwachstellenanalyse des Funkprotokolls der ZigBee Spezifikation. Neben der Schwachstellenanalyse, welche die Bereiche Informationsbeschaffung, Risikoanalyse und Abschlussanalyse umfasst, wird in diesem Fall auch ein aktiver Eindringversuch durchgeführt.

5.1 Informationsbeschaffung

Der Bereich der Informationsbeschaffung ist in vier Teilbereiche unterteilt. Zunächst werden im Rahmen einer Einführung grundlegende Informationen über das zu analysierende Funkprotokoll gegeben. Neben der Darstellung der Funktionsweise werden auch die Sicherheitsmaßnahmen des Funkprotokolls analysiert. Hierbei werden vor allem die verfügbaren Netzwerktopologien des ZigBee-Protokolls, der Protokollstack aber auch die logischen Geräte, die von zentraler Bedeutung für ein ZigBee-Netzwerk sind, betrachtet. Auf der Grundlage der Ergebnisse der Sicherheitsmaßnahmen werden die Bedrohungen durch die Definition von Angriffsvektoren modelliert.

5.1.1 Einführung in das Funkprotokoll

ZigBee ist einer der am weitesten verbreiteten Standards für die drahtlose Kommunikation zwischen verschiedenen IdD-Geräten und wurde von vielen großen Unternehmen, wie beispielsweise Philips, übernommen. Hierbei bildet er einen offenen Standard für stromsparende und kostengünstige Wireless Personal Area Networks (WPANs), die Geräte, vor allem für den persönlichen Gebrauch, miteinander verbinden. Der Standard zielt darauf ab, ein bidirektionales und zuverlässiges Kommunikationsprotokoll für Anwendungen mit einer kurzen Reichweite von typischerweise 10 - 100 Metern, je nach Leistung und Umgebungsbedingungen, bereitzustellen. Bei Sub-GHz-Kanälen ist aber, bei direkter Sichtverbindung, auch eine Übertragungsreichweite von bis zu 1 km möglich. ZigBee basiert auf dem IEEE Standard 802.15.4-2011 und wird von der ZigBee Alliance, einem Konsortium von Unternehmen die das ZigBee-Protokoll standardisieren will, stetig weiterentwickelt. 2007, 2015 und 2017 wurde ZigBee durch die ZigBee PRO Spezifikation aktualisiert (vgl. [55]).

Letztere Spezifikation ist derzeit nur durch Mitglieder der Alliance einsehbar. Die letzte ZigBee Version ist ZigBee 3.0, die 2016 veröffentlicht wurde und auf ZigBee PRO 2015 und höher aufsetzt. Hierbei kann ZigBee in der Star-, Tree- oder Mesh-Topologie verwendet werden und unterstützt maximal 65.000 Knoten. ZigBee nutzt die Vorteile des leistungsstarken physikalischen Funkstandards IEEE 802.15.4-2011 und arbeitet in den nicht lizenzierten Industrial, Scientific and Medical (ISM) Bändern weltweit mit 2,4 GHz (global), 915 MHz (Amerika) und 868 MHz (Europa). Bei den Durchsatzraten von Rohdaten können 250 Kbits/s bei 2,4 GHz (16 Kanäle), 10 Kbits/s bei 915 - 921 Mhz (27 Kanäle) und 100 Kbits/s bei 868 Mhz (63 Kanäle) erreicht werden (vgl. [56]).

5.1.2 Funktionsweise / Grundlagen des Funkprotokolls

5.1.2.1 ZigBee Protokollstack

Der ZigBee Protokollstapel besteht aus vier Schichten – Physical Layer, Medium Access Control (MAC) Layer, Network Layer (NWK) und Application Layer (APL) – wie in Abbildung 5.1 dargestellt. Jede Schicht stellt eine Reihe von Diensten zur Verfügung, die der oberen Schicht über einen Service-Zugangspunkt zur Verfügung gestellt werden. Der Physical Layer und der MAC Layer werden durch den IEEE Standard 802.15.4-2011 geregelt. NWK und APL entsprechend durch den ZigBee Standard. Der Medium Access Control Layer steuert den Zugriff auf den Funkkanal über einen Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) Mechanismus, also einer Kollisionsvermeidung bei Zugriff mehrerer Geräte auf denselben Übertragungskanal. Weitere Aufgaben umfassen die Übertragung der Beacon Frames, die Synchronisation und die Bereitstellung eines zuverlässigen Übertragungsmechanismus. Abbildung 5.2 stellt die Einordnung des ZigBee Protokollstacks in das TCP/IP-Modell dar.

Nachfolgend werden die Schichten des ZigBee Protokollstacks und deren Aufgaben kurz umrissen (vgl. Abbildung 5.1, [57]):

- **Application Layer (APL):**
  Der Application Layer besteht aus dem Application Support Sub-Layer (APS), dem Application Framework und dem ZigBee Device Object (ZDO).
  - *Application Support Sub-Layer (APS):*
    Bietet eine Schnittstelle zwischen dem NWK und dem APL und stellt Dienste für die Datenübertragung – mittels der APS Data Entity (APS-DE) – und die Aufrechterhaltung von Sicherheitsbeziehungen – mittels

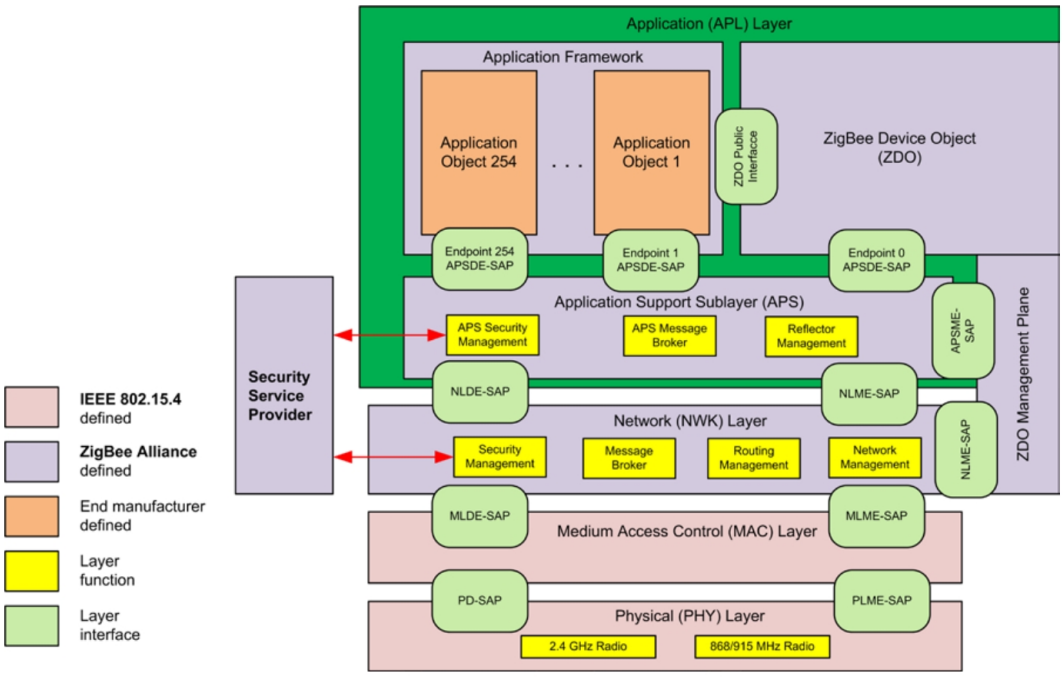

Abbildung 5.1: ZigBee Protokollstack (vgl. [57, 2])

der APS Management Entity (APSME) – bereit. Weiterhin ermöglicht der APS die Rahmensicherheit auf Basis von Link Keys oder dem Network Key zu realisieren. Zudem ist der APS für alle Verarbeitungsschritte verantwortlich, die erforderlich sind, um ausgehende Frames sicher zu übertragen, eingehende Frames sicher zu empfangen und kryptografische Schlüssel sicher zu erstellen und zu verwalten.

– *Application Framework:*
  Das Application Framework ist die Umgebung, in der die Anwendungsobjekte gehostet werden. Hier werden auch die Application Profiles definiert, die den Schlüssel zur Kommunikation zwischen Geräten in einem ZigBee-Netzwerk darstellen, indem sie Vereinbarungen über Nachrichten, Nachrichtenformate und Verarbeitungsaktionen beschreibt. Ein Beispiel für ein Profil wäre Home Automation. Dieses ZigBee Profil würde es den jeweiligen Geräte-Typen erlauben, Steuernachrichten auszutauschen um eine Home Automation Anwendung zu bilden.

– *ZigBee Device Object (ZDO):*
  ZDOs sind Anwendungen, welche in der Lage sind ZigBee End Devices, Router und Coordinators unter Verwendung von NWK und APS Primitiven zu implementieren. Die ZDOs stellen eine Basisklasse von Funktionen

dar, die eine Schnittstelle zwischen den Anwendungsobjekten, dem Geräteprofil und dem APS bildet. Das ZDO ist für die Initialisierung von APS, NWK und Security Service Provider sowie dem Zusammenstellen von Konfigurationsinformationen aus den Endanwendungen zur Bestimmung und Implementierung von beispielsweise Discovery, Security Management und Network Management verantwortlich.

- **Network Layer (NWK):**
  Der NWK gewährleistet den korrekten Betrieb des IEEE 802.15.4-2011 MAC Layers und stellt eine geeignete Serviceschnittstelle zur APL bereit. Zur Anbindung an die Anwendungsschicht beinhaltet die Netzwerkschicht konzeptionell zwei Service-Einheiten, die Network Layer Data Entity (NLDE) als Daten Service und die Network Layer Management Entity (NLME) als Management Service, die die notwendigen Funktionalitäten bereitstellen. Im Speziellen stellt der NLDE den Service zur Generierung der Network Level Protocol Data Unit (PDU) (NPDU) sowie dem topologie-spezifischen Routing bereit. Der NLME wiederum stellt Services zum Konfigurieren neuer Geräte, Starten des Netzwerkes, Adressierung, Nachbarerkennung, Routenermittlung, Empfangskontrolle, Routing oder dem Beitreten, Rejoining oder Verlassen des Netzwerkes bereit. Der NWK ist für die Verarbeitungsschritte verantwortlich, die erforderlich sind, um ausgehende Frames sicher zu übertragen und eingehende Frames sicher zu empfangen. Der Frame-Schutzmechanismus des NWK verwendet den Advanced Encryption Standard (AES) und CCM* (Enhanced Counter with CBC-MAC Mode of Operation) für Authentifizierung und Vertraulichkeit.

- **Medium Access Control (MAC) Layer:**
  Zu den Aufgaben des MAC Layers gehört neben der Kontrolle des Zugangs zum Funkkanal über den CSMA/CA-Mechanismus, die Übertragung von Beacon Frames, die Synchronisation und die Bereitstellungen eines zuverlässigen Übertragungsmechanismus. Die Sicherheit der Schicht basiert auf den IEEE Standard 802.15.4-2011, ergänzt durch CCM*, um entsprechend Verschlüsselungs- und Integritätsfunktionen bereitzustellen.

- **Physical Layer:**
  Wie schon in der Einführung in das Funkprotokoll (vgl. Kapitel 5.1.1) angesprochen, arbeitet der Physical Layer auf zwei getrennten Frequenzbereichen. 868/915 Mhz und 2,4 GHz. Weiterhin ist der Physical Layer für die Paketgenerierung, den Paketempfang, die Datentransparenz und das Energiemanagement verantwortlich.

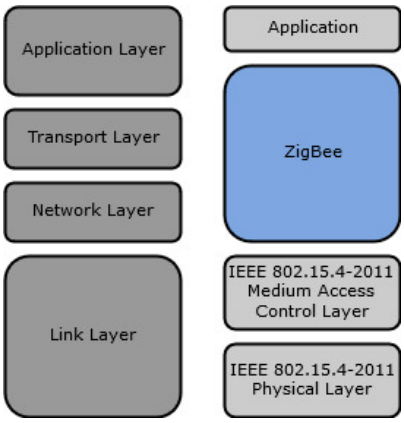

Abbildung 5.2: ZigBee Referenzmodell und Einordnung in das TCP/IP-Modell (eigene Darstellung)

5.1.2.2 Logische Geräte nach der ZDO Definition

Das ZDO definiert drei Arten von logischen Geräten, die jeweils eine bestimmte Rolle spielen. Das Zusammenspiel der logischen Geräte ist in Abbildung 5.3 dargestellt. Nachfolgend werden die einzelnen logischen Geräte kurz dargestellt (vgl. [57]):

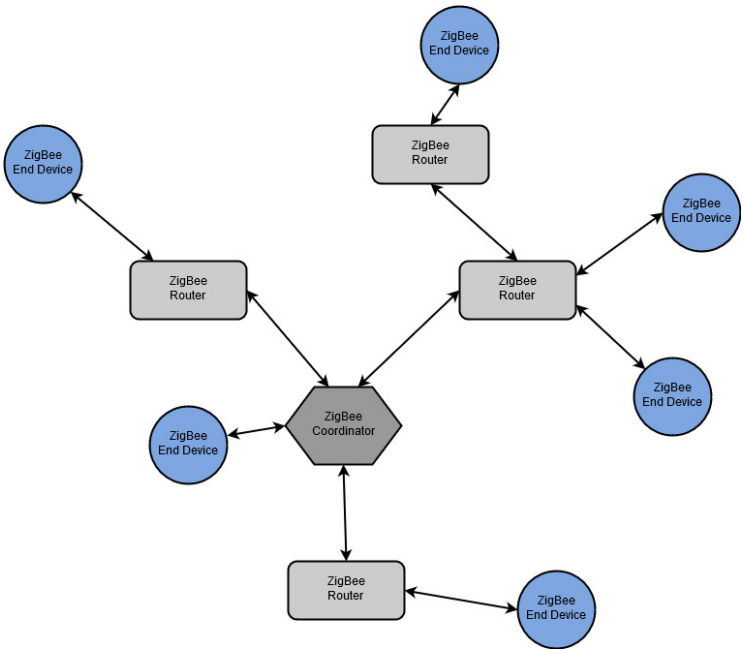

Abbildung 5.3: ZigBee logische Geräte nach der ZDO Definition (eigene Darstellung, angelehnt an [57, 325], [58])

- **ZigBee Coordinator:**
  Der ZigBee Coordinator ist ein Gerät, das für die Einrichtung, Ausführung und Verwaltung des gesamten ZigBee Netzwerks verantwortlich ist. Hierbei ist er vor allem für die Konfiguration der Sicherheitsstufe des Netzwerks und die Konfiguration der Adresse des Trust Centers verantwortlich. Im Normalfall ist die Adresse die eigene Adresse des ZigBee Coordinators. Zudem führt er eine Liste der derzeit verbundenen Geräte und unterstützt damit die Orphan Scan und Rejoin Prozesse, sodass zuvor zugeordnete Geräte wieder dem Netzwerk beitreten können. Pro ZigBee Netzwerk kann es jeweils nur einen Coordinator geben, welcher bei Bedarf auch als Router fungieren kann.

- **ZigBee Router:**
  Der ZigBee Router fungiert als zwischengeschalteter Knoten, der für die Weiterleitung von Paketen zwischen den ZigBee End Devices und dem ZigBee Coordinator verantwortlich ist. Um dem Netzwerk bei aktiver Sicherheit beizutreten, benötigen sich entsprechend die Berechtigung des Trust Centers. Zudem können ZigBee Router auch als Endgeräte eingesetzt werden. Unter bestimmten Voraussetzungen können Router anderen Routern und End Devices die Erlaubnis erteilen, dem Netzwerk beizutreten. Wie der ZigBee Coordinator führen auch die ZigBee Router eine Liste der derzeit verbundenen Geräte und unterstützen dadurch ebenfalls die Orphan Scan und Rejoin Prozesse.

- **ZigBee End Device:**
  Ein ZigBee End Device stellt einen Sensorknoten dar, der eine Umgebung überwacht und Informationen über diese sammelt. Im Gegensatz zu ZigBee Coordinator und Router, welche nie in den Ruhemodus schalten dürfen, werden ZigBee End Devices generell mit niedriger Leistung beziehungsweise Batterien betrieben. Dies erlaubt es ihnen in den Ruhemodus wechseln, um Energie zu sparen, wenn keine Aktivitäten in der Umgebung zu überwachen sind. ZigBee End Devices können weder den Verkehr weiterleiten noch anderen Knoten erlauben, sich dem Netzwerk im Allgemeinen anzuschließen

5.1.2.3 Netzwerktopologien

ZigBee unterstützt drei Arten an WPAN-Topologien. Bei der Auswahl der Topologie sollte man sich entsprechend Gedanken machen. So kann es beispielsweise wichtig sein, zu Wissen, welche Knoten leitungs- oder batteriebetrieben sind, was die erwartete Batterielebensdauer ist oder welche Menge an Netzwerkverkehr erforderlich ist. Nachfolgend werden die drei unterstützen Topologien in Bezug auf ZigBee näher

erläutert (vgl. Abbildung 5.4, [57, 2]). Ein allgemeiner Überblick über Netzwerktopologien wurde bereits in Kapitel 2.3 gegeben.

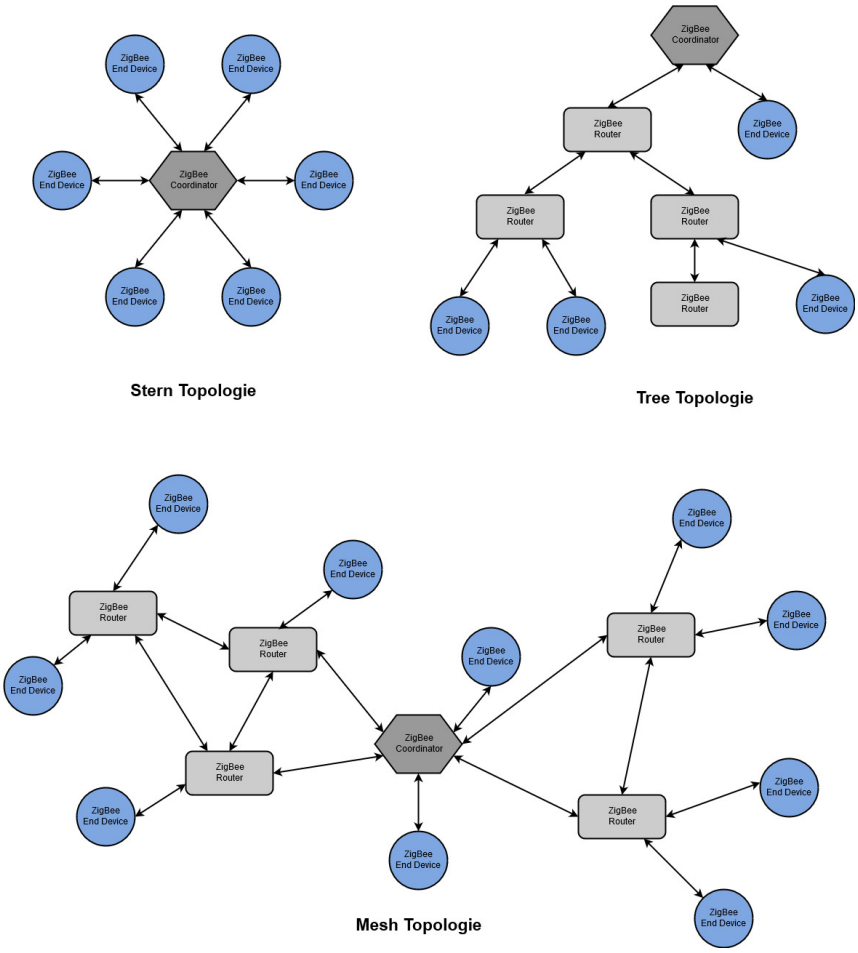

Abbildung 5.4: ZigBee Netzwerktopologien (eigene Darstellung, basierend auf [57, 2])

- **Star:**
  Ist das ZigBee-Netzwerk in Form der Star Topologie aufgebaut, so wird es nur von einem einzigen ZigBee Coordinator verwaltet. Dieser ist dann entsprechend für das Routing der Pakete und die ZigBee End Devices verantwortlich. Letztere können nur über den Coordinator miteinander kommunizieren. Der große Nachteil dieser Topologie ist der Single Point of Failure des ZigBee Coordinators. Fällt dieser aus, bricht das ganze Netzwerk zusammen.

- **Tree:**
  Im Falle der Tree Topologie ist der ZigBee Coordinator für den Aufbau des Netzwerks und der Auswahl von wichtigen Netzwerkparametern verantwortlich. Erweiterungen des Netzwerks erfolgen durch den Einsatz der ZigBee Rou-

ter. Daten- und Kontrollnachrichten werden von den Routern mittels einer hierarchischen Routingstrategie durch das Netzwerk transportiert. Weiterhin ist eine beacon-orientierte Kommunikation nach IEEE Standard 802.15.4 möglich. Nachteil der Tree Topologie ist, dass untergeordnete Knoten unerreichbar werden, wenn ihr übergeordneter Knoten nicht mehr verfügbar ist.

- **Mesh:**
  Die Mesh Topologie ermöglicht eine vollständige Peer-To-Peer Kommunikation. Sie verfügt über einen einzigen ZigBee Coordinator, mehrere ZigBee Router zur Erweiterung des Netzwerks und optionale ZigBee End Devices. Wie auch in der Tree Topologie ist der ZigBee Coordinator für den Aufbau des Netzwerks und der Auswahl von wichtigen Netzwerkparametern verantwortlich. ZigBee Router können zwar als End Devices fungieren, sind aber nicht in der Lage, Beacons nach dem IEEE Standard 802.15.4 zu senden. Diese Topologie weist keinen Single Point of Failure auf und bleibt auch bei Ausfall eines Coordinators verfügbar. Nachteil ist jedoch deren Komplexität und Einrichtung.

5.1.2.4 Applikationsprofile

Damit Geräte unterschiedlicher Hersteller nahtlos miteinander kommunizieren können, gibt es sogenannte Applikationsprofile. Als Beispiel kann hier das ZigBee Light Link (ZLL) Applikationsprofil genannt werden, welches der Beleuchtungsindustrie, auch im Smart Home Bereich, einen globalen Standard für interoperable und sehr einfach zu bedienende Produkte für die Beleuchtung und Steuerung von Verbrauchern bietet (vgl. [59]). ZigBee definiert über 130 Gerätetypen und zugehörige Befehle, um sicherzustellen, dass Geräte von mehreren Anbietern nahtlos zusammenarbeiten (vgl. [60]). Ab ZigBee 3.0 wurden alle Applikationsprofile zu einem einzigen Applikationsprofil zusammengefasst. Beschrieben sind diese in der ZigBee Cluster Library Spezifikation der ZigBee Alliance (vgl. [61]).

5.1.3 Sicherheitsmaßnahmen

Die Sicherheit von ZigBee basiert auf einer symmetrischen Schlüsselkryptographie, bei der zwei Parteien die gleichen Schlüssel zur Kommunikation verwenden. Alle Knoten im Netzwerk verwenden in der Regel den gleichen Schlüssel. Es ist jedoch möglich, einen individuellen Link Key zwischen einem bestimmten Knotenpaar zu verwenden und so die Kommunikation von der normalen Kommunikation im Netzwerk zu trennen. ZigBee verwendet das hochsichere 128-Bit AES-basierte Verschlüsselungssystem (vgl. [62, 30]). Wie in der Einleitung erwähnt, basiert das

ZigBee-Protokoll auf dem IEEE-Funkstandard 802.15.4-2011, der die physikalische Schicht und die MAC Schicht beinhaltet (vgl. Kapitel 5.1.1). ZigBee baut die NWK und APL auf diesen Schichten auf. Als kostengünstiges Protokoll geht ZigBee von einem „Open Trust“-Modell aus, bei dem sich die Schichten des Protokollstacks gegenseitig vertrauen. Daher existiert der kryptografische Schutz nur zwischen Geräten, nicht aber zwischen den verschiedenen Schichten eines Geräts. Das „Open Trust“-Modell hat auch weitreichende Folgen. Es ermöglicht die Wiederverwendung von Schlüsseln zwischen Schichten auf demselben Gerät. Um die Interoperabilität von Geräten zu vereinfachen, verwendet ZigBee die gleiche Sicherheitsstufe für alle Geräte in einem Netzwerk. Darüber hinaus ist die Sicherheitsstufe innerhalb der Schichten eines Geräts gleich. Weiterhin wird das Prinzip, „the layer that originates a frame is responsible for initially securing it“, festgelegt. Jede Schicht, aus der ein Frame entspringt, ist somit für deren initiale Sicherheit verantwortlich (vgl. [62, 376]). In einem Dokument mit dem Thema „ZigBee: Securing the Wireless IoT“ beschreibt die „ZigBee 3.0 Task Force“, die zur ZigBee Alliance gehört, folgende zusätzliche Techniken, die zur Sicherheit beitragen sollen (vgl. [63, 3]). Zum einen beinhaltet jeder ZigBee Befehl einen sogenannten Frame Counter, mit dem Replay Angriffe (vgl. Kapitel 2.1.3.1) verhindert werden sollen. Der empfangende Endpunkt überprüft somit immer den Frame Counter und ignoriert doppelte Nachrichten. Zum anderen unterstützt ZigBee die „Frequenzagilität“. Das ZigBee-Netzwerk kann auf einen anderen Kanal (Frequenz) verlegt werden, wenn der aktuelle Kanal beispielsweise durch einen Jamming Angriff beeinträchtigt wird (vgl. Kapitel 2.1.3.1).

5.1.3.1 Sicherheitsmodell

Um ein breites Anwendungsspektrum abzudecken und gleichzeitig ein optimales Gleichgewicht zwischen Sicherheit, Benutzerfreundlichkeit, Kosten und Akkulaufzeit zu gewährleisten, bietet ZigBee zwei Netzwerkarchitekturen und entsprechende Sicherheitsmodelle an: Das „Centralized Security ZigBee-Netzwerk“ und das „Distributed Security ZigBee-Netzwerk“ (vgl. Abbildung 5.5). Diese unterscheiden sich darin, wie sie neue Geräte in das Netzwerk aufnehmen und wie sie Nachrichten im Netzwerk schützen (vgl. [57, 380], [63, 1]).

- **Distributed Security ZigBee-Netzwerk:**
  Das Distributed ZigBee-Netzwerk, welches einfacher zu konfigurieren aber weniger Sicherheit als das Centralized ZigBee-Netzwerk bietet, besteht aus ZigBee End Devices und ZigBee Routern. Erkennt ein ZigBee Router beim Hochfahren kein vorhandenes Netzwerk, kann er ein „Distributed Security ZigBee-

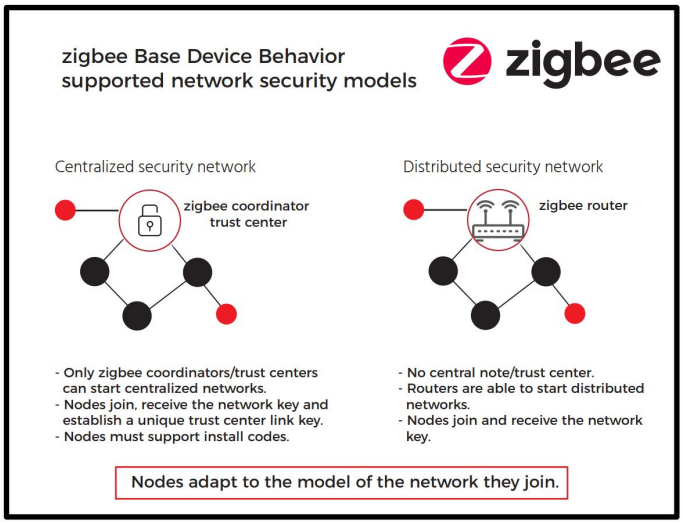

Abbildung 5.5: ZigBee Centralized Security versus Distributed Security (vgl. [63, 2])

Netzwerk“ bilden. Jeder ZigBee Router kann Network Keys ausgeben. Treten neue ZigBee Router oder End Devices dem Netzwerk bei, sendet der im Netzwerk befindliche Router den Network Key verschlüsselt an die beigetretenen Geräte. Alle Geräte im Netzwerk verwenden den gleichen Network Key, um Nachrichten zu verschlüsseln.

- **Centralized Security ZigBee-Netzwerk:**
  Ein Centralized ZigBee-Netzwerk sorgt für mehr Sicherheit, ist aber auch komplizierter, da es einen dritten Gerätetyp, das Trust Center (TC), beinhaltet, welches in der Regel der ZigBee Coordinator ist. Hierbei bildet das TC das Centralized ZigBee-Netzwerk und ermöglicht es ZigBee Router und End Devices dem Netzwerk beizutreten, wenn diese über die entsprechenden Berechtigungen verfügen. Hierfür erstellt das TC für jedes Gerät einen eindeutigen Link Key, das dem Netzwerk beitreten will. Nur das TC ist in der Lage, den Network Key auszugeben. Um an einem Centralized ZigBee-Netzwerk teilzunehmen, müssen alle Geräte mit einem Link Key vorkonfiguriert sein, damit das TC in der Lage ist, den Network Key verschlüsselt an das neu zu verbindende Gerät zu übermitteln.

5.1.3.2 Sicherheitsannahmen

Die ZigBee Alliance beschreibt in ihrer ZigBee Spezifikation auch die Spezifikation der Sicherheitsdienste (vgl. [57, 375 ff.]). Folgt man dieser Beschreibung, so hängt die Sicherheit von ZigBee, neben dem „Open Trust“-Modell, auch von folgenden

Annahmen ab.

1. Als Erstes ist hier die sichere Aufbewahrung der symmetrischen Schlüssel zu nennen. ZigBee geht davon aus, dass geheime Schlüssel außerhalb der Geräte nicht ungesichert verfügbar sind, d.h. die gesamte Übertragung von Schlüsseln muss verschlüsselt werden. Es gibt jedoch Ausnahmen. Zum einen kann der Schlüssel, der beim initialen Schlüsseltransport verwendet wird, ein bekannter Schlüssel sein, zum anderen kann der anfängliche Schlüsseltransport auch mit einem vorab freigegebenen, geheimen Schlüssel durchgeführt werden, der außerhalb des ZigBee-Netzwerks vorkonfiguriert ist. Ersteres führt zu einer kurzen Schwachstelle, bei der der Schlüssel auch von anderen Geräten empfangen werden kann. Hinsichtlich des vorab freigegebenen, geheimen Schlüssels gilt der Vorbehalt, dass man auf Grund der geringen Kosten der Ad-hoc-Netzwerkgeräte nicht von der Verfügbarkeit manipulationssicherer Hardware ausgehen kann. Mittels einem Node Tampering Angriff (vgl. Kapitel 2.1.3.1) wäre somit unter Umständen der Zugriff auf geheimes Schlüsselmaterial möglich.

2. Alle ZigBee Router und End Devices sollen die beiden Sicherheitsmodelle, Distributed Security und Centralized Security, unterstützen und sich jeweils an das Sicherheitsschema anpassen, welches das zugehörige ZigBee-Netzwerk nutzt. Bei Einsatz eines ZigBee Coordinators gilt entsprechend nur das Centralized Security Schema (vgl. [64, 28]).

3. Implementierungen von Sicherheitsprotokollen, wie beispielsweise die Einrichtung von Schlüsseln, sollten das komplette Protokoll ordnungsgemäß ausführen, ohne Schritte davon auszulassen. Die Nutzung von Zufallsgeneratoren für Nummern sollten den Erwartungen entsprechend funktionieren.

5.1.3.3 Sicherheitsarchitektur

Wie bereits in Kapitel 5.1.2.1 erwähnt, baut ZigBee die NWK und APL Schichten (vgl. Abbildung 5.1) auf den IEEE 802.15.4 Physical und MAC Layer auf. Der APL umfasst hierbei den APS, das ZDO sowie Anwendungen. Die Architektur beinhaltet dabei Sicherheitsmechanismen auf zwei Schichten des Protokollstapels: NWK und APS. Die Sicherheitsmechanismen des NWK und des APL, letzteres mittels APS, wurden bereits in Kapitel 5.1.2.1 kurz angesprochen. Abbildung 5.6 stellt einen ZigBee Frame dar, bei dem die Sicherheitsmechanismen auf NWK Ebene aktiviert sind. In diesem Fall wird dem Frame ein Auxiliary Header und ein Integrity Code, dem Message Integrity Code (MIC) hinzugefügt. Der MIC ist dafür zuständig, die

Authentizität von Nachrichten zu gewährleisten. Abbildung 5.7 hingegen zeigt einen ZigBee Frame, bei dem die Sicherheitsmechanismen auf dem APL Layer aktiviert sind. Der APS ist in diesem Fall für die Sicherheit des Frames verantwortlich.

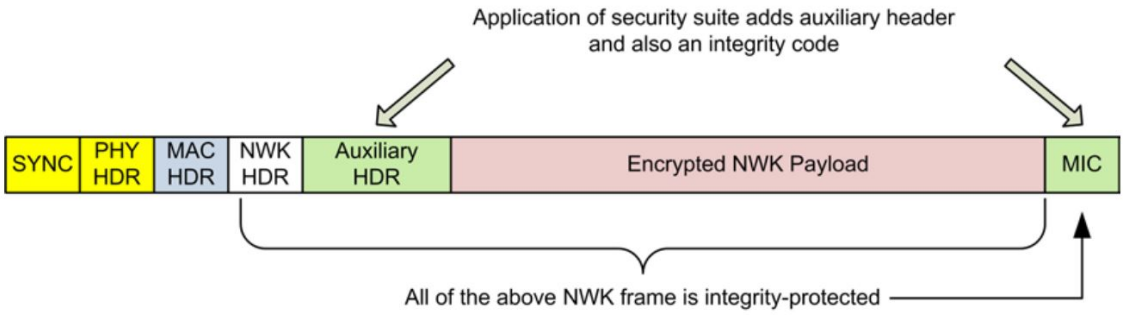

Abbildung 5.6: ZigBee Frame mit Sicherheit auf der NWK Ebene (vgl. [57, 378])

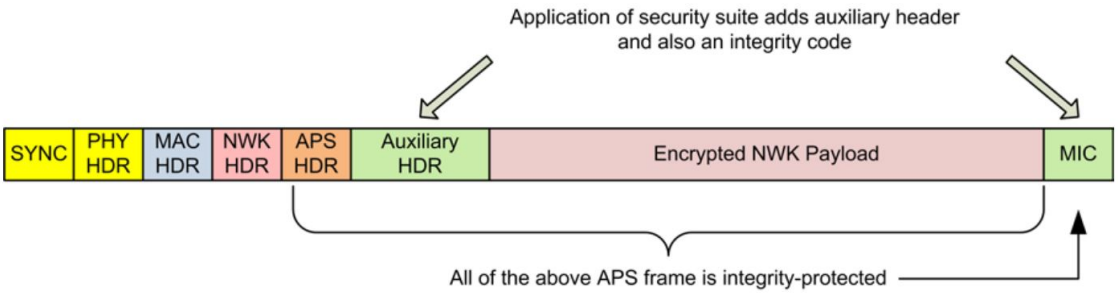

Abbildung 5.7: ZigBee Frame mit Sicherheit auf der APS Ebene (vgl. [57, 379])

Wie in den Abbildung 5.6 und 5.7 erkenntlich, wird den Frames auf NWK beziehungsweise APS Ebene jeweils ein Auxiliary Header hinzugefügt. Dieser enthält wichtige Informationen zur Sicherheit des versendeten Frames und besteht aus Security Control, Frame Counter, Source Address und Key Sequence Number (vgl. Abbildung 5.8). Nachfolgend werden die Felder des Auxiliary Header kurz näher betrachtet (vgl. [57, 424 ff.]).

Octets: 1	**4**	**0/8**	**0/1**
Security control	Frame counter	Source address	Key sequence number

Abbildung 5.8: ZigBee Auxiliary Header Format (vgl. [57, 424])

- **Security Control:**
  Das Security Control Feld ist wiederum in vier Felder untergliedert: Security

Level, Key Identifier, Extended Nonce und Reserved. Letzteres dient aktuell als Reserve und ist nicht näher beschrieben.

- *Security Level:*
  Der Security Level bestimmt, wie ein Frame abgesichert wurde. Zudem gibt er auch an, ob die Payload verschlüsselt ist oder nicht und inwieweit die Datenauthentizität über den Frame bereitgestellt wird. Letzteres spiegelt sich in der Länge des MIC wieder.
- *Key Identifier:*
  Dient zur Identifizierung des Schlüssels, der zum Schutz des Frames verwendet wurde. Hierbei wird zwischen den folgenden Werten unterschieden: *0x00* (Data Key), *0x01* (Network Key), *0x02* (Key-Transport Key) und *0x03* (Key-Load Key)
- *Extended Nonce:*
  Das Extended Nonce Feld gibt an, ob die Source Address im Auxiliary Header vorhanden ist. Ist dies der Fall, so ist der Wert *0x1* gesetzt. Anderenfalls ist der Wert *0x0*.

- **Frame Counter:**
  Der Frame Counter soll verhindern, dass Frames mit dem selben Counter mehrmals hintereinander geschickt werden können. Wie bereits zu Beginn des Abschnitts der Sicherheitsmaßnahmen angesprochen, dient der Frame Counter dem Schutz gegen Replay Angriffe (vgl. Kapitel 5.1.3).

- **Source Address:**
  Die Source Address darf nur dann gesetzt sein, wenn das Extended Nonce Sub-Feld des Security Level Felds entsprechend auf *0x1* gesetzt ist. Wenn gesetzt, enthält die Source Address die 64-Bit Adresse des Gerätes, welches für den Schutz des Frames verantwortlich ist.

- **Key Sequence Number:**
  Die Key Sequence Number darf ebenfalls nur dann gesetzt sein, wenn das Key Identifier Sub-Feld im Security Level Feld den Wert *0x01* aufweist. Es handelt sich somit um den Network Key. Die Key Sequence Number gibt die Sequenznummer des aktuell gültigen Network Key an, welcher für die Verschlüsselung beziehungsweise Entschlüsselung verwendet werden soll.

5.1.3.4 AES-CCM*

Wie bereits in Kapitel 5.1.2.1 angesprochen, nutzt ZigBee die AES Verschlüsselung zusammen mit CCM*. Letzterer ist ein generischer kombinierter Verschlüsselungs- und Authentifizierungsblock-Chiffriermodus, der für die Verwendung mit einer Blockgröße von 128-Bit definiert ist. Hierbei nutzt ZigBee AES-128 für die Verschlüsselung (vgl. [57, 456]). Hinsichtlich der Länge des MIC, welcher für die Authentizität von Nachrichten verantwortlich ist, kann festgestellt werden, dass hier für CCM* eine variable Länge möglich ist. Die Längen beschränken sich hierbei auf 0, 4, 8 oder 16 Byte (vgl. Kapitel 5.1.3.3, [57, 425]). Abbildung 5.9 zeigt die Rolle von AES-CCM* bei der Datenauthentifizierung und Vertraulichkeit. Der Klartext wird auf Senderseite in Form von 127-Bit Datenblöcken eingegeben und mittels AES-CCM* entsprechend verschlüsselt. Neben den Daten wird für die Verschlüsselung auch noch der Schlüssel, beispielsweise der Global Trust Center Link Key, sowie die Nonce benötigt. Die Nonce ist ein 13-Oktett String, der aus den Feldern Security Control, Frame Counter und Source Address des Auxiliary Frame Header gebildet wird (vgl. [65, 128 f.]). In Kapitel 5.1.3.3 sind die einzelnen Felder näher beschrieben. Abbildung 5.10 stellt den Aufbau der CCM* Nonce grafisch dar.

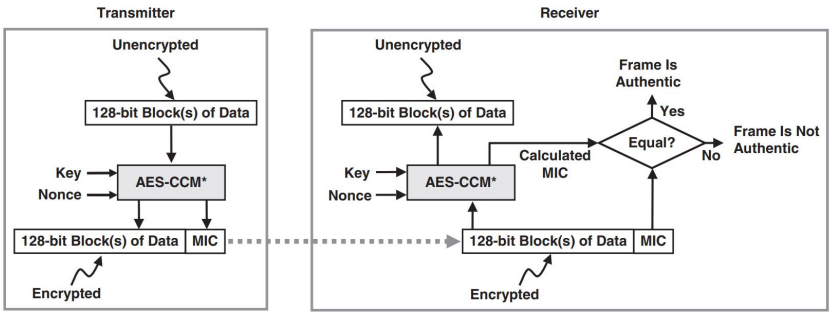

Abbildung 5.9: ZigBee Daten Authentifizierung mittels MIC (vgl. [65, 128])

5.1.3.5 Sicherheitsschlüssel

Die Sicherheit innerhalb eines ZigBee-Netzwerks basiert auf dem Network Key und den Link Keys. Der Link Key ist ein 128-Bit Schlüssel der von zwei Geräten gemeinsam im Zuge der Unicast-Kommunikation zwischen APL-Peer-Entitäten genutzt wird. Die Broadcast-Kommunikation und jede Kommunikation auf dem NWK wird durch einen 128-Bit Network Key gesichert, der von allen Geräten im Netzwerk gemeinsam genutzt wird. Der vorgesehene Empfänger ist sich hierbei immer über die genaue Sicherheitsvereinbarung bewusst, d.h. er weiß, ob ein Frame mit einem Link

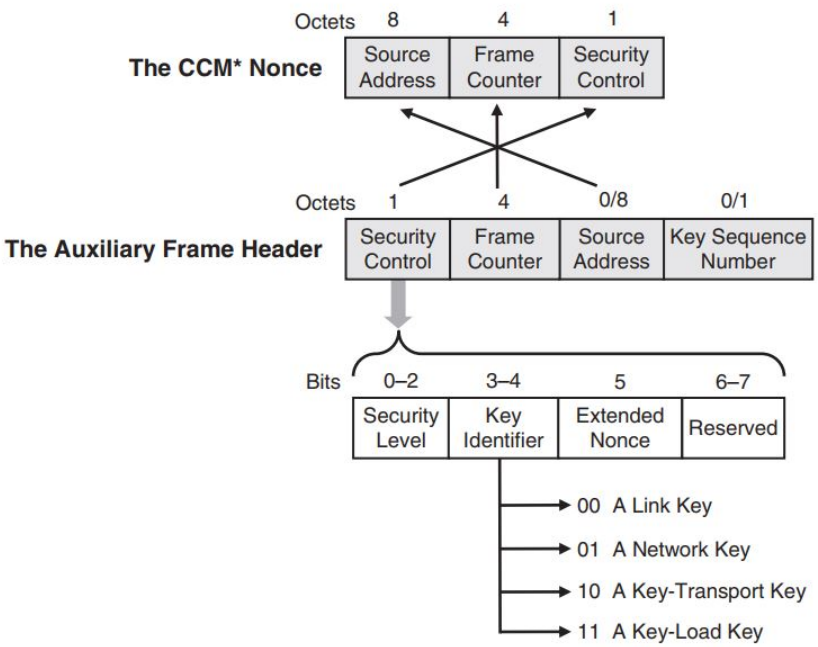

Abbildung 5.10: ZigBee Auxiliary Header Format und die CCM* Nonce (vgl. [65, 129])

Key oder einem Network Key geschützt ist. Bei den Link Keys werden zwischen Global Link Keys und Unique Link Keys unterschieden. Je nach Typ werden die TC Nachrichten vom Gerät unterschiedlich behandelt. Nachfolgende Auflistung gibt einen Überblick über die unterschiedlichen Arten an Link Keys (vgl. [57, 377 f.], [62, 94]):

- **Global Trust Center Link Key:**
  Der Global Trust Center Link Key sichert den Beitritt von Geräten in ein Centralized Security ZigBee-Netzwerk ab. Standardwert des Global Trust Center Link Keys ist der Folgende: 5A 69 67 42 65 65 41 6C 6C 69 61 6E 63 65 30 39 (Hexadezimal für *ZigBeeAlliance09*). Möglich ist auch ein vom Hersteller definierter Key, der nur Geräte desselben Herstellers den Beitritt zum Netzwerk erlaubt.

- **Unique Trust Center Link Key:**
  Der Unique Trust Center Link Key wird exklusiv zwischen dem TC und einem Gerät im Netzwerk geteilt. Er kann für den Beitritt in das Netzwerk genutzt werden und ist für die Unicast-Kommunikation auf dem APL verantwortlich. Er wird nur in Centralized Security ZigBee-Netzwerken verwendet, da es in einem Distributed Security ZigBee-Netzwerk kein eindeutiges TC gibt.

- **Distributed Global Link Key:**

Der Distributed Global Link Key wird verwendet, um Geräte einem Distributed Security ZigBee-Netzwerk ohne eigenes TC hinzuzufügen. Hierbei wird der Schlüssel bereits bei der Herstellung auf dem Gerät gesetzt.

- **Application Link Key:**
  Wird für die Kommunikation zwischen zwei Geräten im Netzwerk genutzt und ist für die Sicherheit der Kommunikation auf dem APL verantwortlich. Hierbei wird der Key vom TC in Verbindung mit den MAC-Adressen der beiden Knoten erzeugt. Anschließend erfolgt eine Verschlüsselung mit dem Network Key und, falls vorhanden, mit dem Unique Trust Center Link Key für jeden Knoten um den Schlüssel zu jedem Knoten zu transportieren.

- **Install Code Link Key:**
  Link Key, der aus einem Installationscode abgeleitet wurde um einen Unique Trust Center Link Key für den Beitritt zum Netzwerk zu generieren (vgl. [64, 29]).

In gesicherten Netzwerken gibt es in der Regel oftmals eine Vielzahl an verschiedenen Sicherheitsdiensten. Zur Verhinderung von Sicherheitslücken sollte daher die Wiederverwendung von Schlüsseln zwischen verschiedenen Sicherheitsdiensten vermieden werden. ZigBee setzt daher auf die Verwendung von eigenen Schlüsseln je Sicherheitsdienst, die sich vom Link Key über eine Einwegfunktion ableiten lassen. Folgende Schlüssel werden aus dem Link Key abgeleitet (vgl. [57, 9], [57, 378], [57, 387]):

- **Key-Transport Key:**
  Wie der Name schon ausdrückt, wird der Schlüssel für den Transport von Nachrichten genutzt, welche einen Schlüssel beinhalten.

- **Key-Load Key:**
  Der Key-Load Key wird für den Schutz des Transports von Link Keys verwendet.

- **Data Key:**
  Der Data Key dient dem Schutz beim Transport von NWK und APL Nachrichten.

Alle vom Link Key abgeleiteten Schlüssel teilen sich den selben Frame Counter. Zusätzlich teilen sich alle Schichten des ZigBee Protokolls den aktiven Network Key sowie den eingehenden und ausgehenden Frame Counter (vgl. [57, 427]). Dieser Frame Counter darf nach der ZigBee Spezifikation auch durch eine Rücksetzung auf

Werkseinstellungen nicht verändert werden (vgl. [57, 385]). In älteren Spezifikation von ZigBee gab es bezüglich der Verwaltung beziehungsweise Speicherung des Frame Counters keine Vorgaben.

5.1.3.6 Secure Network Join Prozedur

Abbildung 5.11 stellt den Ablauf des Beitritts in ein gesichertes Netzwerk dar. Diese Prozedur soll im nachfolgenden Teil näher erläutert werden (vgl. [57, 429 ff.]).

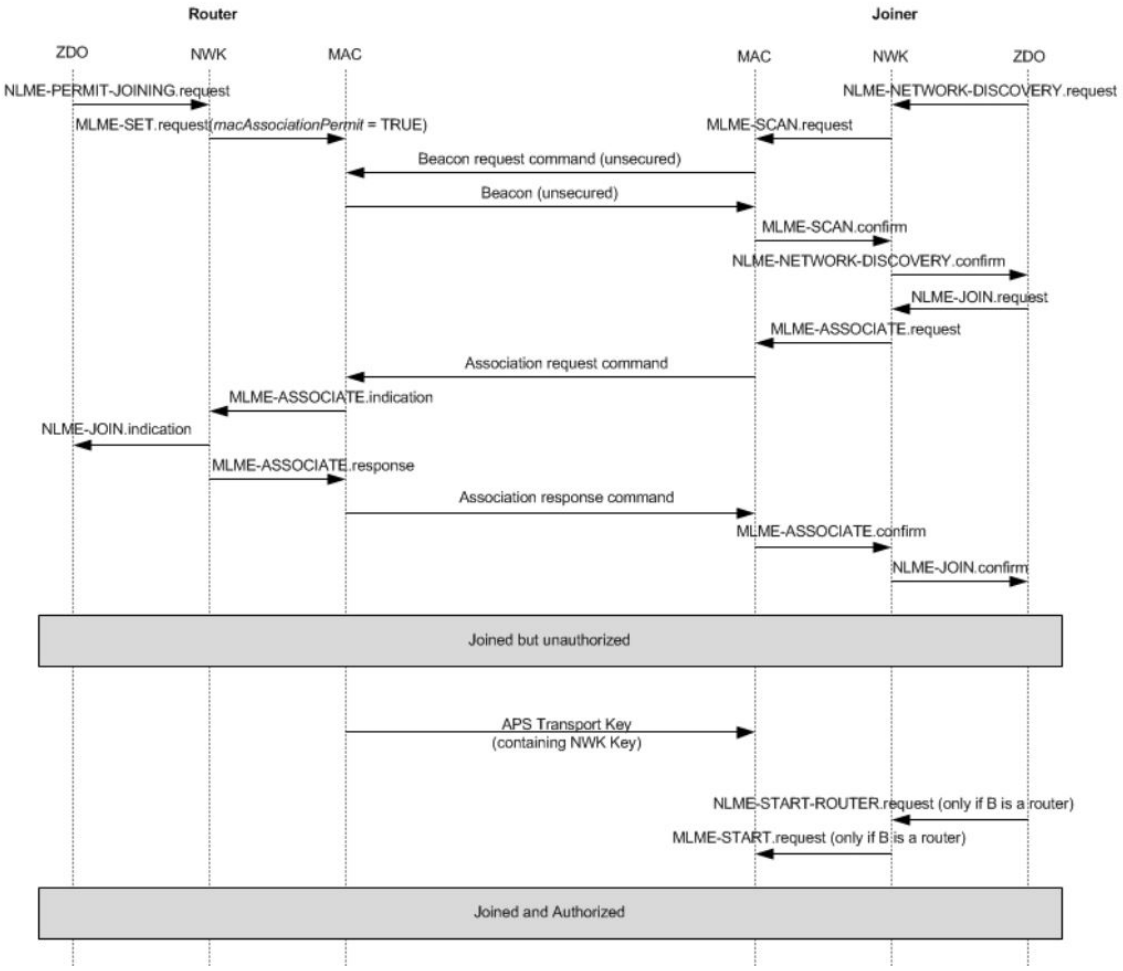

Abbildung 5.11: ZigBee Secure Network Join Prozedur (vgl. [57, 429])

Der Join-Vorgang kann durch das Gerät (Joiner), welches dem Netzwerk beitreten will, über eine NLME-NETWORK-DISCOVERY.request Primitive eingeleitet werden. Diese ruft wiederum eine Medium Access Control Sub-Layer Management Entity (MLME)-SCAN.request Primitive auf, welche in einem Beacon Request Frame resultiert. Nahe gelegene ZigBee Router antworten auf diesen Beaceon Request

entsprechend. Sobald die MAC-Subschicht den Abschluss des Scans durch Ausgabe der MLME-SCAN.confirm Primitive signalisiert, gibt die NWK-Schicht die NLME-NETWORK-DISCOVERY.confirm Primitive mit einer Beschreibung jedes gehörten Netzwerks aus. Jede Netzwerkbeschreibung enthält die ZigBee-Version, das Stack-Profil, die Extended PAN ID, die PAN ID, den logischen Kanal und Informationen darüber, ob ein Beitritt erlaubt ist (vgl. [57, 306]). Abbildung 5.11 zeigt die Annahme, dass der Beitritt in das Netzwerk entsprechend erlaubt ist. Der Joiner entscheidet sich für ein Personal Area Network (PAN) und stellt eine NLME-JOIN.request Primitive zur Verfügung um diesem beizutreten. Dies sorgt für einen „Association Request“ Befehl, welcher an den ZigBee Router gesendet wird. Dieser Befehl enthält „Capability Information“ wie beispielsweise den Gerätetyp, die Energiequelle aber auch Informationen ob Nachrichten erhalten werden können, wenn das Gerät „Idle“ ist (vgl. [57, 307]). Der „Association Request“ wird mit einem „Association Response“ Befehl abgeschlossen. Das Gerät ist damit dem gesicherten Netzwerk beigetreten und befinden sich im „Joined but unauthorized“ Status. Damit dieses autorisiert wird, muss es über den „APS Transport Key“ Befehl den aktiven Network Key erhalten. Der „APS Transport Key“ Befehl beschreibt, wie ein Schlüssel zu einem Gerät transportiert werden soll. Hierbei wird der Schlüssel von einer sogenannten „Key Source“ an ein Gerät gesendet. In einem Centralized Security ZigBee-Netzwerk ist eine solche „Key Source“ das TC, in einem Distributed Security ZigBee-Netzwerk die einzelnen ZigBee Router (vgl. [57, 379]). Die Sicherheit der Kommunikation hängt somit von der sicheren Implementierung des „APS Transport Key“ Befehls ab. Die Übertragung des Network Keys kann hierbei auf mehrere Arten verschlüsselt werden. Beim Erstbeitritt in ein ZigBee-Netzwerk besteht die Möglichkeit der Verwendung des Global Trust Center Link Keys, welcher einen bekannten Standardwert besitzt (vgl. Kapitel 5.1.3.5). Eine weitere Möglichkeit stellt die Verwendung eines vorinstallierten oder von einem Install Code abgeleiteten Unique Trust Center Link Keys dar. War das Gerät bereits im ZigBee-Netzwerk und der Network Key wird aktualisiert, so wird der alte Network Key oder der Key-Transport Key zur Verschlüsselung der Übertragung verwendet. Der gewählte Schlüssel hängt davon ab, ob die Aktualisierung als Broadcast oder Unicast gesendet wird (vgl. [57, 450], Kapitel 5.1.3.8).

5.1.3.7 Lebensdauer der Sicherheitsschlüssel

Je länger ein Schlüssel genutzt wird desto höher ist die Chance, dass dieser kompromittiert wird. Hierbei ist es die Aufgabe des TC festzulegen, wie lange der Network

Key oder die Link Keys valide sind. Im Falle der Link Keys gibt es keine definierte Dauer wie lange diese maximal gültig sind. Die Spezifikation empfiehlt, die Link Keys periodisch zu aktualisieren. Konkrete Vorgaben sind jedoch nicht vorhanden (vgl. [57, 449]). In Bezug auf die Lebensdauer des Network Keys gibt es jedoch genaue Vorgaben. Das TC soll hier in regelmäßigen Abständen einen neuen Network Key verteilen. Hierfür gibt es zwei zentrale Gründe. Zum einen hat der NWK Frame Counter als Limit den Wert *0xFFFFFFFF*. Wird dieser erreicht, ist das Gerät nicht mehr in der Lage verschlüsselte Nachrichten zu versenden. Durch ein Update des Network Keys wird jedoch der NWK Frame Counter von allen Geräten im Network zurückgesetzt. Zum anderen, wie Eingangs erwähnt, reduziert das regelmäßige Wechseln von Keys die Kompromittierung dieser. Als Zeitpunkt für den Wechsel des Network Keys gibt es mehrere Ansätze. Erster Ansatz wäre, den Network Key zu aktualisieren, wenn das TC eine Nachricht entdeckt, deren Frame Counter größer als *0x80000000* ist. Generell sollte der Network Key mindestens einmal im Jahr aktualisiert werden. Es ist jedoch nicht empfohlen den Network Key häufiger als alle 30 Tage zu ändern. Ausnahmen bilden Anforderungen der Applikation oder des Profils. Besitzt das TC keine Real-Time Clock (RTC) oder eine andere Möglichkeit die Zeit zu bestimmen, so sollte der Network Key ab dem Überschreiten eines Frame Counters von *0x40000000* geändert werden (vgl. [57, 450]). Network Keys werden nur in Centralized Security ZigBee-Netzwerken durchgeführt (vgl. [57, 451])

5.1.3.8 Aktualisierung des Network Key

Für die Aktualisierung des Network Key ist das TC verantwortlich. Hierbei kann der Network Key mittels Broadcast oder Unicast aktualisiert werden. Broadcast Network Key Updates stellen hierbei einen einfachen und simplen Weg dar. Hierbei wird der neue Network Key mit dem alten Network Key verschlüsselt. Einen sicheren Wege hingegen stellt das Unicast Network Key Update Verfahren dar. Der Network Key wird hierbei mit dem Unique Trust Center Link Key verschlüsselt, der jeweils exklusiv für ein Gerät und dem TC existiert und eine legetime Verbindung darstellt. Somit ist nur das jeweilige Gerät in der Lage, die Update Nachricht zu entschlüsseln. Der Befehl zum Wechsel auf den neuen Network Key, der „Key Switch“-Befehl, wird in jedem Fall als Broadcast ausgesandt. Geräte wechseln implizit zum neuen Network Key, wenn sie feststellen, dass ein anderes Gerät bereits den neuen Key verwendet. Hierdurch wird sichergestellt, dass auch Geräte, die den Broadcast zum Wechsel auf den neuen Network Key nicht erhalten haben, den neuen Network Key entsprechend verwenden. Geräte können Nachrichten, die mit dem alten Network Key verschlüsselt

worden sind, empfangen. Antworten erfolgen hierbei jedoch immer mit dem neuen Network Key. Um das verwenden eines alten Network Keys zu unterbinden, muss das TC den „Key Switch“-Befehl entsprechend doppelt ausführen (vgl. [57, 450 f.]).

5.1.3.9 End Device Aging Mechanismus

Die ZigBee 3.0 Spezifikation beschreibt einen „End Device Aging“ Mechanismus (vgl. [57, 363]). Dieser Mechanismus ist dafür verantwortlich, ZigBee End Devices nach einer definierten Dauer aus dem ZigBee-Netzwerk zu entfernen. Die Dauer wird hierbei vom ZigBee Router oder ZigBee Coordinator, je nachdem wer das Elternteil des ZigBee End Devices ist, festgelegt. Das Parentgerät hat hierfür eine Tabelle, in der die ZigBee End Devices und der jeweilige Timeout-Wert eingetragen wird. Der Standardwert für den Timeout wird im „nwkEndDeviceTimeoutDefault“ definiert. Dieser kann von Hersteller zu Hersteller unterschiedlich sein. Erreicht der Timeout-Wert für ein Gerät den Wert 0, so wird er aus der Tabelle gelöscht und muss dem Netzwerk erneut beitreten. Damit dies nicht geschieht, muss das ZigBee End Devices periodisch eine „Keep-Alive“ Nachricht schicken, umd den Timeout-Wert zurückzusetzen.

5.1.4 Schwachstellen / Bestimmung der Angriffsvektoren

Die vorausgegangenen Ausführungen zu den Sicherheitsmaßnahmen werden in diesem Kapitel genutzt, um einige der theoretisch möglichen Angriffsvektoren der ZigBee Funkkommunikation näher zu betrachten und zu erschließen.

Hierzu gab es bereits schon einige Forschungsarbeiten und Bücher, die sich mit der Sicherheit von ZigBee und möglichen Angriffen auf das ZigBee Netzwerk und deren Geräten beschäftigt haben (vgl. [66], [67], [68], [69]). Hierbei lassen sich erfolgreich durchgeführte Angriffe zumeist auf die folgenden Angriffsvektoren zurückführen:

- Key Sniffing
- Replay Angriff
- Insecure Rejoin
- ZigBee Light Link

In den nachfolgenden Kapiteln werden die zuvor aufgeführten Angriffsvektoren näher betrachtet. Hierbei bilden diese die Basis für etwaige aktive Angriffsversuche im Kapitel 5.3.

5.1.4.1 Key Sniffing

Damit ein ZigBee End Device mit anderen Geräten im ZigBee-Netzwerk kommunizieren kann, benötigt es einen Network Key. Den Network Key erhält das Gerät entsprechend innerhalb der Network Join Prozedur. Wie bereits in Kapitel 5.1.3.6 angesprochen, kann der Network Key beim Netzwerkbeitritt mit dem Global Trust Center Link Key oder einem Unique Trust Center Link Key verschlüsselt übertragen werden. Bei diesem Angriff wird der Umstand genutzt, dass viele Geräte den Global Trust Center Link Key zur sicheren Schlüsselübertragung einsetzen.

Ablauf des Angriffs

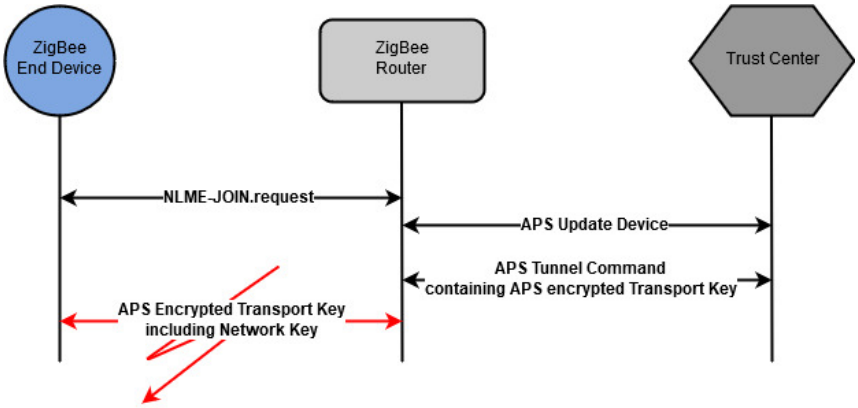

Abbildung 5.12: ZigBee Übertragung des Network Key beim Network Join (eigene Darstellung, in Anlehung an [57, 429] sowie [57, 435])

Abbildung 5.12 stellt den Zeitpunkt der Übertragung des Network Keys beim Beitritt des ZigBee-Netzwerks dar. Der Angreifer muss in der Lage sein, die Kommunikation innerhalb des ZigBee-Netzwerkes mitzulesen, um den Moment der ersten Verbindung eines Gerätes abzufangen. Grundsätzlich funktioniert der Angriff, wie bereits erwähnt, nur dann, wenn der Global Trust Center Link Key zur Absicherung der Schlüsselübertragung genutzt wird. Hat der Angreifer die Übertragung erfolgreich mitgelesen, so ist er nun in Besitz des mit dem Global Trust Center Link Key verschlüsselten Network Keys. Der Global Trust Center Link Key hat, wie in Kapitel 5.1.3.5 beschreiben, den Standardwert *5A 69 67 42 65 65 41 6C 6C 69 61 6E 63 65 30 39*. Der Network Key kann mit diesem und der entsprechenden Nonce unter Verwendung des AES-CCM* Algorithmus entsprechend entschlüsselt werden (vgl. Kapitel 5.1.3.4). Diese Schwachstelle ist auch in der Spezifikation entsprechend beschrieben und als Risiko akzeptiert (vgl. [57, 380]). Sobald der Angreifer den

Network Key entschlüsselt hat, kann jegliche NWK und bestimmte APL Kommunikation entschlüsselt werden und somit die Integrität des Netzwerkes geschädigt werden.

5.1.4.2 Replay Angriff

Wie bereits allgemein in Kapitel 2.1.3.1 angesprochen, versucht ein Angreifer bei einem Replay Angriff bereits gesendete Frames erneut an Geräte im Netzwerk zu schicken. Ziel hierbei ist es, Befehle und Aktionen am Gerät erneut durchzuführen, ohne Zugriff auf jegliches Schlüsselmaterial wie Network Key oder Link Key zu besitzen. Die ZigBee Spezifikation setzt als Schutz gegen diesen Angriff den sogenannten Frame Counter ein, der Bestandteil des Auxiliary Headers ist. Dieser wurde bereits in Kapitel 5.1.3.3 näher beschrieben.

Ablauf des Angriffs

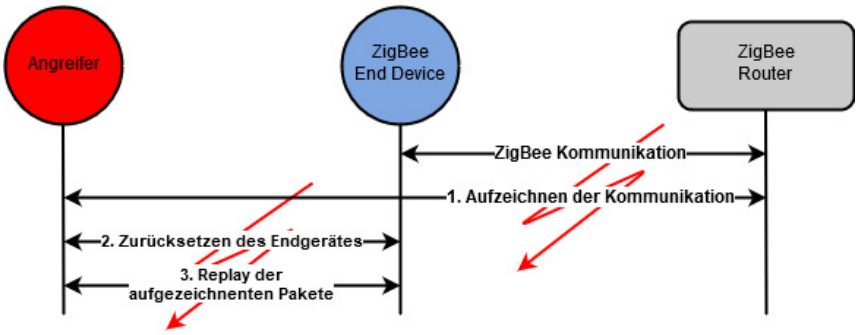

Abbildung 5.13: ZigBee Ablauf eines Replay Angriffs (eigene Darstellung)

Geräte, welche eine ZigBee Version vor ZigBee 3.0 verwenden, speichern den Frame Counter in der Regel nicht persistent ab, da es in der Spezifikation keine Vorgaben gab (vgl. Kapitel 5.1.3.5). Dies kann durch Angreifer ausgenutzt werden und die Möglichkeit eröffnen, integritäts- und authentizitätsgesicherte Frames erneut zu senden. Voraussetzung hierfür ist das Löschen des Frame Counters. Erreicht werden kann dies durch das Zurücksetzen oder Neustarten des Gerätes dem ein Rejoin zum ZigBee-Netzwerk folgt. Um einen Neustart des Gerätes aus der Ferne auszulösen, kann man sich beispielsweise einem „Energy Depletion“ Angriff bedienen (vgl. [69]). Hierbei werden mit einem zufälligen Schlüssel verschlüsselte Nachrichten an das Gerät gesendet. Das Zielgerät kann diese Nachrichten nicht entschlüsseln und verarbeiten. Handelt es sich bei dem Gerät um ein batteriebetriebenes Gerät, wird durch den Energieverbrauch der versuchten Verarbeitung und Entschlüsselung der

Nachrichten die Batterie schneller aufgebraucht. Früher oder später muss die Batterie ersetzt werden und das Gerät wird neu gestartet. Der Frame Counter wird neu initialisiert und zurückgesetzt. Diese Methode funktioniert, da der Frame Counter im Auxiliary Header nicht verschlüsselt wird. Der Angreifer kann nun eine Nachricht mit höherem Frame Counter versenden. Das anzugreifende System wird diese Nachricht verarbeiten, da ihr Frame Counter entsprechend niedriger ist. Die Funktionalität eines solchen Angriffs kann weitreichende Folgen haben. Als Beispiel wären hier ZigBee Türschlösser aber auch Alarmanlagen zu nennen, welche mittels einem Replay Angriffs, ohne Wissen des Eigentümers, entsperrt beziehungsweise deaktiviert werden können. Geräte mit einer ZigBee Version vor ZigBee 3.0 sind anfällig für diesen Angriff. Mit ZigBee 3.0 sollte dieser Angriff nicht mehr möglich sein. Wie in Kapitel (vgl. Kapitel 5.1.3.5) beschrieben, darf der Frame Counter selbst durch eine Rücksetzung auf Werkseinstellungen nicht zurückgesetzt werden. Die einzige Möglichkeit stellt die Auststellung eines neuen Network Keys dar, wir in Kapitel 5.1.3.7 beschrieben. Die aufgezeichnete Kommunikation wäre in diesem Fall aber nicht mehr gültig, das sie mit dem alten Key verschlüsselt wurde. Philips Hue unterstützt bereits seit Februar 2018 die neue ZigBee Spezifikation ZigBee 3.0 (vgl. [70]).

5.1.4.3 Insecure Rejoin

Besitzt eine ZigBee End Device nicht den aktuellen Network Key, beispielsweise wenn es den Tausch des Network Keys verpasst, so kann es mittels dem „Insecure Rejoin“, welcher auch als Trust Center Rejoin bezeichnet wird, dem Netzwerk erneut beitreten (vgl. [57, 371]). Ob ein Insecure Rejoin möglich ist, hängt von jeweils vom Hersteller der Geräte ab. Nach der aktuellen Spezifikation sollte es standardmäßig deaktiviert sein.

Ablauf des Insecure Rejoin

Abbildung 5.14 zeigt den Ablauf des Insecure Rejoin. Das ZigBee End Device, hier bezeichnet als der Joiner, sendete einen unverschlüsselten NWK Rejoin Request an den nächsten ZigBee Router. Dieses fügt das Gerät vorübergehend dem Netzwerk hinzu. Der Joiner ist hierbei aber noch nicht autorisiert. Im weiteren Verlauf sendet der ZigBee Router dann ein Update-Device Command and das TC. Das TC prüft nun, ob das Gerät akzeptiert werden kann. Basis hierfür ist die Network Information Base des TC. Ist beziehungsweise war der Joiner hier bereits eingetragen, so akzeptiert das TC den Rejoin des Gerätes und sendet einen Transport Key an den ZigBee

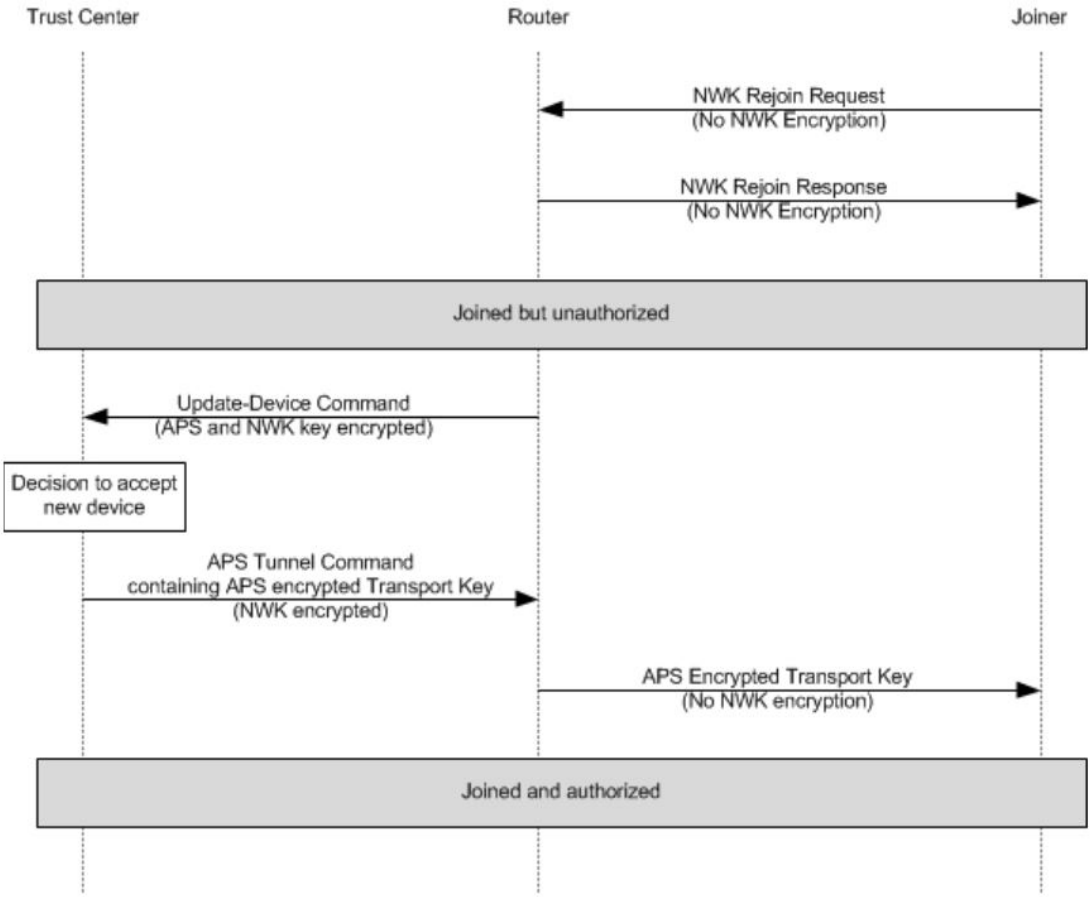

Abbildung 5.14: ZigBee Insecure Rejoin Prozedur (vgl. [57, 435])

Router. Dieser wiederum sendet den Network Key mittels APS Verschlüsselung an das ZigBee End Device. Der Network Key ist hierbei, wie auch beim initialen Schlüsselaustausch, mit dem Default Trust Center Link Key verschlüsselt. Nach Erhalt des Network Keys muss der Joiner eine valide NWK verschlüsselte Nachricht an den ZigBee Router senden. Kann diese Nachricht erfolgreich bearbeitet werden, so ist das Gerät im Status „Joined and authorized“. Der Prozess muss hierbei innerhalb der in *apsSecurityTimeOutPeriod* definierten Millisekunden ablaufen. Sollte dies nicht der Fall sein, so wird das Gerät wieder aus dem ZigBee-Netzwerk entfernt und muss einen neuen Rejoin versuchen (vgl. [57, 435 f.]).

Möglicher Angriff

Kann ein Angreifer die Netzwerkverbindung eines ZigBee End Devices gezielt stören, beispielsweise durch Jamming, so wird es nach einer bestimmten Zeit aus dem ZigBee-Netzwerk entfernt. Mit ZigBee 3.0 wurde, wie in Kapitel 5.1.3.9 beschrie-

ben, der „End Device Aging“ Mechanismus eingeführt. Nachdem sich das Gerät nicht mehr beim Parentgerät meldet, wird es nach dem Timeout aus dem Netzwerk entfernt. Will das Gerät dem Netzwerk wieder beitreten, so kann die Insecure Rejoin Kommunikation durch einen Angreifer mitgelesen werden. Wie beim initialen Schlüsselaustausch wird der Network Key mit dem Global Trust Center Key verschlüsselt übertragen. Dies stellt dieselbe Schwachstelle dar, wie in Kapitel 5.1.4.1 in Bezug auf das Key Sniffing bei der initialen Schlüsselübertragung dargestellt.

5.1.4.4 Hijacking

Der nachfolgende Angriff kann auf Geräten durchgeführt werden, welche das ZLL Applikationsprofil nutzen und wurde in einer Arbeit von Tobias Zillner vorgestellt (vgl. [66], [71]). Als Grundlage für den Angriff wird jedoch zunächst ein Überblick über das ZLL Applikationsprofil gegeben.

Das ZLL Applikationsprofil

Das ZLL Applikationsprofil ist das am häufigsten verwendete Applikationsprofil, das von ZigBee Geräten verwendet wird die mit Beleuchtung zu tun haben. Als Beispiel kann hier Philips Hue genannt werden.

Touchlink Commissioning

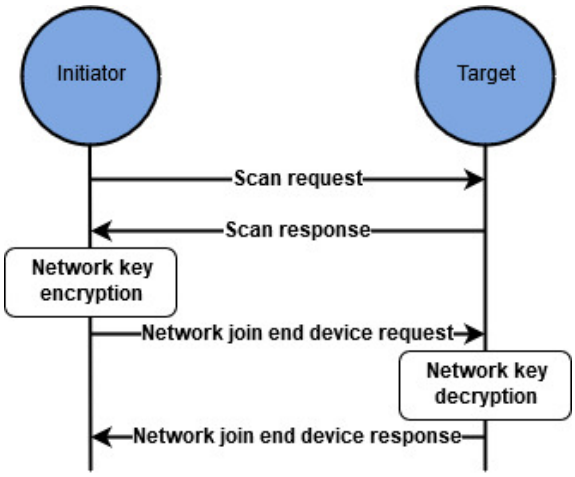

Abbildung 5.15: ZigBee Touchlink Commissioning (eigene Darstellung, in Anlehnung an [72, 3])

Geräte, die unter dem ZLL Applikationsprofil arbeiten, nutzen häufig das Touchlink Commissioning, um dem ZigBee-Netzwerke beizutreten. Diese Art des Beitritts ist

spezifisch für Geräte, die unter dem ZLL Applikationsprofil arbeiten. Die Geräte können hierbei dem Netzwerk über einen Controller beitreten, ohne dass der Besitzer des Geräts mit dem beizutretenden Gerät in irgendeiner Weise agieren muss. Als Beispiel für einen Controller kann die Philips Hue Bridge genannt werden. Abbildung 5.15 stellt den Ablauf des Touchlink Commissioning grafisch dar. Hierbei sendet der Initiator, in diesem Fall der Controller, einen „Scan request“ Befehl an das Target, also das Zielgerät. Dieses muss entsprechend das Touchlink Commissioning unterstützen. Ist das Gerät bereit dem ZigBee-Netzwerk beizutreten, so schickt es einen entsprechenden „Scan response“ als Antwort an den Initiator. Dieser verschlüsselt den Network Key und schickt diesen mittels einem „Network join end device request“ Befehl an das Zielgerät. Mit dem Network Key ist das Zielsystem entsprechend in der Lage mit dem ZigBee-Netzwerk zu kommunizieren. Als Abschluss des Beitritts antwortet das Zielsystem mit einem „Network join end device response“ an den Initiator, um diesem mitzuteilen, dass der Netzwerkbeitritt erfolgreich war.

ZigBee Light Link Master-Key

Alle ZigBee Geräte, die nach ZLL zertifiziert sind, verfügen über einen vorinstallierten Link Key. Dieser wird als ZigBee Light Link Master-Key bezeichnet. Hersteller von ZigBee Geräten die ZLL unterstützen, erhalten diesen von der ZigBee Allicane. Damit sollen die zertifizierten Geräte sicher einem ZigBee-Netzwerk beitreten, ohne den Default Trust Center Link-Key zu nutzen, der wie bereits in Kapitel 5.1.4.1 angesprochen, eine Schwachstelle aufweist. Der ZigBee Light Link Master-Key ist also entsprechend geheimzuhalten. Im Zeitraum um das Jahr 2015 wurde dieser allerdings für Philips Hue geleakt und in öffentlichen Netzwerken und Internetseiten verbreitet (vgl. [73]). Der ZigBee ZLL Master-Key hat für das Philips Hue System den folgenden Wert: *9F 55 95 F1 02 57 C8 A4 69 CB F4 2B C9 3F EE 31*. Die Sicherheit gilt somit als kompromittiert.

Ablauf des Angriffs

Damit das Hijacking, also die Übernahme eines Gerätes funktioniert, muss dieses in die Ausgangslage gebracht werden, erneut nach verfügbaren Netzwerken zu suchen. Bei Geräten, die das ZLL Applikationsprofil verwenden, kann dies durch einen „Reset to factory new“ Befehl erreicht werden. Der Angreifer muss hierbei nicht im Besitz des Network Keys sein. Abbildung 5.16 zeigt den Ablauf des „Reset to factory new“ Prozesses. Der Prozess startet mit einem „Scan request“ Befehl, welchen

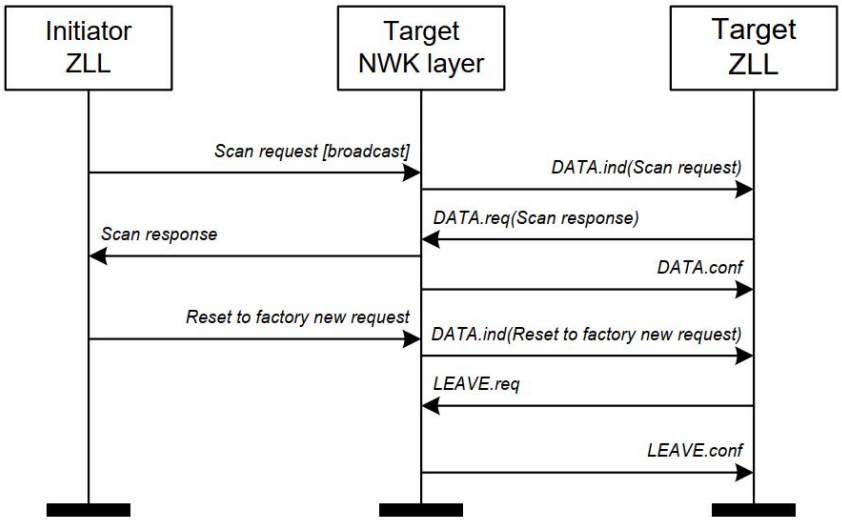

Abbildung 5.16: ZigBee Resetting a device to factory new (vgl. [71, 98])

der Angreifer an das zu übernehmende Gerät schickt. Hierbei wird eine Geräteerkennung und ein erweiterter Scan der verfügbaren Kanäle ausgelöst. Der Initiator schickt anschließend mittels einem inter-PAN Befehl die „Reset to factory new“ Anforderung. Enthält dieser die korrekten Parameter, wie beispielsweise den „command identifier“, akzeptiert das Zielgerät den Befehl und verlässt das Netzwerk und wird entsprechend zurück gesetzt. Der „command identifier“ für eine „Reset to factory new“ Anforderung ist *0x07*. Beim Zurücksetzen wird jegliches Schlüsselmaterial gelöscht. Das Zielgerät geht anschließend in den Modus nach verfügbaren ZigBee-Netzwerken zu suchen, denen es beitreten kann. Genau an diesem Punkt kann dann der Angreifer ansetzen, um das Gerät zu übernehmen. Durch die Nutzung eines bösartigen Controllers, kann der Angreifer dafür sorgen, dass das Ziel-Gerät mittels dem bekannten ZLL Master-Key mit ihm verbunden wird. Hierfür nutzt er den, wie in Abbildung 5.15 dargestellt, Befehl „Network join end device request“. Hierbei kann er einen eigenen Network Key ausstellen, indem er diesen mit dem geleakten ZLL Master-Key verschlüsselt und an das beizutretende Gerät überträgt.

5.2 Bewertung der Informationen / Risikoanalyse

Die zuvor dargestellten Angriffsvektoren der ZigBee Spezifikation wurden anhand des in Kapitel 2.5.2 dargestellten „DREAD“-Modell bewertet. Das Ergebnis ist in Tabelle 5.1 dargestellt. Es ergibt sich entsprechend von selbst, dass die Angriffsvektoren, die als *High* bewertet wurden, ein erhebliches Risiko darstellen und daher schnellstmöglich behoben werden sollten.

	Key Sniffing	Replay Angriff	Insecure Rejoin	Hijacking
***D**amage Potential*	3	1	3	3
***R**eproducibility*	2	1	3	3
***E**xploitability*	2	2	3	3
***A**ffected Users*	2	1	1	2
***D**iscoverability*	3	3	3	3
Ergebnis	**12**	**8**	**13**	**14**
Einstufung	**High**	**Medium**	**High**	**High**

Tabelle 5.1: Bewertung dargestellter Angriffsvektoren von ZigBee

Mehr als die Hälfte der dargestellten Angriffsvektoren weisen eine Einstufung als „High“ auf. Dies liegt vor allem darin, dass die Angriffe *Key Sniffing*, *Insecure Rejoin* sowie *Hijacking* auch in der aktuellen ZigBee Spezifikation für ZigBee 3.0 noch möglich sind. Der Angriff des Hijacking und dessen Umsetzung wurde beispielsweise von Morgner, Mattejat, Benenson, Müller und Armknecht in einem Beitrag zur 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks beschrieben (vgl. Kapitel 5.1.4.4, [72]). Die Vorgehensweise ist hierbei die gleiche wie bei der Übernahme älterer ZigBee-Geräte. Dabei konnte auch hier mit Hilfe des geleakten ZLL Master-Key ein Gerät in das Netzwerk des Angreifers eingefügt werden. Dies bedeutet, dass auch in ZigBee 3.0 der Master-Key nicht verändert wurde. Wie bereits im Angriff zum Insecure Rejoin beschrieben, sollte diese Möglichkeit nach der aktuellen Spezifikation deaktiviert sein (vgl. Kapitel 5.1.4.3). Letztendlich hängt dies aber von der Implementierung des jeweiligen Herstellers der Geräte ab (vgl. [57, 371]). Somit können in der Praxis durchaus Geräte gefunden werden, die trotz aktueller Spezifikation einen Insecure Rejoin erlauben. Der letzte Angriff, welcher als *High* eingestuft werden kann, ist das Key Sniffing (vgl. Kapitel 5.1.4.1). Wie bereits in den Ausführung zu diesem Angriff ausgeführt, ist die Schwachstelle bekannt und entsprechend in der Spezifikation angegeben. Es besteht zwar die Möglichkeit, laut Spezifikation, auf Install Codes zurückzugreifen, diese Möglichkeit ist jedoch nicht verpflichtend (vgl. [57, 380]). Als letztes bleibt der Replay Angriff offen, welcher als *Medium* bewertet wurde. Wie im Kapitel 5.1.4.2 beschrieben, dürfen die Frame Counter ab ZigBee 3.0, selbst bei einem Zurücksetzen auf Werkseinstellung, nicht zurückgesetzt werden. Dies beinhaltet auch einen Neustart des Gerätes. Da ZigBee im Jahre 2016 veröffentlicht wurde, kann zwar davon ausgegangen werden, dass es

noch Geräte mit einer älteren Version gibt. Die Masse der Geräte sollte aber schon durch Updates die neue Spezifikation umsetzen (vgl. Kapitel 5.1.1, [70]).

5.3 Aktive Eindringversuche

Grundlage für dieses Kapitel bildet die im Anhang B dargestellte Hardware und Software. Im Zuge der folgenden Ausführungen soll der Replay Angriff, welcher in Kapitel 5.1.4.2 beschrieben wurde, in einer Philips Hue Umgebung nachgestellt werden.

Als Betriebssystem wird die im Anhang beschriebene Distribution AttifyOS genutzt, die auf einer virtuellen Maschine ausgeführt wird. Über USB angebunden ist die APIMote Hardware (vgl. Kapitel B.1.1). Ob diese funktionsfähig ist, muss entsprechend geprüft werden. Hierbei bedient man sich dem „zbid“ Befehl des KillerBee Frameworks. In Listing 5.1 ist der Befehl und dessen Ausgabe dargestellt. Hierbei ist erkennbar, dass die APIMote Hardware über den USB Serial Port Adapter angebunden und angesprochen werden kann.

```
1 /home/oit/tools/newkillerbee/killerbee/tools [git::master *] [oit@ubuntu] [14:59]
  > sudo zbid
3 [sudo] password for oit:
  Dev Product String       Serial Number
5 /dev/ttyUSB0 GoodFET Api-Mote v2
```

Listing 5.1: KillerBee zbid Befehl

Im nächsten Schritt gilt es festzustellen, auf welchem Kanal das ZigBee Netzwerk agiert. Hierzu wird der „zbstumbler“ Befehl genutzt, der in der Lage ist eine Netzwerkerkennung durchzuführen. Das Tool geht hierbei nacheinander die einzelnen Kanäle durch und versendet Beacon Request Frames. Wird ein Netzwerk erkannt, gibt es eine entsprechende Antwort. Wie aus Listing 5.2 erkenntlich, arbeitet das ZigBee Netzwerk auf Kanal 20.

```
1 /home/oit/tools/newkillerbee/killerbee/tools [git::master *] [oit@ubuntu] [15:31]
  > sudo zbstumbler -v
3 zbstumbler: Transmitting and receiving on interface '/dev/ttyUSB0'
  Setting channel to 11.
5 Transmitting beacon request.
  Setting channel to 12.
7 Transmitting beacon request.
  Setting channel to 13.
9 Transmitting beacon request.
  Setting channel to 14.
```

```
Transmitting beacon request.
Setting channel to 15.
Transmitting beacon request.
Setting channel to 16.
Transmitting beacon request.
Setting channel to 17.
Transmitting beacon request.
Setting channel to 18.
Transmitting beacon request.
Setting channel to 19.
Transmitting beacon request.
Setting channel to 20.
Transmitting beacon request.
# DEBUG Clearing overflow
Received frame.
Received frame is not a beacon (FCF=4188).
```

Listing 5.2: KillerBee zbstumbler Befehl

Um einen Replay Angriff, also bereits durchgeführte Aktionen erneut durchzuführen, müssen die Aktionen zunächst aufgezeichnet werden. Aus dem vorherigen Befehl haben wir die Information erhalten, auf welchem Kanal das ZigBee Netzwerk arbeitet. Mit dieser Information kann der „zbdump“ Befehl gestartet werden, der in eine definierte Datei aufzeichnet. In unserem Fall ist dies die Datei „out.dump“ (vgl. Listing 5.3). Zu Testzwecken wurden während der Aufzeichnungszeit einige Philips Hue Geräte an- und ausgeschaltet, um entsprechende Daten zu generieren.

```
/home/oit/tools/newkillerbee/killerbee/tools [git::master *] [oit@ubuntu] [15:40]
> sudo zbdump -c 20 -w out.dump
zbdump: listening on '/dev/ttyUSB0', link-type DLT_IEEE802_15_4, capture size 127
    bytes
# DEBUG Clearing overflow
# DEBUG Clearing overflow
# DEBUG Clearing overflow
^C77 packets captured
```

Listing 5.3: KillerBee zbdump Befehl

Die aufgezeichneten Daten können beispielsweise in Wireshark näher analysiert und gegebenenfalls bearbeitet werden (vgl. Abbildung 5.17). Um den Replay Angriff durchzuführen, müssen die aufgezeichneten Daten noch in das richtige Format gebracht werden. Hierzu wird der „zbconvert“ Befehl genutzt, der das File in ein Daintree Capture File (DCF) umwandelt (vgl. Listing 5.4).

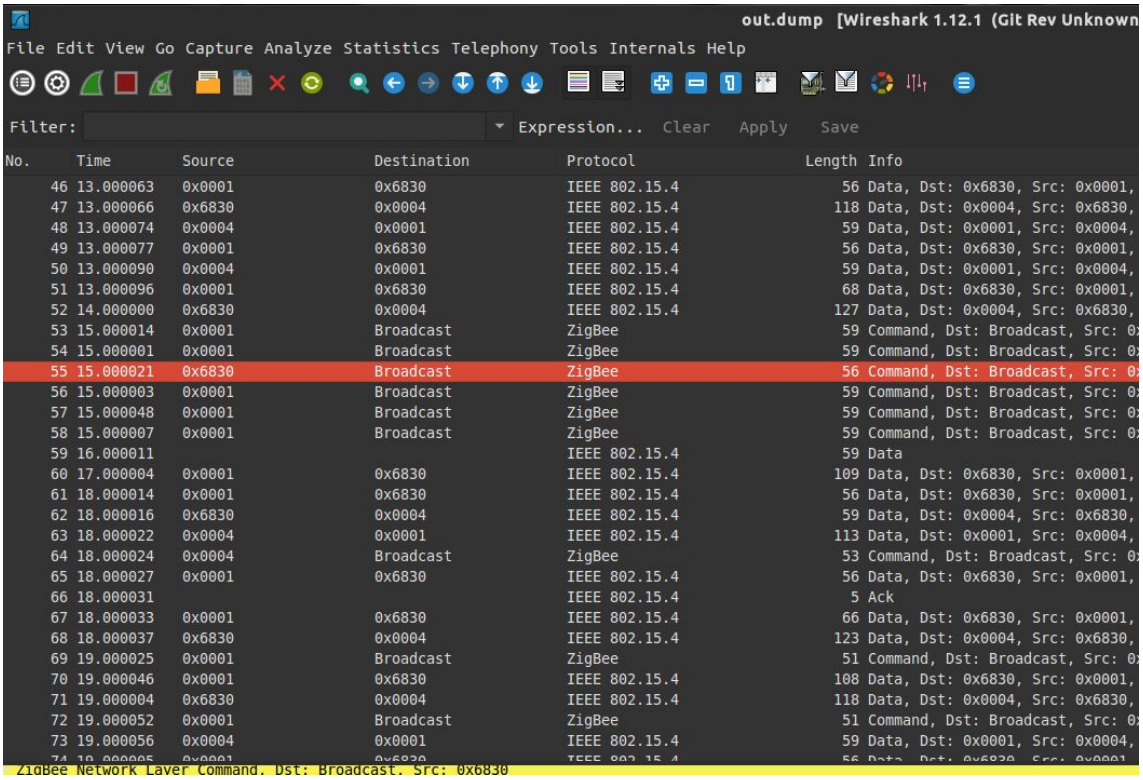

Abbildung 5.17: Wireshark mit geöffneter out.dump (eigene Darstellung)

```
/home/oit/tools/newkillerbee/killerbee/tools [git::master *] [oit@ubuntu] [15:45]
> sudo zbconvert -i out.dump -o out.dcf
Converted 77 packets.
```

Listing 5.4: KillerBee zbconvert Befehl

Mit dem erstellten DCF kann nun der Replay Angriff über den „zbreplay“ Befehl gestartet werden. Benötigte Eingabewerte sind neben dem DCF File auch das Interface und der entsprechende Kanal.

```
/home/oit/tools/newkillerbee/killerbee/tools [git::master *] [oit@ubuntu] [16:19]
> sudo zbreplay -R out.dcf -f 20 -s .1 -i /dev/ttyUSB0
zbreplay: retransmitting frames from 'out.dcf' on interface '/dev/ttyUSB0' with a
    delay of 0.1 seconds.
75 packets transmitted
```

Listing 5.5: KillerBee zbreplay Befehl

Ergebnis:

Alle Befehle wurden ohne Fehlermeldungen durchgeführt. Das gewünschte Ergebnis blieb allerdings aus. So wurden die aufgezeichneten Aktionen der Philips Hue Kom-

ponenten nicht erneut durchgeführt. Dies zeigt, dass die Darstellung des Replay Angriffs aus Kapitel 5.1.4.2 valide und somit ein Angriff nicht mehr möglich ist. Grund hierfür ist, wie bereits erwähnt, die Einführung von ZigBee 3.0. Hierbei wurden die Philips Hue Systeme schon im Februar 2018 auf die neue Version umgestellt, was auch die Einführung der Überprüfung des Frame Counters mit sich brachte (vgl. Kapitel 5.1.3). Ab Philips Hue Firmware-Version *1801260942* ist ZigBee 3.0 aktiv. In Abbildung 5.18 ist die iPhone Variante der Philips Hue App dargestellt. Dort kann festgestellt werden, dass die Firmware-Version schon weitere Updates erfahren hat. Als Hinweis sei hier auch auf die Kanalkonfiguration verwiesen, die auf den Kanal 20 eingestellt ist.

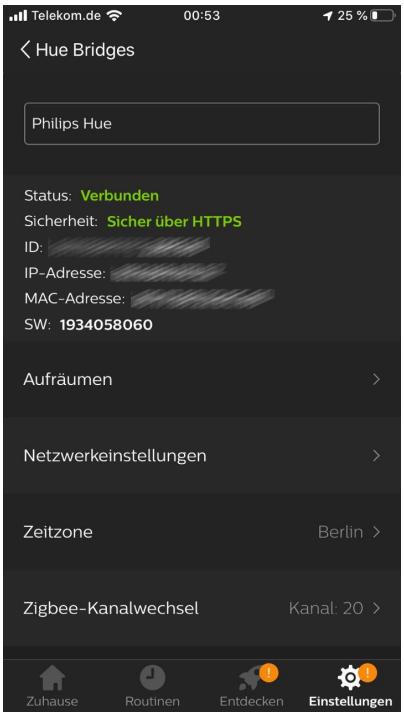

Abbildung 5.18: Philips Hue App für iPhone (eigene Darstellung)

5.4 Abschlussanalyse

In den vorhergehenden Kapiteln wurden die Sicherheitsmaßnahmen, Schwachstellen und Angriffsvektoren der ZigBee Spezifikation dargestellt. Hierbei ist generell festzustellen, dass ZigBee 3.0 viele Schwachstellen aus den älteren ZigBee Spezifikationen teilweise behoben hat. So wurde spezifiziert, dass der Frame Counter persistent zu speichern ist, was das Protokoll widerstandsfähig gegen Replay Angriffe macht. Dies konnte im Zuge des aktiven Eindringversuches positiv nachgewiesen werden. So war ein Replay Angriff auf eine Philips Hue Infrastruktur mit aktueller ZigBee 3.0 Imple-

mentierung nicht möglich. Zusätzliche Änderungen wie eine maximale Lebensdauer von Network Keys oder dem End Device Aging Mechanismus runden die Verbesserungen der neuen Spezifikation ab. Es gibt jedoch auch Verbesserungsvorschläge, die korrigiert werden sollten. So ist es bis jetzt immer noch ein akzeptiertes Risiko, dass der initiale Schlüsselaustausch des Network Keys mit dem Default Global Trust Center Link Key erfolgen kann. Abschließend lässt sich jedoch sagen, dass ein ZigBee Netzwerk, sobald alle Geräte eingerichtet sind, sehr gut abgesichert ist.

6 Schwachstellenanalyse - Z-Wave

Dieses Kapitel behandelt die Schwachstellenanalyse von Z-Wave. Diese beschränkt sich rein auf die Schwachstellenanalyse, welche die Bereiche Informationsbeschaffung, Risikoanalyse und Abschlussanalyse umfasst. Aktive Eindringversuche werden in diesem Fall nicht durchgeführt.

6.1 Informationsbeschaffung

Der Bereich der Informationsbeschaffung ist in vier Teilbereiche unterteilt. Zunächst werden im Rahmen einer Einführung grundlegende Informationen über das zu analysierende Funkprotokoll gegeben. Neben der Darstellung der Funktionsweise werden auch die Sicherheitsmaßnahmen des Funkprotokolls analysiert. Hierbei sei vorweg erwähnt, dass es sich bei Z-Wave um eine proprietäre Technologie handelt. Trotz Z-Wave Public, die Standards zu einigen Protokoll Schichten veröffentlicht hat, sind hier die unteren Schichten nicht öffentlich beschrieben. Z-Wave ist ein geschlossenes Protokoll, welches von der Z-Wave Alliance verwaltet wird. Alle Firmen, die Z-Wave einsetzen wollen, stehen unter einem Non-Disclosure Agreement (NDA), also einem Geheimhaltungsvertrag (vgl. [74, 3 ff.]). Die Ausführung in dieser Schwachstellenanalyse werden sich daher auf die wenigen frei verfügbaren Informationen stützen.

6.1.1 Einführung in das Funkprotokoll

Z-Wave ist, wie bereits in der Einführung zu diesem Kapitel erwähnt, ein proprietäres Protokoll, das von der Z-Wave Alliance, einem globalen Konsortium aus über 600 Firmen, verwaltet wird. Die Produktentwicklung von Z-Wave wurde jedoch durch Geheimhaltungs- und Vertraulichkeitsvereinbarungen erheblich eingeschränkt, sodass es bisher nicht viele Studien zur Z-Wave Technologie gibt. Generell lässt sich dennoch einiges feststellen. Z-Wave nutzt für den Physical und MAC Layer die ITU-T Recommendation G.9959, die öffentlich zugänglich ist. Über den Network Layer hingegen gibt es wenig öffentliche Details (vgl. [75, 1]). Die Reichweite der Übertragung beträgt zwischen 30 und 100 Metern, je nach Umgebungsbedingungen. Hierbei verwendet Z-Wave die Mesh-Topologie mit Unterscheidungen zwischen Master und

Slaves. Maximal unterstützt werden hierbei 232 Geräte innerhalb eines Z-Wave Netzwerks (vgl [76]). Im Jahr 2015 wurde Z-Wave Plus veröffentlicht, welches rückwärtskompatibel zum ursprünglichen Standard ist und beispielsweise Verbesserungen der Protokollfunktionalität einführte. 2016 folgte dann Z-Wave Public, welches zwar einige Standards der Protokollebene veröffentlichte, dennoch aber einige Bereiche unter dem NDA verschlossen hält (vgl. [74, 3 ff.]). Z-Wave arbeitet in den nicht lizenzierten ISM Bändern 868,42 MHz in Europa und 908,42 MHz in Amerika. Es gibt jedoch weltweite noch andere genutzte Frequenzen. Die Datendurchsatzrate richten sich nach der in der ITU-T Recommendation G.9959 angegeben Empfehlung und betragen 9,6 Kbit/s (Rate 1), 40 Kbit/s (Rate 2) und 100 Kbit/s (Rate 3) (vgl. [77, 11]).

6.1.2 Funktionsweise / Grundlagen des Funkprotokolls

6.1.2.1 Z-Wave Protokollstack

Wie bereits in der Einführung in das Funkprotokoll erwähnt, spezifiziert die ITU-T Recommendation G.9959 den Physical und MAC Layer für „Short range narrowband digital radiocommunication transceivers“, zu denen auch Z-Wave zählt (vgl. [77]). Hersteller halten sich entsprechend an die dort definierten Spezifikation für den Physical und MAC Layer, um die Interoperabilität zu gewährleisten. Das Z-Wave Protokoll besteht aus vier Schichten. Abbildung 6.1 zeigt das Z-Wave Referenzmodell und dessen Einordnung in das TCP/IP Referenzmodell. Die Aufgaben der Schichten lassen sich wie folgt beschreiben: Der Physical Layer steuert den Zugriff auf das Hochfrequenzmedium, der MAC Layer übernimmt das Senden und Empfangen von Frames zwischen benachbarten Knoten, der Routing Layer steuert den Nachrichtenfluss im gesamten Netz und der Application Layer führt Befehle aus, die auf den End Devices ausgeführt werden (vgl. [78, 211]).

Nachfolgend werden die einzelnen Schichten und deren Aufgaben dargestellt (vgl. Abbildung 6.1, [77], [78, 211 ff.]):

- **Physical Layer:**
  Der Physical Layer steuert den Zugriff auf das drahtlose Medium. Hierfür verwendet er CSMA/CA zur Kollisionsvermeidung. Wie bereits in Kapitel 6.1.1 erwähnt, nutzt Z-Wave die nicht lizenzierten ISM Bänder, die sich je nach Region unterscheiden. Je nach Datenrate werden unterschiedliche Modulationen und Codierungsverfahren genutzt. So kommt Frequency Shift Keying (FSK)

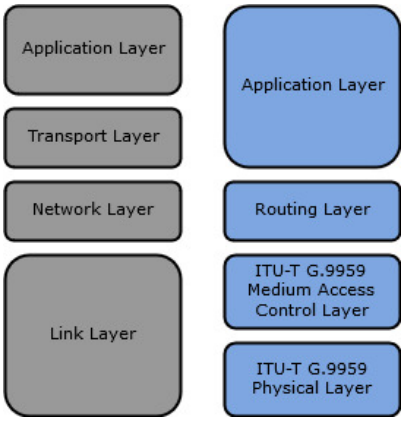

Abbildung 6.1: Z-Wave Referenzmodell und Einordnung in das TCP/IP-Modell (eigene Darstellung, in Anlehnung an [78, 211])

in Verbindung mit Manchester-Kodierung oder Non Return to Zero (NRZ) Codierung sowie Gaussian Frequency Shift Keying (GFSK) mit NRZ Codierung zum Einsatz. In Abbildung 6.2 ist der Frame des Physical Layers, die Physical Protocol Data Unit (PPDU), dargestellt. Diese besteht aus drei Hauptteilen. Zu Beginn kommt der Start Header (SHR), der die Präambel für Symbol- und Bit-Synchronisation sowie den Start Frame Delimiter (SFD) enthält. Gefolgt wird dies von der Payload, der Physical Service Data Unit (PSDU). Zum Schluss kommt der End Header (EHR), welcher nur bei Übertragungen der Data Rate 1, also 9,6 Kbit/s, benötigt wird.

- **MAC Layer:**
  Der MAC Layer wird ebenfalls wie der Physical Layer in der ITU-T Recommendation G.9959 beschrieben. Die Schicht ist verantwortlich für die Steuerung des Datentransfers zwischen zwei Knoten. Dies umfasst neben der Datenvalidierung auch die Benachrichtigung batteriebetriebener Geräte, damit diese während einer eingehenden Übertragung nicht in den Ruhemodus wechseln. Der Frame des MAC Layers, auch als MAC Protocol Data Unit (MPDU) bezeichnet, wird als PSDU in den Frame des Physical Layers eingebunden. Unterschieden werden hierbei drei Arten an MPDU Frames. Singlecast Frames, Acknowledgement Frames und Multicast Frames. Acknowledgement Frames werden als Reaktion auf Singlecast Frames gesendet. Wenn der sendende Knoten kein Acknowledgment vom empfangenden Knoten erhält, finden entsprechende Retransmissionen statt. Multicast Frames werden ohne Rückmeldung an mehrere Zielknoten gesendet. Grundsätzlich folgen alle drei Frames dem selben Aufbau und bestehen aus MAC Header (MHR), MAC Service Data Unit

(MSDU) und MAC Footer (MFR). Hierbei enthält der MHR wichtige Angaben wie beispielsweise die Home ID, Source ID und Destination ID. Abbildung 6.2 stellt den MAC Frame entsprechend dar.

- **Routing Layer:**
  Die Z-Wave Mesh-Topologie wird vom Routing Layer aus verwaltet. Dies stellt sicher, dass die Nachrichten erfolgreich zwischen Control und End Device Nodes weitergeleitet werden. Laut Protokoll können in einem Netzwerk maximal 232 Nodes existieren. Hierbei gibt es nur einen primären Control Node, auch wenn sekundäre Control Nodes vorhanden sein können. Alle Nodes, mit Ausnahme der batteriebetriebenen Geräte, nehmen am Routing teil. Weiterhin legt die Spezifikation fest, dass der maximale „Hop Count“ zwischen Control Node und den End Device Nodes maximal vier Hops betragen darf. Der Routing Layer ist daher auch verantwortlich das Netzwerk zu scannen und eine Routingtabelle im primären Control Node zu pflegen. Hierbei wird die Routingtabelle vom primären Control Node basierend auf Informationen aufgebaut, die von jedem End Device Node über die Nachbarn jedes Nodes empfangen werden. Z-Wave unterstützt eine automatische Topologieerkennung und kann so das Mesh entsprechend heilen, wenn sich die Position eines Nodes geändert hat oder ein Node aus dem Netzwerk entfernt wurde. Dies schließt die Optimierung der Routingtabellen mit ein.

- **Application Layer:**
  Der Großteil der Application Layer Implementierung hängt von der jeweiligen Entwicklung der Hersteller ab. Die Ausführungen sollen daher nur allgemeingültige Informationen bieten. Der Rahmen der Anwendungsschicht besteht grundsätzlich aus dem Header, Informationen zur Command Class und Command Parametern (vgl. Abbildung 6.2). Die Anwendungsschicht ist für die Ausführung der an sie übergebenen Befehle verantwortlich. Befehle werden hierbei in zwei Klassen unterteilt: Command Classes und Device Classes. Jegliche Kommunikation in Z-Wave Netzwerken wird über „Command Classes“ gesteuert. In diesen sind Gruppen an Befehlen und Antworten definiert, die sich auf bestimmte Funktionen eines Gerätes beziehen. Drei grundlegende Befehle die für alle Z-Wave Geräte gelten sind SET, GET und REPORT. SET wird entsprechend zum Ein- oder Ausschalten genutzt. Mit GET kann der Status des Gerätes erfragt werden. REPORT bildet die Antwort auf die Anfrage. Zusätzlich gibt es noch die „Device Classes“. Diese werden in Basic, Generic und Specific Device Classes aufgeteilt. Die Device Class „Basic“ definiert ein

Gerät generell als Controller, Slave oder Routing-Slave. Daher gehört jedes Gerät zu einer Basic Device Class. Die Device Class „Generic“ definiert die Grundfunktionalität, die die Geräte als Controller oder Slave unterstützen. Die letzte Device Class „Specifig“ dient der näheren Spezifizierung der Funktionalitäten der „Generic“ Device Classes (vgl. [79]).

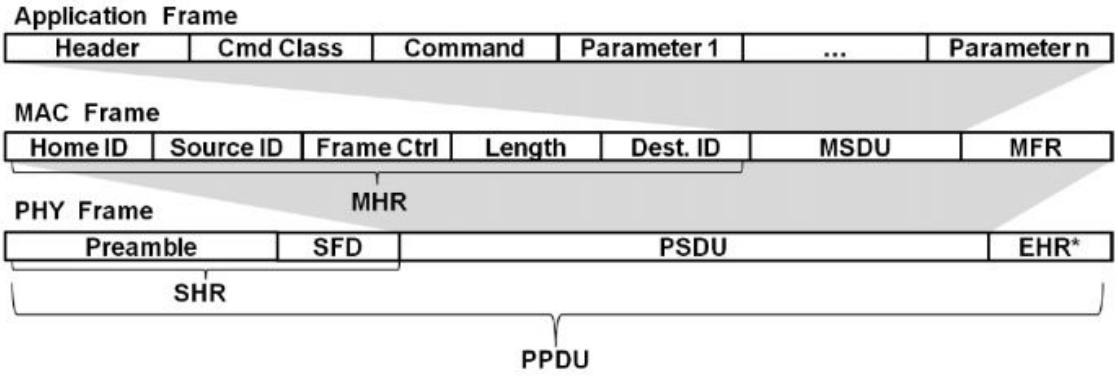

Abbildung 6.2: Z-Wave Frames (vgl. [78, 212])

6.1.2.2 Home ID / Node ID

Jedes Geräte innerhalb des Z-Wave Netzwerks benötigt einen eindeutige Identifikation, um es von anderen Geräten entsprechend unterscheiden zu können. Das Z-Wave Protokoll definiert zwei Identifikatoren zur Organisation des Netzwerks. Dies ist zum einen die Home ID und zum anderen die Node ID (vgl. [80]). Nachfolgend werden die zwei ID´s erläutert. In Abbildung 6.3 sind die Zusammenhänge entsprechend grafisch dargestellt.

Home ID:

Die Home ID ist die gemeinsame Identifikation aller Nodes, die zu einem logischen Z-Wave Netzwerk gehören. Mittels der Home ID können Z-Wave Netzwerke voneinander getrennt werden, sodass Nodes mit unterschiedlichen Home IDs nicht in der Lage sind, miteinander zu kommunizieren. Die Home ID hat hierbei die ID des primären Controllers und besteht aus 32 Bits. Tritt ein neuer Slave Node dem Netzwerk bei, wird im die Home ID des primären Controllers entsprechend zugewiesen. Dies wird als „Inclusion“ bezeichnet. Slave Nodes die keinem Netzwerk angehören, haben die Home ID auf Null gesetzt. Wird ein Slave Node nicht mehr im Netzwerk benötigt beziehungsweise soll in ein anderes Z-Wave Netzwerk eingebunden werden,

so muss es erst aus dem alten Netzwerk entfernt werden, damit die Home ID auf Null gesetzt wird. Dieser Prozess wird als „Exclusion“ bezeichnet (vgl. [79], [80, 4]).

Node ID:

Die Node ID ist die Adresse eines einzelnen Nodes im Netzwerk und vergleichbar mit einer IP Adresse. Der Controller vergibt die Node ID an den Node innerhalb des „Inclusion“-Prozesses. Die Node ID ist hierbei 8 Bit lang. Wie in Kapitel 6.1.1 bei der Darstellung des Application Layers angesprochen, kann jedes Z-Wave Netzwerk 232 Nodes umfassen. Hierbei ist zu beachten, dass der Controller selbst einen Node darstellt und daher die zur Verfügung stehende Zahl entsprechend auf 231 ändert (vgl. [79], [80, 4]).

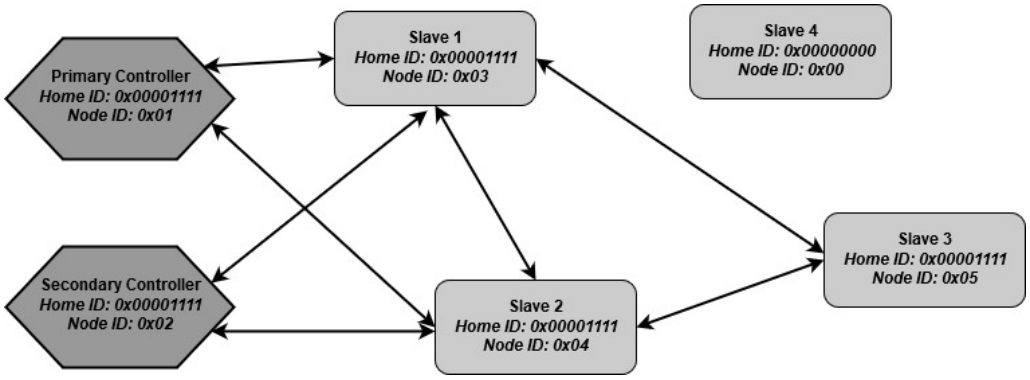

Abbildung 6.3: Z-Wave Mesh-Topologie mit Controller und Slaves (eigene Darstellung, in Anlehnung an [80, 4])

6.1.2.3 Komponenten

Wie in der Einführung, in Kapitel 6.1.1 und 6.1.2.2 angesprochen, wird bei Z-Wave zwischen zwei Gerätetypen unterschieden. So gibt es zum einen die Controller und zum anderen die Slaves (vgl. [80, 4]). Diese sind auch in Abbildung 6.3 dargestellt.

Controller:

Der Controller verfügt über eine vollständige Routingtabelle des Z-Wave Netzwerkes. Durch diese kann er mit allen Nodes innerhalb des Netzwerks kommunizieren. Unterschieden werden hierbei zwei Arten von Controllern: Primary Controller und Secondary Controller. Der Controller, der das Z-Wave Netzwerk erstellt, wird zum Primary Controller. Dieser ist der Master im Netzwerk. Der Primary Controller

ist zudem in der Lage, die Nodes im Netzwerk hinzuzufügen beziehungsweise auszuschließen („Inclusion“ und „Exclusion“). Hierdurch ist der Primary Controller immer auf dem neuesten Stand in Bezug auf die Topologie des Netzwerks. Weitere Aufgabe des Primary Controllers ist die Verwaltung der Zuordnung der Node ID´s. Alle Controller, die über den Primary Controller in das Z-Wave Netzwerk eingefügt werden, werden als Secondary Controller bezeichnet. Diese sind nicht in der Lage, Nodes hinzuzufügen beziehungsweise auszuschließen, erhalten aber Kopien der Routingtabellen vom Primary Controller.

Slave Nodes:

Die Slave Nodes empfangen Befehle und führen diese entsprechend aus. Ein Slave kann als Repeater fungieren, wenn sein Status auf „Listen“ gesetzt ist. Neben normalen Slave Nodes gibt es auch sogenannte Routing Slaves, welche die Fähigkeit besitzen Routen bereitzustellen, über die mit anderen Slaves und Controllern kommuniziert werden kann. Routing Slaves müssen an das Stromnetz angeschlossen sein, um als Repeater zu fungieren. Einen weiteren Typ stellen die Frequently Listening Routing Slaves (FLiRSs) dar, die so konfiguriert sind, auf „Wake Up Beams“ in jedem Aufwachintervall zu lauschen.

6.1.2.4 Routing

Jeder Node ist in der Lage zu bestimmen, welche Nodes sich in direkter Reichweite befinden. Während des „Inclusion“-Prozesses aber auch auf Anfrage kann der Node den Primary Controller über seine Liste der Nachbarnodes informieren. Anhand dieser Informationen kann der Primary Controller eine Routingtabelle aller möglichen Routen des Z-Wave Netzwerks errechnen. Erhält ein Slave eine Nachricht, vergleicht er die enthaltene Node ID mit seiner eigenen. Stimmt diese nicht überein, so leitet er die Nachricht entsprechend weiter. Z-Wave versucht immer den effizientesten Weg zu jedem Node zu bestimmen. Ist dies nicht möglich, sucht der Controller nach einer alternativen Route. Schlägt dies nach dem dritten Mal fehl und der Controller bekommt keine Bestätigung vom Ziel, meldet er entsprechend einen Fehler. Abbildung 6.4 soll die Logik und Wegfindung in Z-Wave Mesh-Netzwerken darstellen. In diesem Beispiel verwendet der Primary Controller den Weg über Slave 1 und Slave 4 um zu Slave 6 zu gelangen. Dies stellt den kürzesten Weg dar. Eine alternativ Route über Slave 2, Slave 3 und Slave 5 benötigt entsprechend mehr Hops um Slave 6 zu erreichen. Anhand der Informationen der Nodes und der errechneten Routingtabelle weiß der Primary Controller immer, wo sich welcher Node befindet und welche

die effizienteste Route ist. Zudem nutzt Z-Wave eine Zweiwege-Kommunikation mit Rückbestätigung. Nur erfolgreiche und bestätige Datagramme gelten als erfolgreich versendet (vgl. [79], [80, 6])).

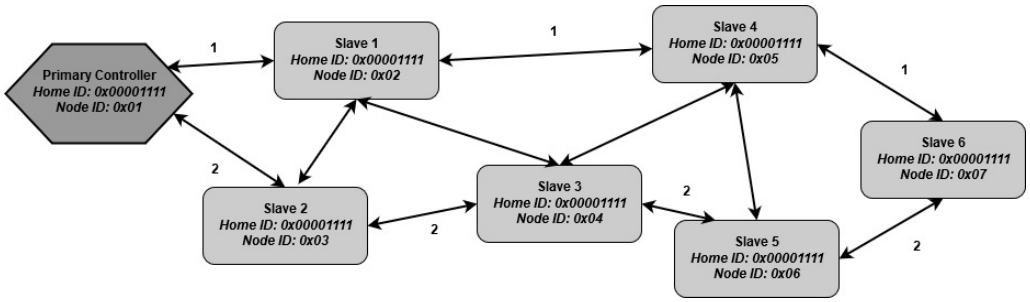

Abbildung 6.4: Z-Wave Routing (eigene Darstellung, in Anlehnung an [79], [80, 6])

6.1.2.5 Beaming

FLiRSs wurden in Kapitel 6.1.2.3 bereits angesprochen. Hierbei handelt es sich um batteriebetriebene Nodes, die eine Batteriesparfunktion aufweisen. Wenn ein Z-Wave Controller oder ein anderer Node im Netzwerk mit einem batteriebetriebenen Gerät, wie beispielsweise einem Türschloss, kommunizieren will, sendet der Controller ein spezielles „Beam“-Signal aus. Das FLiRS-Gerät wechselt zwischen einem Schlafmodus und einem teilweise Wachmodus, in dem es auf das „Beam“-Signal mit einer Rate von 1,0 bis 4,0 Sekunden wartet. Empfängt das FLiRS-Gerät den „Beam“, wacht es sofort vollständig auf und kommuniziert mit dem Controller oder einem anderen Node unter Verwendung von Standardbefehlen des Z-Wave-Protokolls. Hört das Gerät den „Beam“ nicht, geht es für eine weitere Zeitspanne in den Schlafmodus zurück, bis es teilweise wieder aufwacht und auf einen „Beam“ wartet. Durch dieses Verfahren kann eine hohe Akkulaufzeit und gleichzeitig eine Kommunikationslatenz von etwa einer Sekunde erreicht werden (vgl. [80, 6]).

6.1.3 Sicherheitsmaßnahmen

Z-Wave stellt einen eigenen Security Layer zur Verfügung, der auch in Command Classes definiert ist. Hierbei gibt es aktuell zwei Security Command Classes. Die originale, aber bereits abgelöste „S0“ Security Command Class und die „S2“ Security Command Class. Neue Z-Wave Geräte haben alle den „S2“ Security Layer implementiert, der seit April 2017 obligatorisch ist (vgl. [81]). Dieser wird daher in den nachfolgenden Ausführen betrachtet (vgl. [82]).

6.1.3.1 Sicherheitsklassen und Network Keys

„S2“ Security arbeitet, wie Wi-Fi, mit dem Konzept eines Network Keys, den alle Nodes nutzen, um miteinander zu kommunizieren. Hierbei wird das logische Z-Wave Netzwerk jedoch in drei dedizierte Sicherheitsklassen, welche jeweils einen eigenen eindeutigen Network Key verwenden. Die Sicherheitsklassen unterscheiden sich hierbei jedoch nicht nur in den Network Keys, sonder bestimmen auch Authentifizierungs-Regeln für die Aufnahme neuer Nodes. Unterschieden werden die folgenden Sicherheitsklassen (vgl. [82, 7]):

- **S2 Access Control:**
  Bildet die vertrauenswürdigste Klasse, die für die Verwendung bei Zutrittskontrollgeräten, wie Türschlössern, vorgesehen ist.
- **S2 Authenticated:**
  Wird für alle gängigen Haushaltsgeräten, wie Sensoren oder Lichtdimmer, verwendet.
- **S2 Unauthenticated:**
  Bildet die am wenigsten vertrauenswürdige Klasse und ist nur für Controller gedacht, die beispielsweise aufgrund einer eingeschränkten Benutzeroberfläche, nicht in der Lage sind, angeschlossen Nodes im Netzwerk zu authentifizieren.

Ein Node kann während der Aufnahme mehrere Sicherheitsklassen anfordern und entsprechend Zugriff erhalten, akzeptiert aber nur eingehende Befehle in der vertrauenswürdigsten der gewährten Klassen. Dies bedeutet, dass eine Glühbirne, die sowohl Mitglied der Klassen „S2 Authenticated“ als auch „S2 Unauthenticated“ ist, nur Befehle akzeptiert, die mit dem Network Key der „S2 Authenticated“ Sicherheitsklasse verschlüsselt sind.

6.1.3.2 Schlüsselaustausch

Die Übertragung eines Schlüssels auf einen neuen Knoten über ein unsicheres Medium erfordert stets einen sicheren Kanal. Doch dieser muss entsprechend mit einem Schlüssel gesichert werden. Z-Wave Geräte bieten hier keine Schnittstelle, über die lokal ein Schlüssel eingegeben werden kann. Der Diffie-Hellman (DH) Schlüsselaustausch löst dieses Problem. S2-Nodes verwenden einen gemeinsamen Elliptic Curve Diffie-Hellman (ECDH)-Schlüssel, um einen temporären Verbindungsschlüssel abzuleiten. Der Verbindungsschlüssel ermöglicht dann dem Controller, einen oder mehrere Network Keys sicher an einen Node zu übertragen. S0 und S2 verwenden

AES-128-basierte Netzwerkschlüssel, die symmetrisch sind. Das bedeutet, dass alle Knoten in einer bestimmten S2-Sicherheitsklasse Befehle mit demselben Netzwerkschlüssel verschlüsseln und entschlüsseln können (vgl. [82, 8]).

6.1.3.3 Aktiver Schutz vor Angriffen

„S2“ Network Keys sind geheim unter vertrauenswürdigen Geräten und sollten nicht über vorhersehbare Muster im Payload an andere weitergegeben werden. Um dies zu gewährleisten, wird jeder Frame vor der eigentlichen Verschlüsselung durch eine lange Nonce verschlüsselt. Diese 13 Byte lange Nonce, Singlecast Pre-Agreed Nonce (SPAN), wird vor jeder neuen Übertragung automatisch aktualisiert. Hierdurch wird verhindert, dass ein Angreifer einen Befehl aufzeichnet und später einen Replay-Angriff durchführen kann. Mittels der Supervision Command Class kann ein sendender „S2“ Node eine Lieferbestätigung für jeden Befehl anfordern. Durch die oben beschriebene Technik der Nonce, kann jedoch der Angreifer keine gültige Antwort für einen Befehl erstellen. Der „S2“ Node erkennt den versuchten Angriff und kann entsprechend den Benutzer warnen (vgl. [82, 13]).

6.1.4 Schwachstellen / Bestimmung der Angriffsvektoren

In den vorausgegangen Teilen wurde ein Überblick über Z-Wave gegeben und die öffentlich zugänglichen Sicherheitsmaßnahmen dargestellt. Im nachfolgenden Teil werden allgemeine Angriffsvektoren sowie eine bekannte Schwachstelle des Z-Wave Protokolls und deren Angriffsvektor näher erläutert.

Da Z-Wave ein proprietäres Protokoll ist, existieren im Vergleich zu ZigBee aber auch Bluetooth nur wenige Forschungsarbeiten zu dessen Sicherheit (vgl. [83], [84], [85]). Auch die generellen Informationen zur Funktionsweise sind eher spärlich gesät, sodass nur allgemeine Angriffsvektoren definiert werden können, denen drahtlose Funknetze generell ausgesetzt sind (vgl. [78, 213 ff.]). Zu diesen zählen die nachfolgend genannten Angriffe, die die Vertraulichkeit, Verfügbarkeit und Integrität des Netzwerks untergraben.

- **Reconnaissance:**
  Reconnaissance Angriffe, also Aufklärungsangriffe, bestehen aus der passiven Erfassung des Netzwerkverkehrs oder der aktiven Untersuchung eines Zielnetzes, um Informationen über dieses zu erhalten ohne den normalen Betrieb zu stören.

- **DoS:**
  DoS wurde im Zuge dieser Ausarbeitung schon dargestellt (vgl. Kapitel 2.1.3.2). Es handelt sich hierbei um die Zugriffsverhinderung auf drahtlose Systeme, welches zu einer unterschiedlichen Systemverfügbarkeit führt.
- **Packet Injection:**
  Ein Packet Injection Angriff beinhaltet die Übertragung von speziell entwickelten Paketen, um das Verhalten von Netzwerken oder Geräten zu manipulieren.

Auf Grund des Umfangs dieser Ausarbeitung liegt das Augenmerk auf die Darstellung der bekannten Schwachstelle. Es wird daher nur eine der allgemeinen Angriffsvektoren kurz erläutert.

6.1.4.1 Reconnaissance

Ein „Reconnaissance“ Angriff bildet die Grundlage für Folgeangriffe. Zu den gesammelten Informationen des Angriffs zählen neben den verwendeten Protokollen, Gerätetypen unter Umständen auch Verschlüsselungskeys, wenn diese nicht ordnungsgemäß behandelt werden. Ein mit einer Richtantenne ausgestatteter Angreifer kann entsprechende Übertragungen aufzeichnen und für weitere Analysen nutzen. Im Falle von Z-Wave könnte dies beispielsweise der Inhalte des Frame Headers sein, welcher neben der eindeutigen Home ID auch Source ID und Destination ID enthält (vgl. Kapitel 6.1.2.1)

6.1.4.2 Z-Shave

Der Z-Shave Angriff basiert auf der Implementierung des Z-Wave Protokolls, Rückwärtskompatibilität und Interoperabilität mit älteren Versionen von Z-Wave Geräten aufrechtzuerhalten (vgl. [86]). Hierbei müssen Kompromisse in Bezug auf die Systemsicherheit eingegangen werden. Selbst nachdem Silicon Labs, das Unternehmen, das Z-Wave besitzt, es für zertifizierte IdD-Geräte zur Pflicht gemacht hat, den neuesten Sicherheitsstandard S2 zu verwenden, unterstützen Millionen von Smart Devices aus Kompatibilitätsgründen immer noch die ältere unsichere Version des Pairing-Prozesses, das sogenannte S0-Framework. Dieses weißt aber eine kritische Schwachstelle auf, welche 2013 veröffentlicht wurde. So tauscht das S0-Pairing den Network Key aus, indem er ihn mit einem festen Schlüssel verschlüsselt (Wert: *0000000000000000*). Dies bedeutet, dass ein Angreifer die Kommunikation abhören und somit den Netzwerkschlüssel erhalten kann. Jedes Gerät im Netzwerk wäre

hierdurch angreifbar. Das Problem ist bekannt, aber bislang immer noch eine offene Schwachstelle. Genau hier setzt der Z-Shave Angriff an. Auch die Nutzung des S2-Pairing, welches eine Abfangen des Network Keys verhindert (vgl. Kapitel 6.1.3.3), schützt nicht vor diesem Angriff. Nach einer Analyse von Z-Wave entdeckten Sicherheitsforscher von Pen Test Partners aus Großbritannien, dass Geräte, die beide Versionen des Key-Sharing-Mechanismus unterstützen, gezwungen werden könnten, den Pairing-Prozess von S2 auf S0 herunterzustufen (vgl. [83]). Gelingt dies, so ist es, wie bereits beschrieben, möglich den Network Key abzufangen. Grundlage hierfür bildet der „Node Info“ Befehl, welcher unverschlüsselt übertragen wird. Abbildung 6.5 stellt das Key Exchange Verfahren für S0 und S2 dar. Bei beiden Verfahren sind die ersten Steps identisch. Zu Beginn wird der Controller in den „Add“-Modus gebracht. Zur Verbindung muss dann am Node selbst ein Knopf gedrückt werden, der das Senden der „Node Info“ auslöst. Dieses empfängt der Controller und leitet entsprechend das Pairing ein. Beide Verfahren unterscheiden sich grundlegend erst nach dem „Ack“ Befehl. Hierbei ist festzustellen, dass die „Node Info“ unterschiedlich ist.

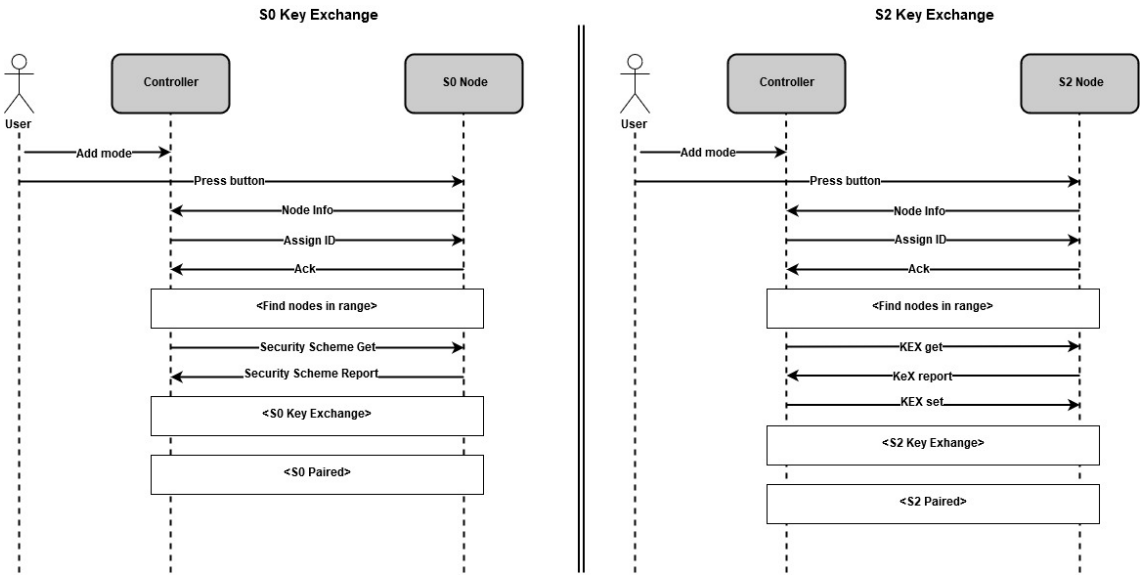

Abbildung 6.5: Z-Wave S0 und S2 Key Exchange (eigene Darstellung, in Anlehnung an [83])

Abbildung 6.6 zeigt beispielhaft die „Node Info“ eines S2 Gerätes. Dies ist erkennbar an der unterstützten Command Class *9F - COMMAND_CLASS_SECURITY_2*. Wie bereits vorhergehend erwähnt, ist der „Node Info“ Befehl völlig unverschlüsselt und unauthentifiziert. Dies ermöglicht es einem Angreifer, die S2 Command Class zu

entfernen. Hieraus resultiert, dass der Controller das Gerät nicht mehr als S2-fähig einstuft und entsprechend ein S0-Pairing durchführt. Hierdurch ist der Angreifer in der Lage, den Schlüsselaustausch abzufangen und entsprechend den Network Key zu extrahieren. Wichtig ist zu erwähnen, dass die verfälschte „Node Info“ die gleiche Home ID enthalten muss, wie der noch nicht verbundene Node (vgl. [83]).

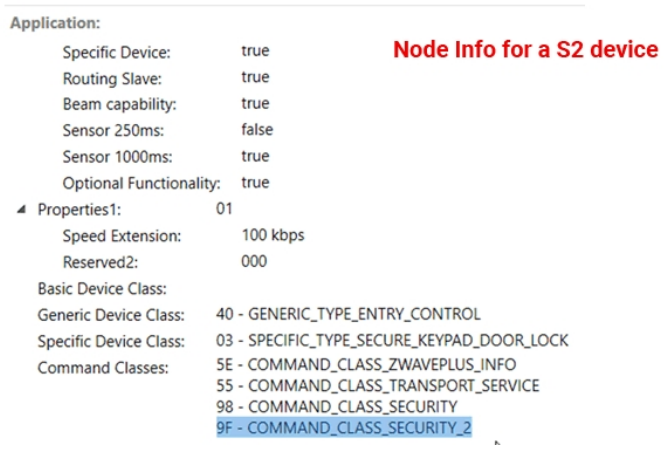

Abbildung 6.6: Z-Wave Node Info eines S2 Gerätes (vgl. [87])

Der Angriff kann mit verschiedenen Methoden durchgeführt werden. Zwei der Möglichkeiten werden nachfolgend dargestellt (vgl. [83]).

S2 Downgrade mittels Timing

Der S2 Downgrade Angriff mittels Timing benötigt, wie der Name schon sagt, den richtigen Zeitpunkt um durchgeführt zu werden. Ein Benutzer würde zunächst den „Add“ Befehl auf dem Controller ausführen, bevor er am Gerät entsprechend das Pairing einleitet. Von Zeit zu Zeit kann diese Reihenfolge durch Leichtsinnigkeit abweichen und der Benutzer aktiviert zuerst das Pairing am Gerät bevor der „Add“ Befehl am Controller gesetzt wird. Dies hat zur Folge, dass vor dem „Add“ Befehl das Gerät seine „Node Info“ in das Netzwerk schickt. Ein Angreifer kann diese nun empfangen, abändern und an den Controller weiterleiten und dafür sorgen, dass das Pairing als S0-Pairing vonstatten geht. In Abbildung 6.7 ist der Prozess entsprechend dargestellt.

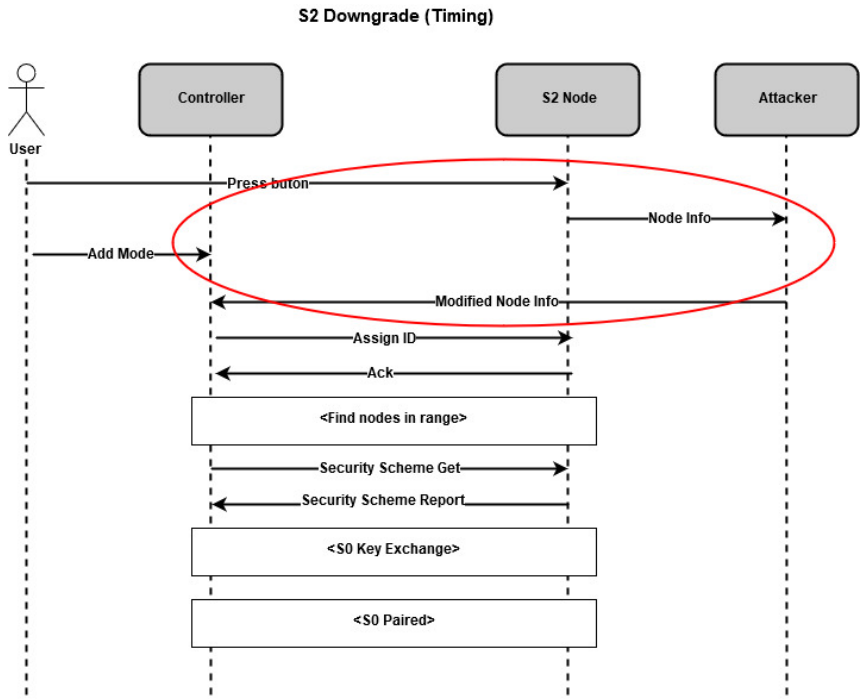

Abbildung 6.7: Z-Wave S2 Downgrade via Timing (eigene Darstellung, in Anlehnung an [83])

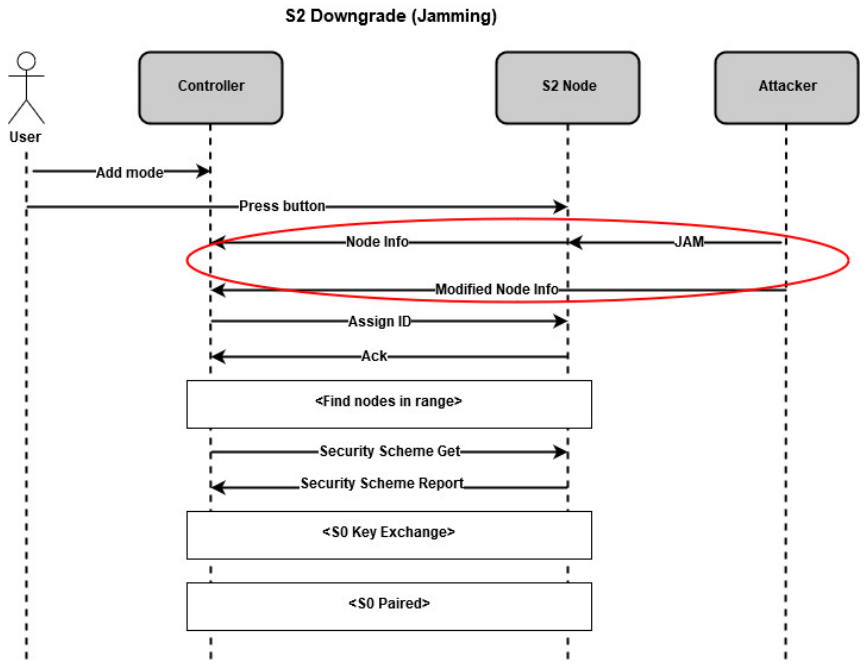

Abbildung 6.8: Z-Wave S2 Downgrade via Jamming (eigene Darstellung, in Anlehnung an [83])

S2 Downgrade mittels Jamming

Die zweite Methoden, den S2 Downgrade Angriff durchzuführen, ist die Jamming-Methode. Dieser ist jedoch sehr komplex in seiner Durchführung. In Abbildung 6.8

ist der Ablauf dargestellt. Der Angreifer muss bei dieser Methode kontinuierlich nach der „Node Info“ des Gerätes suchen. Sobald er dessen Home ID erhält, versucht er mittels Jamming den Rest des Pakets aktiv zu blockieren und zu verhindern, dass der Controller dieses erhält. Gelingt im dies, kann er die modifizierte „Node Info“ an den Controller senden. Wichtig hierbei ist, dass das Jamming alleine nicht zum Erfolg führt. Das Jamming muss in der Lage sein, vor Ende des Paketes zu stoppen und so eine Übertragung der modifizierten „Node Info“ zu gewährleisten. In der Praxis stellt sich dies als sehr schwierig dar, wodurch diese Methode nicht empfohlen ist.

6.2 Bewertung der Informationen / Risikoanalyse

Die zuvor dargestellten Angriffsvektoren von Z-Wave werden anhand des in Kapitel 2.5.2 dargestellten „DREAD“-Modell bewertet. Das Ergebnis ist in Tabelle 7.1 dargestellt.

	Reconnaissance	**Z-Shave**
***D**amage Potential*	2	3
***R**eproducibility*	3	2
***E**xploitability*	2	1
***A**ffected Users*	2	1
***D**iscoverability*	3	2
Ergebnis	**12**	**9**
Einstufung	**High**	**Medium**

Tabelle 6.1: Bewertung dargestellter Angriffsvektoren von Z-Wave

Aufgrund der Tatsache, dass Z-Wave ein proprietäres Protokoll ist, gestaltete sich eine Analyse der Schwachstellen äußerst schwierig, da nicht auf eine allumfassende öffentliche Spezifikation, wie beispielsweise bei Bluetooth oder ZigBee, zurückgegriffen werden konnte. Spärlich sind auch Forschungsarbeiten in diesem Bereich gesät. Dies ist auch der Grund, warum in der Analyse der Schwachstellen nur eine aktuell bekannte Schwachstelle analysiert und dargestellt ist. Diese ist nach der „DREAD“-Kategorisierung als *Medium* zu bewerten, da sie zwar öffentlich beschrieben, aber dennoch komplex in der Nachahmung ist. Als zweites wurde ein allgemeingültiger Angriffsvektor dargestellt, der nicht nur für Z-Wave, sondern auch für

ZigBee oder Bluetooth eine Bedrohung darstellt. Der „Reconnaissance“ Angriff wurde als *High* bewertet, da er als Ausgangspunkt für weitere Angriffe zu sehen ist. Weiterhin bedarf er, zumindest in der Durchführung, kein großes Verständnis, da in den öffentlichen Medien zuhauf Anleitungen existieren, wie diese Art des Angriffs durchzuführen ist.

6.3 Abschlussanalyse

Abschließend lässt sich feststellen, dass aufgrund der Proprietät des Z-Wave Protokolls, ein Angreifer nicht tiefgreifende Einblicke in die Umsetzung des Protokolls erhält, wie dies beispielsweise bei ZigBee oder Bluetooth der Fall ist. Dort finden sich öffentliche Spezifikationen, die das jeweilige Funkprotokoll bis ins Detail darstellen. Dies ist, auch wenn Teile öffentlich zugänglich sind, bei Z-Wave nicht der Fall. Ersichtlich wird dies auch in der geringen Anzahl an Forschungsarbeiten die das Z-Wave Protokoll zum Thema haben. Ähnlich verhält es sich mit öffentlich bekannten Schwachstellen und Vorfällen. Recherchen hierzu enden stets bei der vorgestellten Z-Shave Schwachstelle. Weitere, direkt auf das Z-Wave Protokoll zugeschnittene Schwachstellen, lassen sich nicht finden. Hier greifen eher die Angriffsmöglichkeiten, welchen drahtlosen Funkprotokollen generell ausgesetzt sind. Zusammengefasst kann man Z-Wave eine gute Robustheit gegenüber Angriffen bescheinigen, auch wenn in dem Punkt des S2 Downgrades (vgl. Kapitel 6.1.4) noch nachgebessert werden muss.

7 Schwachstellenanalyse - Bluetooth

Dieses Kapitel behandelt die Schwachstellenanalyse von Bluetooth. Diese beschränkt sich rein auf die Schwachstellenanalyse, welche die Bereiche Informationsbeschaffung, Risikoanalyse und Abschlussanalyse umfasst. Aktive Eindringversuche werden in diesem Fall nicht durchgeführt.

7.1 Informationsbeschaffung

Der Bereich der Informationsbeschaffung ist in vier Teilbereiche unterteilt. Zunächst werden im Rahmen einer Einführung grundlegende Informationen über das zu analysierende Funkprotokoll gegeben. Neben der Darstellung der Funktionsweise werden auch die Sicherheitsmaßnahmen des Funkprotokolls analysiert.

7.1.1 Einführung in das Funkprotokoll

Wer oder was ist eigentlich Bluetooth? Bluetooth ist der Name eines Technologiestandards, der über Funkverbindungen mit kurzer Reichweite verfügt und somit Kabel, welche tragbare und/oder feste elektronische Geräte verbinden, ersetzt. Der Standard definiert eine einheitliche Struktur für eine Vielzahl von Geräten, um mit minimalem Aufwand seitens des Benutzers miteinander zu kommunizieren.

Die wichtigsten Merkmale sind (vgl. [88, 180]):

- Robustheit
- Geringe Komplexität
- Niedrige Leistung
- Standardisierte, kostengünstige Hardware

Die genannten Merkmale prädestinieren Bluetooth vor allem für mobile Endgeräte. Weiterhin bietet diese Technologie für eine Vielzahl von Haushaltsgeräten und mobilen Endgeräten den drahtlosen Zugriff auf Local Area Networks (LANs), Festnetz, das Mobilfunknetz und das Internet. Der Standard hat weltweite Akzeptanz

gefunden, so dass jedes Bluetooth-Gerät überall auf der Welt eine Verbindung zu anderen Bluetooth-Geräten in seiner Nähe herstellen kann, unabhängig von dessen Marke. Somit ist es auch nicht verwunderlich, dass, laut dem aktuellen Bluetooth Market Update von 2019, die jährlichen Auslieferungen von Bluetooth Smart Home Geräten bis 2023 voraussichtlich 1,15 Milliarden erreichen wird (vgl. [89], [90, 38]). Im Gegensatz zu Wi-Fi ist Bluetooth vordringlich auf eine drahtlose, serielle Kommunikation zwischen Geräten über sehr kurze Distanz bis maximal 10 Meter ausgerichtet und zählt aufgrund der geringen Reichweite zu den WPANs. Neuere Geräte weisen aber bereits Reichweiten von bis zu 100 Metern auf. Im Vergleich mit Wi-Fi hat Bluetooth allerdings eine deutlich geringere Datenübertragungsrate (vgl. [50, 926 f.]). Bluetooth basiert auf dem IEEE Standard 802.15.1 und wird von der Bluetooth Special Interest Group (SIG), einem internationalen Konsortium von Unternehmen die an der Entwicklung und Verbreitung der Bluetooth-Technologie interessiert sind, stetig durch neue Spezifikationen verbessert. Letzte Bluetooth Version ist Bluetooth 5.1, welche im Januar 2019 vorgestellt wurde. Hierbei wird Bluetooth in zwei Bereiche unterteilt: Basic Rate (BR) und Low Energy (LE). Beide Systeme beinhalten Geräteerkennung, Verbindungsaufbau und Verbindungsmechanismen. Optional beinhaltet BR die Enhanced Data Rate (EDR) sowie Alternate MAC und Physical Layer Erweiterungen (AMP). BR bietet synchrone und asynchrone Verbindungen mit Datenraten von 721,2 kb/s für BR, 2,1 Mb/s für EDR und für den Hochgeschwindigkeitsbetrieb bis zu 54 Mb/s mit dem 802.11 AMP. Bluetooth Low Energie zielt hierbei vor allem auf Anwendungen mit geringem Stromverbrauch, keinen hohen Datenraten und Anwendungen, die auf keinen kontinuierlichen Datenstrom angewiesen sind. Verzichtet wird hierbei auf hohe Reichweiten (50m statt 100m) und hohen Datendurchsatz (0,27 Mbit/s statt 2,1 Mbit/s) (vgl. [88, 180]). Bluetooth Low Energie findet zum Beispiel Einsatz bei Smartwatches. Zu beachten ist hierbei jedoch, dass BLE nicht rückwärts-kompatibel zu Vorgängerversionen ist. Bluetooth, sowohl BR/EDR als auch LE, arbeiten wie ZigBee im nicht lizenzierten 2,4 GHz ISM Band. Bluetooth unterstützt im Bereich der Netzwerktopologien für LE Point-to-Point (mit Piconet) und Mesh Topologien. Für BR/EDR kann die Point-to-Point (mit Piconet) Topologie zum Einsatz kommen (vgl. [91]). Abbildung 7.1 stellt mögliche Einsatzbereiche von Bluetooth dar.

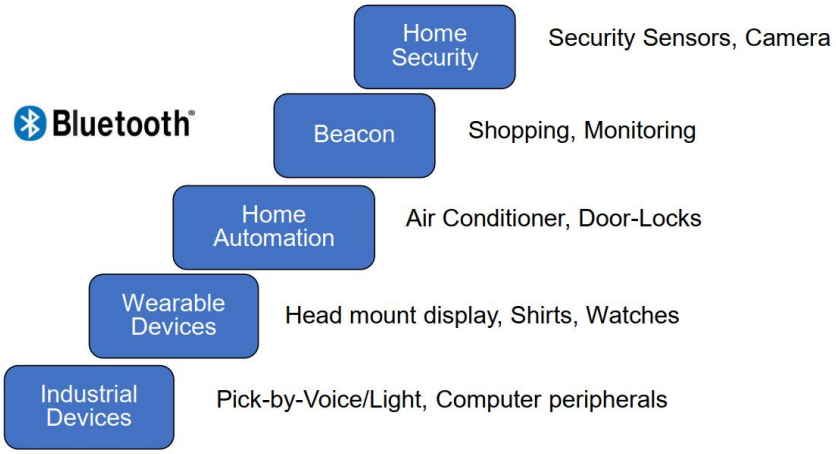

Abbildung 7.1: Bluetooth Nutzungsbeispiele (eigene Darstellung, angelehnt an [92])

7.1.2 Funktionsweise / Grundlagen des Funkprotokolls

7.1.2.1 Bluetooth Protokollstack

Der Bluetooth Protokollstack enthält neben den Bluetooth-spezifischen Protokollen, wie Link Manager Protocol (LMP) oder Logical Link Control and Adaption Protocol (L2CAP), auch allgemeine Protokolle wie Object Exchange Protocol (OBEX), User Datagram Protocol (UDP), TCP oder Wireless Application Protocol (WAP) (vgl. [50, 930]). Im nachfolgenden Teil wird der Bluetooth Protokollstack, wie in vereinfachter Form in Abbildung 7.2 dargestellt, besprochen (vgl. [93]). Der Bereich Bluetooth Radio wurde bereits in Abschnitt 7.1.2.3 erläutert.

- **Baseband:**
  Eine Ebene über dem Bluetooth Radio befindet sich das Baseband, welches auch als Baseband Link Controller bezeichnet wird. Dieses agiert, wie Bluetooth Radio, auf Schicht 1 des OSI-Modells, dem Physical Layer. Das Baseband ist in der Lage, unterschiedliche Vermittlungstechniken für unterschiedliche physische Verbindungsarten einzusetzen. Hierbei unterscheidet man zwischen der synchronen leitungsvermittelten Verbindung (Synchronous Connection Oriented (SCO) Link), der erweiterten synchronen leitungsvermittelten Verbindung (Enhanced SCO (eSCO) Link) und der asynchronen paketvermittelten Verbindung (Asynchronous Connection-Oriented Logical (ACL) Link). Der SCO Link ist eine symmetrische Punkt-zu-Punkt-Verbindung. Hierbei weist der Master dem Slave in regelmäßigen Zeitabständen einen Slot zur Datenübertragung zu. Die Kommunikation wirkt hier wie in einem leitungsvermittelten

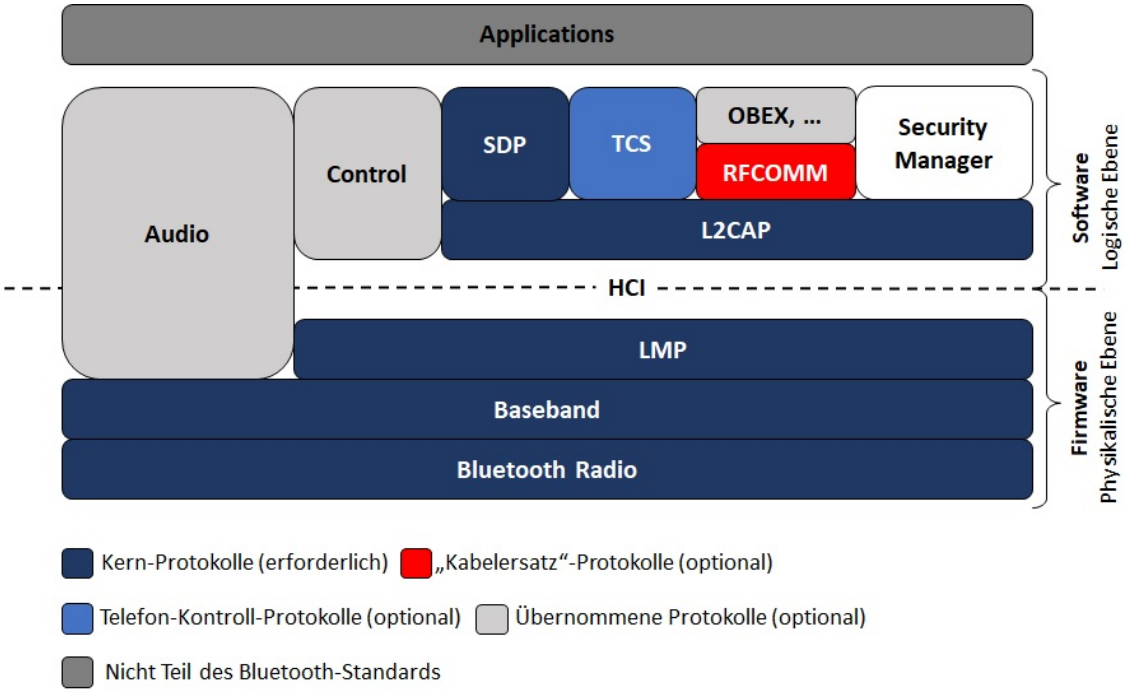

Abbildung 7.2: Bluetooth Protokollstack (eigene Darstellung, angelehnt an [94], [95], [50, 930])

Netz und ist auf die Übertragung von Sprache ausgelegt. Maximal können bis zu drei Datenströme gleichzeitig übertragen werden. eSCO Link verhält sich im Grunde wie der SCO Link. Unterscheidung findet sich hier aber in der zyklischen Redundanzprüfung (Cyclic Redundancy Check (CRC)) sowie einem Zeitfenster unmittelbar nach einem reservierten Slot, in dem Daten nochmals übertragen werden können. Bei der asynchronen paketvermittelten Verbindung kann der Master mit jedem Slave im Piconet slotweise in den Slots, die nicht für den synchronen Datentransport reserviert sind, Datenpakete austauschen.

- **LMP:**
  Das LMP ist auf der Schicht 2 des OSI-Modells, dem Data Link Layer, angesiedelt. Das Protokoll dient dem Aufbau und der Steuerung von Verbindungen zwischen Bluetooth-Geräten und übernimmt auch Sicherheitsfunktionen, wie Authentifikation, Schlüsselaustausch und Verschlüsselung. Weiterhin wird es vom Linkmananger genutzt, der auf jedem Bluetooth-Gerät laufen muss um die Funktionalität an sich zu gewährleisten.

- **L2CAP:**
  Ebenso wie das LMP ist das L2CAP auf der Schicht 2 des OSI-Modells angesiedelt, kommt allerdings nur bei asynchronen Verbindungen zum Einsatz.

L2CAP stellt den darüber liegenden Protokollen zum Beispiel ein Protokoll-Multiplexing zur Verfügung, wozu das Baseband nicht in der Lage ist. Zu den weiteren Aufgaben zählt, größere Datenpakete zu zerlegen bzw. zusammenzusetzen, da die Größe im Baseband auf 341 kB begrenzt ist.

- **Service Discovery Protocol (SDP):**
  Wenn Bluetooth-Geräte auf spezielle Dienste zugreifen bzw. diese finden wollen, so verwenden sie das SDP. Mit dessen Hilfe können Geräteinformationen, sowie Dienste und Leistungsdaten abgefragt werden.

- **Radio Frequency Communication (RFCOMM):**
  RFCOMM wird auch als Kabelersatzprotokoll bezeichnet und emuliert eine serielle Schnittstelle ähnlich zu RS232. Hierüber wird zum Beispiel auch kabelloses Drucken ermöglicht. Aufsetzend auf RFCOMM stehen die bekannten Netzwerk-Protokolle wie Point-to-Point Protocol (PPP), IP, TCP beziehungsweise UDP zur Verfügung.

- **Telephony Control Specification Binary (TCS):**
  Das TCS ist ein bitorientiertes Telefonie-Protokoll und emuliert wie RFCOMM herkömmliche Kabelverbindungen. Hierbei erlaubt es Rufsignalisierung, sowie den Aufbau von Sprach- und Datenkommunikation.

- **OBEX:**
  Das OBEX wurde ursprünglich zur Infrarotkommunikation entwickelt. Mit dessen Hilfe können Standards wie vCal und vCard zum Beispiel Kalendereinträge oder Kontaktdaten austauschen.

7.1.2.2 Netzwerktopologien

Mittels Bluetooth können sich Geräte (auch Ad-hoc) zu Netzen, den Piconetzen zusammenfinden (vgl. [50, 928 f.]). Abbildung 7.3 zeigt ein solches Piconet. Jede Einheit kann gleichzeitig mit bis zu sieben weiteren Einheiten pro Piconet kommunizieren. Der Initiator der ersten Verbindung übernimmt die Masterfunktion (Blau). Alle anderen Teilnehmer agieren als Slaves (Orange) (vgl. Abbildung 7.4). Möchte eine Gerät dem Piconetz beitreten, so sendet es eine Inquery-Nachricht aus. Alle erreichbaren und aktiven Geräte antworten entsprechend auf die Anfrage. Neben den 8 aktiven Geräten (Master und Slave) können noch mehr als 200 weitere Geräte in einem Stromspar-Modus (z.B. Parked-Mode) passiv dem Piconetz angehören (Grau) (vgl. Abbildung 7.5). Diese zählen dann nicht zu der maximalen Anzahl der

8 Teilnehmer pro Netz. Der Parked-Mode ist mit Bluetooth Version 5 als "deprecated“, also veraltet, erklärt worden. Da Version 5 in 2016, Version 5.1 erst in 2019 veröffentlicht wurde, ist aber damit zu rechnen, dass der geparkte Modus noch heute möglich und somit auch anzutreffen ist.

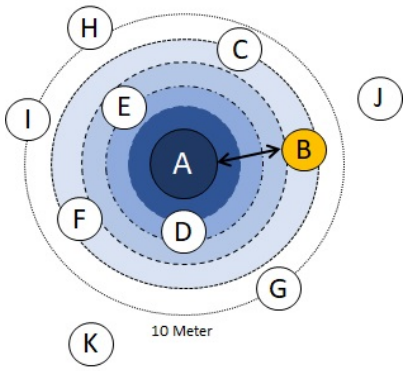

Abbildung 7.3: Bluetooth Piconet mit einer aktiven Verbindung (eigene Darstellung, angelehnt an [96])

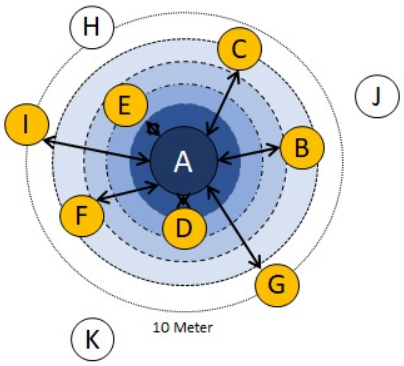

Abbildung 7.4: Bluetooth Piconet mit der maximalen Anzahl an aktiven Verbindungen (eigene Darstellung, angelehnt an [96])

Piconets werden dynamisch beim Betreten und Verlassen der Funkreichweite von Bluetooth-Geräten gebildet. Bluetooth-Geräte können an mehreren Piconets gleichzeitig teilhaben. Es entsteht eine Topologie, die als Scatternet bezeichnet wird (vgl. Abbildung 7.6). Ein Gerät kann in mehreren Piconetzen als Slave fungieren. Die Funktion als Master kann ein Gerät aber nur maximal in einem Netz ausführen. Es ist aber möglich, dass ein Master gleichzeitig als Slave in einem anderen Piconetz agiert. Scatternets können sich zu extrem komplexen Strukturen entwickeln, die ein dichtes Gefüge aus vielen Geräten bilden. Diese Topologie nennt man dann Advanced Scatternets.

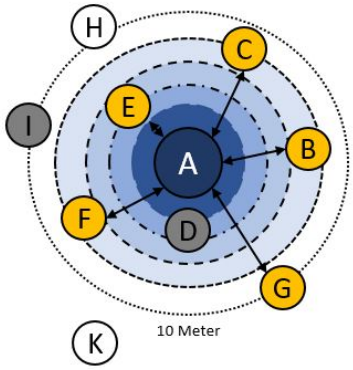

Abbildung 7.5: Bluetooth Piconet mit passiven Slaves (eigene Darstellung, angelehnt an [96])

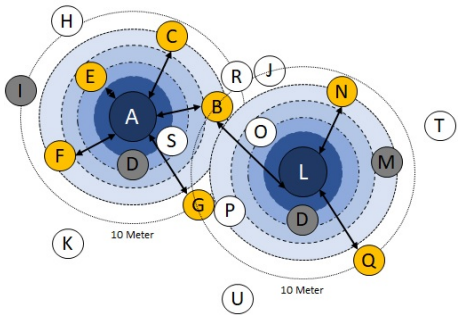

Abbildung 7.6: Bluetooth Scatternet mit aktiven und passiven Slaves (eigene Darstellung, angelehnt an [96])

Ab Bluetooth Version 4.0 kann bei BLE auch ein Mesh-Netz mit many-to-many (m:m) Verbindungen aufgebaut werden. Somit können alle Knoten untereinander eine Verbindung aufbauen (vgl. Abbildung 7.7). Diese Netze finden vor allem Einsatz in der Gebäudeautomation, Heimautomatisierung und dem IoT als Konkurrent zu ZigBee und Z-Wave. Hierbei punktet BLE gegenüber Z-Wave mit der herstellerübergreifenden Interoperabilität. Des weiteren kann Bluetooth Smartphones und Tablets ins Mesh-Netzwerk einbinden, indem diese selbst ein Teil des Mesh-Netzwerks werden. Bei ZigBee und Z-Wave sind für die selbe Funktionalität spezielle Gateways erforderlich (vgl. [97]).

7.1.2.3 Frequenzbereich/-spektrum

Bluetooth verwendet, wie bereits in Kapitel 7.1.1 angesprochen, das lizenzfreie ISM Band, welches zwischen 2,402 GHz und 2,480 GHz liegt. Der Vorteil hierbei ist dessen allgemeine Verfügbarkeit. Bluetooth kann somit ohne nennenswerte geografi-

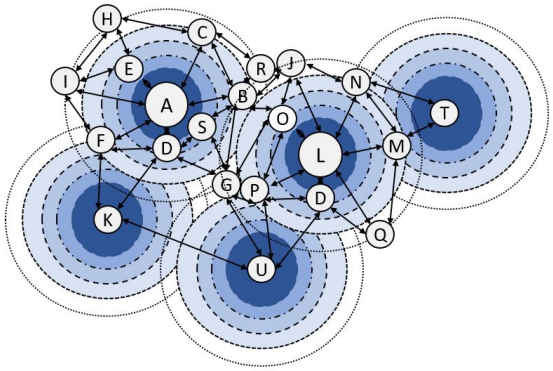

Abbildung 7.7: Bluetooth Mesh (eigene Darstellung, angelehnt an [98])

sche Einschränkungen weltweit betrieben werden. Problematisch am ISM Band ist jedoch die Tatsache, dass neben Bluetooth auch Haushaltsgeräte (z.B. Mikrowellenherde), Wi-Fi und Short Range Devices (SRD) (z.B. Funkmäuse) aber auch ZigBee innerhalb des definierten Frequenzbereiches operieren. Eine gegenseitige Beeinflussung bzw. Störung ist somit nicht ausgeschlossen. Bei Wi-Fi und Bluetooth gibt es hierfür aber spezielle Fehlerkorrekturmaßnahmen, die Fehler durch die gegenseitige Beeinflussung erkennen und entsprechend beheben.

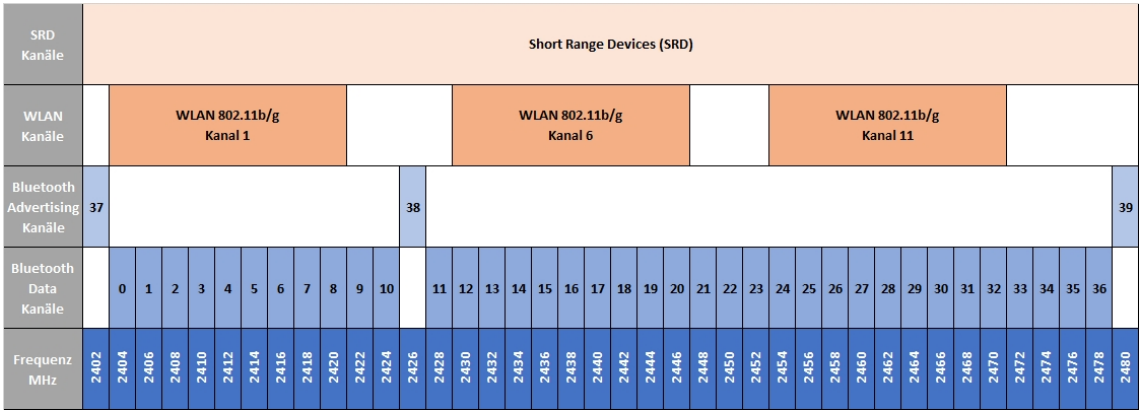

Abbildung 7.8: Bluetooth Frequenzspektrum mit Low Energie-Kanalaufteilung (eigene Darstellung, angelehnt an [93])

Abbildung 7.8 stellt das Frequenzspektrum für den LE Betrieb unter Einbeziehung der Bereiche der SRD und WLAN Kanäle dar. Der Frequenzbereich teilt sich hier in 40 Kanäle mit je 2 MHz Kanalbandbreite auf. Bei BR/EDR gibt es 79 Kanäle mit je 1 MHz Kanalbandbreite. Die Datenkanäle dienen der Übertragung von Daten, nachdem eine Verbindung zwischen den Geräten aufgebaut wurde. Hierbei gibt es

für LE 37 für BR/EDR 79 Kanäle. Die Advertising-Kanäle werden verwendet um in der Nähe befindliche Bluetooth-Geräte zu erkennen und Daten, Informationen von lokalem Interesse oder Werbung verbindungslos als Broadcast anzubieten. Bei der Verbindungsanfrage werden über die Advertising-Kanäle die Verbindungsparameter ausgetauscht. Nach dem Aufbau der Verbindung werden dann die regulären Daten-Kanäle zur Kommunikation verwendet. Es gibt drei feste Frequenzen für den Advertising-Dienst: 2402 MHz (Kanal 37), 2426 MHz (Kanal 38) und 2480 MHz (Kanal 39). Wie zu erkennen ist, liegen die Advertising-Kanäle in unterschiedlichen Teilen des Frequenzspektrums. Diese Maßnahme erhöht die Störsicherheit gegenüber Wi-Fi. Um die Störfestigkeit und Sicherheit zu erhöhen wird Frequency Hopping Spread Spectrum (FHSS) zur Funkübertragung verwendet. Hierbei handelt es sich um eine Bandbreitenspreizung durch Frequenzsprung. Die Trägerfrequenz des Bluetooth-Senders springt hier 1600 mal pro Sekunde zwischen den 37 (LE) bzw. 79 (BR/EDR) Datenkanälen hin und her. Der Kanal ist hierbei in 625 µs lange Intervalle, sogenannte Slots, unterteilt. Die Umschaltreihenfolge wird durch eine Pseudozufallszahlen-Sequenz bestimmt, welche Sender und Empfänger bekannt sein muss. Es sind sechs Typen von Hop-Sequenzen definiert, auf die hier jedoch nicht weiter eingegangen wird. Welche Sequenz letztendlich verwendet wird, ist von mehreren Faktoren, wie z.B. der Hardwareadresse (Bluetooth Device Address, BD_ADDR) des Masters abhängig. Die Übertragung der Daten erfolgt im Time Division Duplex (TDD)-Verfahren bei dem Master und Slave abwechselnd senden. Ab Bluetooth Version 1.2 kann Bluetooth mittels Adaptive Frequency Hopping Spread Spectrum (AFH) erkennen, welche Kanäle von anderen Netzen bereits belegt oder gestört sind. Die Hopping-Sequenz wird dann entsprechend angepasst. Hiervon wurde auch die Bezeichnung Adaptive Frequency Hopping abgeleitet. Abbildung 7.9 stellt die Aufteilung der Slots, die Frequenzsprünge und das abwechselnde Senden von Daten im TDD-Verfahren zwischen Master (Blau) und Slave (Rot) dar. Ebenso ist als „Störer“ der WLAN 802.11b/g Kanal 6 eingetragen.

7.1.2.4 Bluetooth Pakete

Nachfolgend sollen die verschiedenen Typen der Bluetooth Pakete kurz dargestellt werden. Dies dient, in Kombination mit dem Kapitel über die Bluetooth Device Address (vgl. Kapitel 7.1.3.3), dazu, Zusammenhänge mit möglichen Schwachstellen nachzuvollziehen.

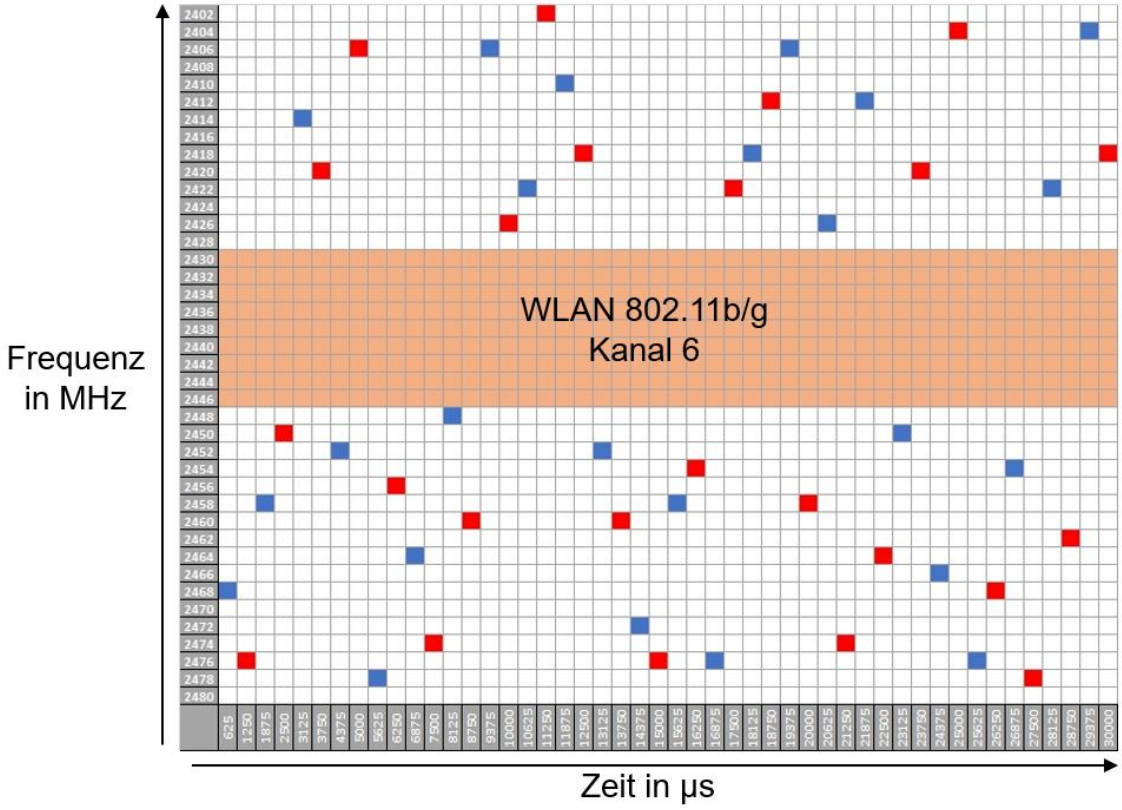

Abbildung 7.9: Bluetooth Funkübertragung mittels FHSS (eigene Darstellung, angelehnt an [99])

Bluetooth BR Packet Format

Das allgemeine Paketformat der BR ist in Abbildung 7.10 dargestellt. Oftmals wird diese Darstellung in der Literatur auch als das „allgemeine Format“ eines Bluetooth Pakets beschrieben (vgl. [50, 927]). Daher wird dessen Aufbau nachfolgend im Detail betrachtet (vgl. [88, 427 ff.], [93]). Die Paket Formate für EDR und LE werden nur kurz beschrieben.

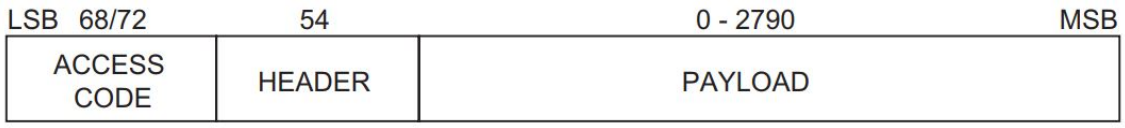

Abbildung 7.10: Bluetooth Basic Rate Packet Format (vgl. [88, 427])

- **Access Code:**
  Jedes Bluetooth Paket beginnt mit dem Access Code. Wie in Abbildung 7.10

ersichtlich, kann dieser 68-Bit oder 72-Bit lang sein. Dies hängt davon ab, ob nach dem Access Code ein Paket Header folgt oder nicht. Folgt ein Paket Header, so ist der Access Code 72-Bit lang. Falls nicht, beträgt die Länge 68-Bit und wird als „Shortened Access Code“ bezeichnet. Dieser enthält dann keinen Trailer (vgl. Abbildung 7.12). Verwendung findet der „Shortened Access Code“ bei Paging- und Inquery-Nachrichten (vgl. Kapitel 7.1.2.2). In diesem Fall wird der Access Code selbst als Signalisierungsnachricht gesendet und es ist weder ein Header noch ein Payload vorhanden. Weiterhin identifiziert der Access Code alle Pakete, die auf einem physikalischen Kanal ausgetauscht werden. Allen Paketen, die auf demselben physikalischen Kanal übertragen werden, geht derselbe Access Code voraus. Zudem dient der Access Code zur Synchronisation des Piconets sowie zum Geräteaufruf und -abfrage. Der Access Code beinhaltet verschiedene Typen die unterschiedliche Teile des Lower Address Part (LAP) nutzen um das entsprechende „Sync Word“ zu bilden (vgl. Kapitel 7.1.3.3).

- **Header:**
  Der Header enthält Link Control (LC) Informationen und besteht aus sechs Feldern (vgl. Abbildung 7.11): Logical Transport Address (LT_ADDR), TYPE, FLOW, Automatic Repeat Request Scheme (ARQN), Sequential Numbering Scheme (SEQN) und Header-Error-Check (HEC). Er enthält somit die Zieladresse, die Kennung des Pakettyps, Informationen für die Flusssteuerung und die HEC als Prüfsumme. Der gesamte Header, einschließlich des HEC, besteht aus 18-Bit. Da er essentielle Daten für den Transport des Pakets enthält, ist er mittels Forward Error Correction (FEC) geschützt und kommt so auf eine Gesamtgröße von 54-Bit.

- **Payload:**
  Die Länge des Payloads hängt jeweils vom Pakettyp und Verbindungsverfahren ab.

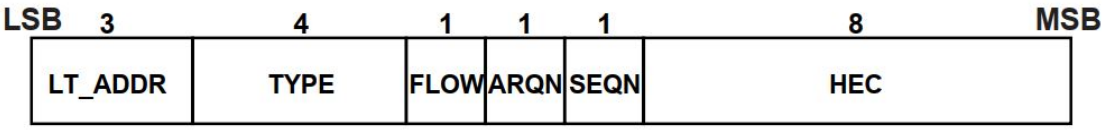

Abbildung 7.11: Bluetooth Header Format (vgl. [88, 433])

Bluetooth EDR Packet Format

Access Code und Header des EDR Paketformats sind in Format und Modulation mit den Paketen der BR identisch (vgl. Abbildung 7.10). Im Gegensatz zu BR, welche nur die GFSK Modulation einsetzt, wechselt die Modulationsart bei EDR innerhalb des Paketes. Die Synchronisationsphase (SYNC), der Payload und der Trailer werden hingegen mit der Differential Phase-Shift Keying (DPSK) Modulation moduliert. Durch die Verwendung von DPSK kann die Symbolrate, also die Anzahl der übertragenen Symbole pro Zeitspanne, erhöht werden.

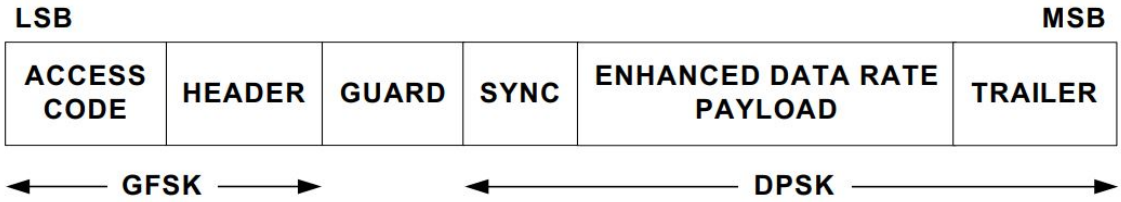

Abbildung 7.12: Bluetooth Enhanced Data Rate Packet Format (vgl. [88, 427])

Bluetooth LE Packet Format

Bluetooth LE verwendet nur ein Paketformat. Hierbei umfasst jedes Paket vier Felder: Preamble, Access-Address, PDU und CRC. Die Praemble dient hierbei beispielsweise zur Frequenz-Synchronisation. Die Access-Address ist mit dem Access Code vergleichbar. Die PDU unterscheidet sich, je nach Betriebsmodus des Gerätes nach Advertising PDUs oder Data Channel PDUs.

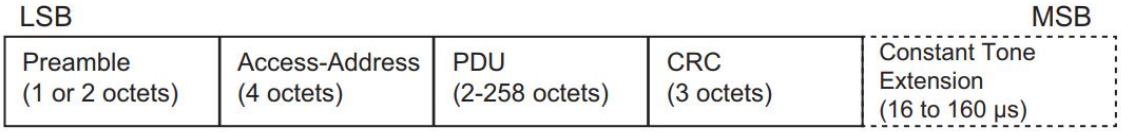

Abbildung 7.13: Bluetooth Low Energy Packet Format (vgl. [88, 2691])

7.1.3 Sicherheitsmaßnahmen

Der nachfolgende Abschnitt soll einen Überblick über die Sicherheitsarchitektur und -mechanismen des Bluetooth Standards geben und damit eine Basis für die Be-

drohungsmodellierung beziehungsweise der Bestimmung der Angriffsvektoren schaffen.

7.1.3.1 Sicherheitsarchitektur

Die Sicherheitsdienste unter Bluetooth befinden sich, in Bezug auf das OSI-Modell, auf Schicht 2, dem Data Link Layer. Diese umfassen Maßnahmen zur einseitigen und gegenseitigen Authentifizierung von, durch eine Bluetooth Device Address (BD_ADDR), identifizierten Kommunikationspartnern. Weiterhin schließt dies auch die Verschlüsselung der übertragenen Daten und die Autorisierung von Diensten ein. Hierbei ist beachten, dass es sich bei den Objekten in Bluetooth ausschließlich um Geräte handelt. Somit findet die Autorisierung als auch die Authentifizierung nur auf Gerätebasis und statt und wird nicht für einzelne Dienste oder Benutzer durchgeführt. Was die Zugriffsrechte betrifft, so kennt Bluetooth nur den erlaubten und den verbotenen Zugriff, wobei die Zugriffsberechtigung unbedingt oder bedingt erteilt werden kann. Hierbei kann die Zugriffsberechtigung einerseits von der Vertrauenswürdigkeit des zugreifenden Gerätes aber auch von einer korrekten, im Voraus durchgeführten Authentifizierung abhängen (vgl. [50, 932]). Zusammengefasst umfasst das Bluetooth Sicherheitsmodell fünf verschiedene Sicherheitsmerkmale: Pairing, Bonding, Device Authentication, Encryption und Message Integrity. Diese werden nachfolgend zusammenfassend dargestellt (vgl. [88, 255], [100, 11 f.]):

- **Pairing:**
  Hierunter versteht man das Verfahren zum Erstellen eines oder mehrerer gemeinsamer Sicherheitsschlüssel.
- **Bonding:**
  Bonding ist der Vorgang der Speicherung der während der Kopplung erstellten Schlüssel. Diese Schlüssel werden in nachfolgenden Verbindungen genutzt, um ein vertrauenswürdiges Gerätepaar zu bilden.
- **Device Authentication:**
  Hierbei wird überprüft, ob die beiden Geräte die gleichen Schlüssel besitzen. Zudem wird die Identität der kommunizierenden Geräte anhand ihrer Bluetooth Device Address (BD_ADDR) überprüft. Bluetooth bietet keine native Benutzerauthentifizierung.
- **Encryption:**
  Soll Informationskompromisse durch Abhören verhindern, indem sichergestellt wird, dass nur autorisierte Geräte auf übertragene Daten zugreifen können.

- **Message Integrity:**
  Message Integrity soll sicherstellen, dass eine zwischen zwei Bluetooth-Geräten gesendete Nachricht während der Übertragung nicht geändert wurde.

7.1.3.2 Sicherheitsmanager

Die zentrale Komponente der Sicherheitsarchitektur bildet der Sicherheitsmanager, der auch in Abbildung 7.2 dargestellt ist (vgl. Kapitel 7.1.2.1, [88, 2423 ff.], [50, 933]). Er verwendet einen Schlüsselverteilungsansatz, um Identitäts- und Verschlüsselungsfunktionalitäten in der drahtlosen Kommunikation durchzuführen. Dies bedeutet, dass jedes Gerät die Schlüssel generiert und kontrolliert, die es selbst vergibt. Kein anderes Gerät beeinflusst die Erzeugung dieser Schlüssel. Nachfolgend sollen die Aufgaben kurz aufgeführt werden:

- Verwaltung von Sicherheitsattributen von Diensten und Geräten
- Berechtigungsüberprüfung der Zugriffsanfragen beim Verbindungsaufbau
- Authentifikation
- Ver- und Entschlüsselung von Daten vor Verbindungsherstellung zur Anwendung
- Initiierung und Bearbeitung der Eingabe einer External Security Control Entity (ESCE). Hierunter versteht man beispielsweise die PIN-Eingabe eines Benutzers
- Initiierung des Pairing zwischen zwei Geräten

7.1.3.3 Bluetooth Device Address

Jedes Bluetooth-Gerät hat zur Identifikation eine eindeutige 48-Bit lange Adresse, die als Bluetooth Device Address oder BD_ADDR bezeichnet wird. Diese wird nach dem IEEE Standard 802-2014 für Local and Metropolitan Area Networks definiert und stellt einen 48-Bit Extendend Unique Identifier (EUI-48) dar. Über diesen lassen sich bis zu 281 Billionen Bluetooth-Geräte eindeutig identifizieren. Die Struktur der Bluetooth Device Address ist in Abbildung 7.14 dargestellt. Im Grunde ist sie vergleichbar mit einer MAC-Adresse für Ethernet- und Wi-Fi-Netzwerke, allerdings gibt es einige wesentliche Unterschiede zu diesen. Im Gegensatz zu einer MAC-Adresse wird die Bluetooth Device Address im gesamten Bluetooth Protokoll für Identität, Authentizität und Low-Level-Kommunikation verwendet (vgl. [88, 383 f.]). Die

Bluetooth Device Address unterteilt sich in drei Bereiche: LAP, Upper Address Part (UAP) und Non-significant Address Part (NAP). Nachfolgend werden die drei Bestandteile kurz erläutert (vgl. [93], [101]).

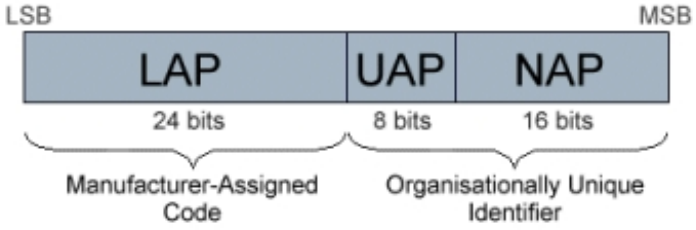

Abbildung 7.14: Bluetooth Device Address (vgl. [101])

- **LAP:**
  Dieser Teil der Bluetooth Device Address wird dem Hersteller zugewiesen. Sie ist Teil des Access Codes der Bluetooth-Pakete im Baseband. Da der LAP im gesamten Bluetooth-Protokoll vielfach Verwendung findet, sollte er entsprechend geheim gehalten werden.
- **UAP:**
  Der UAP wird den Herstellern von der IEEE zugewiesen. Er wird unter anderem dafür verwendet das HEC zu erzeugen, das zur Erkennung von Fehlern in Bluetooth Paketen verwendet wird.
- **NAP:**
  Der NAP hat, wie der Name schon sagt, keine besondere Bedeutung für die Bluetooth Kommunikation.

7.1.3.4 Sicherheitsmodi BR/EDR

Bluetooth BR und EDR arbeiten in einem von vier definierten Sicherheitsmodi (vgl. [88, 2121 ff.], [50, 934 ff.]). Die Sicherheitsmodi bestimmen hierbei, wann ein Bluetooth-Gerät die Sicherheit einleitet, nicht aber ob es Sicherheitsfunktionen unterstützt. Grundlegend können die Sicherheitsmodi in zwei Kategorien eingeteilt werden: „Service Level Enforced Security“ und „Link Level Enforced Security“. Ersterer bezieht sich auf Authentifizierungs- und Verschlüsselungs-Setup-Verfahren, die stattfinden, nachdem die physische Bluetooth-Verbindung bereits vollständig hergestellt und die logischen Kanäle teilweise eingerichtet wurden. „Link Level Enforced Security“ bezieht sich hingegen auf Authentifizierungs- und Verschlüsselungs-Setup-Verfahren,

die stattfinden, bevor die physische Bluetooth-Verbindung vollständig hergestellt ist. Soll eine Bluetooth-Verbindung etabliert werden, wird überprüft, in welchem Sicherheitsmodus sich die Geräte befinden. Abhängig davon werden dann die unterschiedlichen Sicherheitsdienste angestoßen. Nachfolgend werden die vier Sicherheitsmodi kurz betrachtet:

- **Security Mode 1 (Non-Secure):**
  Geräte des Security Mode 1 gelten als nicht sicher. Sicherheitsfunktionen hinsichtlich Authentifizierung und Verschlüsselung werden nie initiiert, so dass das Gerät und die Verbindungen anfällig für Angreifer sind. Das schließt auch mit ein, dass Geräte in diesem Modus keine Mechanismen verwenden, um eingehende Verbindungen von anderen Geräten zu unterbinden. Gemäß der Bluetooth Spezifikation können alle Geräte ab Version 2.1 und höher diesen Security Modus, auf Grund der Abwärtskompatibilität mit älteren Geräte, nutzen. Dies ist jedoch nicht empfehlenswert.

- **Security Mode 2 (Service Level Enforced Security):**
  Im Security Mode 2 können Sicherheitsverfahren nach dem Verbindungsaufbau, aber vor dem Aufbau des logischen Kanals, eingeleitet werden. Für diesen Sicherheitsmodus steuert ein lokaler Sicherheitsmanager (vgl. Kapitel 7.1.3.2) den Zugriff auf bestimmte Dienste und bietet somit Sicherheitsdienste auf der Dienstebene. Hierdurch können unterschiedliche Anwendungen mit unterschiedlichen Sicherheitsfunktionen ausgestattet werden. Bevor ein sicherheitskritischer Dienst genutzt werden kann, muss jedoch erst eine L2CAP-Verbindung aufgebaut werden. Bevor dies geschieht, muss jedoch erst überprüft werden, ob der Zugriff auf den Dienst beziehungsweise das Gerät erlaubt ist. Diese Informationen stehen entsprechend in der Gerätedatenbank und ergeben sich aus dem Vertrauenslevel des Gerätes/Dienstes. Ist beispielsweise ein Gerät als vertrauenswürdig eingestuft, ist keine Authentifikation mehr erforderlich und die Verbindungsanfrage wird entsprechend akzeptiert. Security Mode 2 initiiert den Aufbau von Link Keys über Personal Identification Number (PIN) Pairing.

- **Security Mode 3 (Link Level Enforced Security):**
  Security Mode 3 ist der „Link Level Enforced Security“ Mode, in dem ein Bluetooth-Gerät Sicherheitsverfahren einleitet, bevor die physische Verbindung vollständig hergestellt ist. Er setzt dabei eine Ebene tiefer, nämlich auf der Link-Ebene, an und bietet eine Art Grundschutz, der für alle Anwendungen gleich ist. Eine anwendungsspezifische Differenzierung wie im Security Mode 2

ist hierbei nicht gegeben. Bluetooth-Geräte, die im Security Mode 3 betrieben werden, erfordern Authentifizierung und Verschlüsselung für alle Verbindungen zum und vom Gerät. Daher kann auch die Diensterkennung erst nach der Authentifizierung, Verschlüsselung und Autorisierung durchgeführt werden. In der Praxis wird in der Regel der Security Mode 3 verwendet. Security Mode 3 initiiert, wie Security Mode 2, den Aufbau von Link Keys über PIN (PIN) Pairing.

- **Security Mode 4 (Service Level Enforced Security):**
  Security Mode 4 entspricht dem Security Mode 2 und zählt ebenfalls zu den „Service Level Enforced Security“ Modis. Zusätzlich ist im Mode 4 festgelegt, mit welcher Variante der Secure Simple Pairing (SSP) Authentisierung der gemeinsame Link Key vereinbart werden soll.

7.1.3.5 Sicherheitsstufen

Zusätzlich zu den in Kapitel 7.1.3.4 angesprochenen Sicherheitsmodi, gibt es im Security Mode 2 beziehungsweise 4 verschiedene Sicherheitsstufen für die Geräte. Diese werden auch als „trust-level“ bezeichnet (vgl. [50, 936]). Unterschieden wird hierbei zwischen vertrauenswürdigen („trusted“) und nicht vertrauenswürdigen („untrusted“) Geräten. Hierbei kennzeichnen sich vertrauenswürdige Geräte dadurch aus, dass sie in einer festen, authentifizierten Verbindung mit einem anderen Gerät stehen. Diese haben unbeschränkten Zugriff auf alle von ihnen angebotenen Dienste. Nicht vertrauenswürdige Geräte hingegen stehen in keiner festen Beziehung zueinander. Der Zugriff auf Dienste ist daher beschränkt.

7.1.3.6 Sicherheitsmodi LE

Die Sicherheitsmodi für Bluetooth LE ähneln den Sicherheitsmodi auf der „Service Level Enforced Security“-Ebene von Bluetooth BR/EDR, da jeder Dienst seine eigenen Sicherheitsanforderungen haben kann (vgl. [100, 29]). Bluetooth LE legt jedoch auch fest, dass jede Serviceanforderung auch eigene Sicherheitsanforderungen aufweisen kann. Ein Gerät erzwingt die dienst-bezogenen Sicherheitsanforderungen, indem es dem entsprechenden Sicherheitsmodus und der entsprechenden Sicherheitsstufe entspricht. Die Sicherheitsmodi werden nachfolgende ebenfalls kurz dargestellt:

- **Low Energy Security Mode 1:**
  Der Low Energy Security Mode 1 umfasst mehrere Stufen der Verschlüsselung. Stufe 1 spezifiziert keine Sicherheit, d.h. es wird keine Authentifizierung

und Verschlüsselung eingeleitet. Stufe 2 steht für eine nicht authentifizierte Kopplung mit Verschlüsselung. Stufe 3 umfasst eine authentifizierte Kopplung mit Verschlüsselung. Stufe 4 erfordert letztendlich eine authentifizierte, energiesparende und sichere Verbindung, die mit einer Verschlüsselung gekoppelt ist.

- **Low Energy Security Mode 2:**
  Im Hinblick auf den Low Energy Security Mode 2 ist festzustellen, dass er ebenfalls mehrere Stufen umfasst. Im vorliegenden Fall betrifft dies die Datensignierung. Diese bietet zwar eine hohes Maß an Datenintegrität, aber keine Vertraulichkeit. Stufe 1 erfordert eine nicht authentifizierte Kopplung mit Datensignierung. Stufe 2 hingegen erfordert eine authentifizierte Kopplung mit Datensignierung.

7.1.3.7 Schlüsselarten

Bluetooth nutzt für verschiedene Aufgaben unterschiedliche Schlüssel. Der wohl wichtiges Schlüssel ist der Link Key, mit dem sich die Geräte gegenseitig authentifizieren. Aus diesem wird auch, nach der Authentisierung, der Encryption Key berechnet. Man unterscheidet hierbei vier verschiedene Typen an Link Keys. Diese sollen nachfolgend dargestellt werden (vgl. [50, 938 ff.], [88, 1772 ff.]):

- **Unit Key (Geräteschlüssel):**
  Der Unit Key wird bei der erstmaligen Verwendung des Gerätes, unter Verwendung einer 128-Bit Zufallszahl und der Bluetooth Device Address (vgl. Kapitel 7.1.3.3), generiert. Er wird im nicht-flüchtigen Speicher des Geräts abgelegt und normalerweise nie geändert. Als Link Key sollte er nur in Ausnahmefällen genutzt werden, beispielsweise wenn ein Geräte keine weiteren Schlüssel speichern kann.

- **Combination Key (Kombinationsschlüssel):**
  Der Combination Key ist ein Key, der in Abhängigkeit von den beiden an der Sitzung beteiligten Geräten und für jede Sitzung neu erzeugt wird. Für dessen Berechnung findet auch die Bluetooth Device Address und Zufallszahlen ihre Verwendung.

- **Master Key (Masterschlüssel):**
  Der Master Key ist ein temporärer Schlüssel, der in einem Piconet gemeinsam als Link Key und Encryption Key genutzt wird. Er vereinfacht die 1-n-Kommunikation, also den Broadcast, da die Daten nicht mehr mit mehreren Link Keys

der unterschiedlichen Geräte verschlüsselt werden müssen. Das hat zur Folge, dass alle vom Master kontaktierten Slaves statt ihres eigenen Link Keys den vom Master verschlüsselt verteilten Master Key nutzen. Jeder Slave ist somit in der Lage, Nachrichten des Masters zu entschlüsseln, die unter Umständen auch nicht für ihn gedacht waren.

- **Init Key (Initialisierungsschlüssel):**
  Der Init Key ist ein 128-Bit Schlüssel, der aus einer beiden Geräten bekannten Zufallszahl, den jeweiligen BD_ADDR-Werten und einer geheimen PIN erstellt wird (vgl. Kapitel 7.1.3.3, 7.1.3.8). Er dient zum sicheren Austausch von Informationen in der Phase der Vereinbarung eines Verbindungsschlüssels. Anschließend wird er vernichtet.

7.1.3.8 Pairing und Link Key Generierung

Ein wesentlicher Bestandteil der von Bluetooth bereitgestellten Authentifizierungs- und Verschlüsselungsmechanismen ist die Generierung eines geheimen symmetrischen Schlüssels. Bei Bluetooth BR/EDR bezeichnet man diesen Schlüssel als *Link Key*. Bluetooth LE hingegen verwendet die Bezeichnung *Long Term Key (LTK)*. Weiterhin wird beim Bluetooth LE Legacy Pairing ein *Short Term Key (STK)* generiert, welcher für die Verteilung des Slave und/oder Master LTK verwendet wird. Bei Bluetooth LE Secure Connections Pairing wird der LTK hingegen direkt von jedem Gerät erstellt und nicht verteilt (vgl. [100, 15]). Wie bereits in Kapitel 7.1.3.4 angesprochen gibt es unterschiedliche Arten der Generierung von Link Keys. Der Bluetooth BR/EDR Security Mode 2 und 3 nutzt hier das PIN/Legacy Pairing. Bluetooth BR/EDR Security Mode 4 hingegen nutzt zur Generierung der Link Keys das SSP. Recherchiert man in der Bluetooth Spezifikation in Bezug auf die Sicherheitsarchitektur nach den Pairing-Prozessen und der unterschiedlichen Verfahren zur Link Key Generierung, so kann man folgende Feststellung tätigen (vlg. [88, 255 f.]). Vor Bluetooth Version 2.1 nutzt Bluetooth BR/EDR das PIN/Legacy Pairing. Ab Version 2.1 wurde dann das SSP Pairing eingeführt, welches nun auch freigegebene Algorithmen nach dem Federal Information Processing Standard (FIPS) nutzt. Als Beispiel ist hier P-192 Elliptic Curve zur Key Generierung zu nennen. Mit Version 4.1 setzt BR/EDR auf Secure Connections. Dies stellt ein Update des SSP dar. Beispielsweise wird hier nun die P-256 Elliptic Curve für die Key Generierung verwendet. Bluetooth LE setzt in den Version 4.0 und 4.1 auf das SSP und wird, wie oben angesprochen, als Legacy Pairing bezeichnet. Ab Bluetooth LE Version 4.2 wird ebenfalls auf Secure Connections gesetzt. Nachfolgend sollen die unterschied-

lichen Pairing Arten sowie die Erstellung der Link Keys von Bluetooth BR/EDR sowie LE näher betrachtet werden.

BR/EDR PIN/Legacy Pairing:

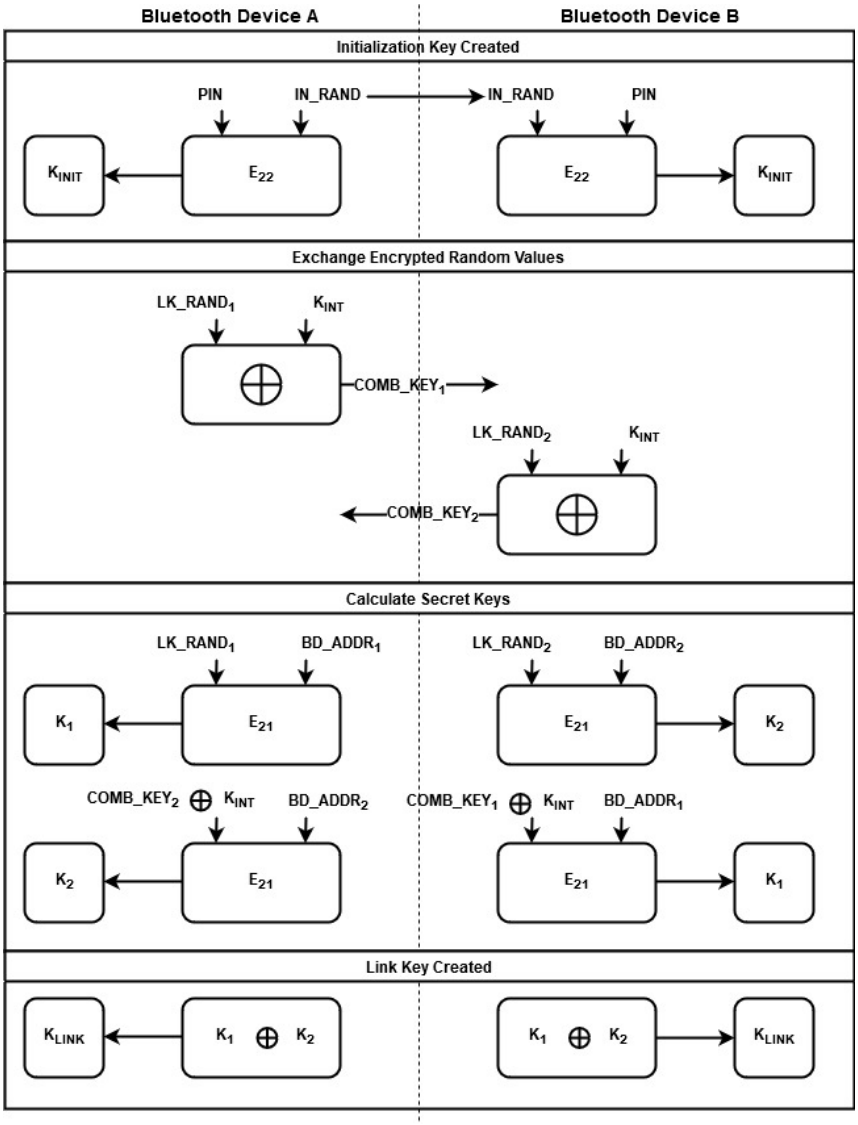

Abbildung 7.15: Bluetooth Link Key Generierung mit PIN (eigene Darstellung, angelehnt an [100, 16])

Die gesamte Sicherheit der Bluetooth-Kommunikation hängt, wie oben erwähnt, am symmetrischen Link Key. Beim PIN- beziehungsweise Legacy-Pairing wird dieser von der eingegebenen PIN abgeleitet. Abbildung 7.15 stellt die Generierung des Link Keys grafisch dar. Den ersten Schritt bildet hierbei die Erstellung des Init Keys K_{INIT}, also des Initialisierungsschlüssels. Dieser wird mittels dem E_{22}-Algorithmus

aus der PIN sowie einer Zufallszahl IN_RAND gebildet. Ist die PIN kleiner als 16 Byte, so wird sie mit der BD_ADDR des initialisierenden Gerätes aufgefüllt. Die Zufallszahl IN_RAND wird ebenfalls vom initialisierenden Gerät, in diesem Fall Bluetooth Device A, erzeugt und unverschlüsselt an das zweite Gerät gesendet. Anschließend erzeugen beide Geräte einen Zufallswert $COMB\_KEY_x$, welcher jeweils aus K_{INIT} und einer weiteren Zufallszahl LK_RAND_x, je Gerät, welche durch XOR-Verknüpfung generiert und jeweils mittels K_{INIT} verschlüsselt an das andere Gerät gesendet wird. Im nächsten Schritt berechnen beide Geräte jeweils zwei Schlüssel, K_1 und K_2. Hierbei kommen die zuvor genutzten Zufallszahlen LK_RAND_x, die BD_ADDR_x der Geräte, der Init Key K_{INIT},die Zufallswerte $COMB\_KEY_x$ sowie der E_{22}-Algorithmus zum Einsatz. Anschließend sind beide Geräte dazu in der Lage, mittels einer XOR-Verknüpfung von K_1 und K_2, den Link Key K_{LINK} zu bilden. Als Hinweis ist zu erwähnen, dass die E_x-Algorithmen in der jeweiligen Bluetooth Spezifikation definiert sind (vgl. [88, 1797 ff.]). Nachdem die Generierung des Link Keys abgeschlossen ist, wird das Pairing abgeschlossen. Hierzu authentifizieren sich die Geräte gegenseitig, um sicherzustellen, dass sie über den gleichen Link Key verfügen.

SSP, LE Legacy Pairing und Secure Connections:

Im Vergleich mit dem PIN/Legacy Pairing vereinfacht SSP und Secure Connections den Pairing-Prozess dahingehend, dass sie eine Reihe von Modulen – sogenannte Association Models – bereitstellen, die flexibel in Bezug auf die Ein- und Ausgabemöglichkeiten der Geräte genutzt werden können. Hierbei gibt es vier Association Models (vgl. [50, 948 ff.], [88, 259 ff.]).

- **Numeric Comparison:**
  Kann für Geräte genutzt werden, die beide ein Display und die Möglichkeit für die Eingabe von „Ja“ beziehungsweise „Nein“ haben. Dem Benutzer wird auf beiden Displays eine sechsstellige Nummer gezeigt. Dies gilt es zu vergleichen und entsprechend zu bestätigen. Im Gegensatz zum PIN/Legacy Pairing wird die PIN nicht zur Schlüsselerzeugung verwendet, so dass ein Angreifer, der die PIN ausspioniert, den verwendeten Schlüssel nicht ableiten kann.

- **Passkey Entry:**
  Diese Methode wurde für den Fall entwickelt, wenn ein Gerät zwar ein Display aber keine Eingabemöglichkeit besitzt. Dieses Gerät zeigt eine PIN an, die dann im anderen Gerät eingegeben werden muss.

- **Just Works:**
  Bei dieser Methode haben beide Geräte weder ein Display noch Eingabemöglichkeiten. Als Beispiel kann hier ein Bluetooth Lautsprecher oder Kopfhörer genannt werden. Die Methode entspricht technisch der „Numeric Comparison“. Unterschied ist jedoch, dass der Benutzer den verwendeten Wert nicht prüfen kann und er somit keinen Schutz gegen „Man-in-the-Middle“ Angriffe bietet.

- **Out of Band (OOB):**
  Diese Methode ist in erster Linie für Geräte gedacht, bei denen ein Out of Band-Mechanismus verwendet wird, um sowohl andere Geräte zu entdecken als auch kryptographische Daten, die im Pairing-Prozess verwendet werden, auszutauschen oder zu übertragen. Um aus sicherheitstechnischer Sicht effektiv zu sein, sollte der Out of Band Kanal im Vergleich zum Bluetooth Funkkanal unterschiedliche Sicherheitseigenschaften aufweisen. Zudem sollte er resistent gegen „Man-in-the-Middle“ Angriffe sein. Andernfalls kann die Sicherheit während der Authentifizierung beeinträchtigt werden.

Abbildung 7.16 stellt den Pairing-Prozess für SSP, LE Legacy Pairing und Secure Connections dar. Der Prozess basiert auf einem ECDH Public Key Verfahren und ist nachfolgend kurz beschrieben:

- **Phase 1:**
  Die Phase 1 findet unverschlüsselt, als Klartext statt. Hierbei tauschen die Geräte Informationen darüber aus, was sie können und welche Pairing-Methoden in der Phase 2 zur Verwendung kommen sollen.

- **Phase 2:**
  Phase 2 variiert, je nachdem ob es sich um SSP/LE Legacy Pairing oder Secure Connections handelt. Wie bereits zu Beginn dieses Kapitels angesprochen, wird bei SSP beziehungsweise LE Legacy Pairing ein STK, bei Secure Connections ein LTK erzeugt. Voraussetzung hierfür ist der vorherige Austausch von Zufallswerten und die Einigung auf einen temporären Schlüssel.

- **Phase 3:**
  In der letzten Phase wird der in Phase 2 erzeugte Schlüssel, STK oder LTK, genutzt, um die restlichen benötigten Schlüssel für die sichere Kommunikation zu verteilen. Zudem werden auch Schlüssel, beispielsweise zum Signieren von Daten, erzeugt.

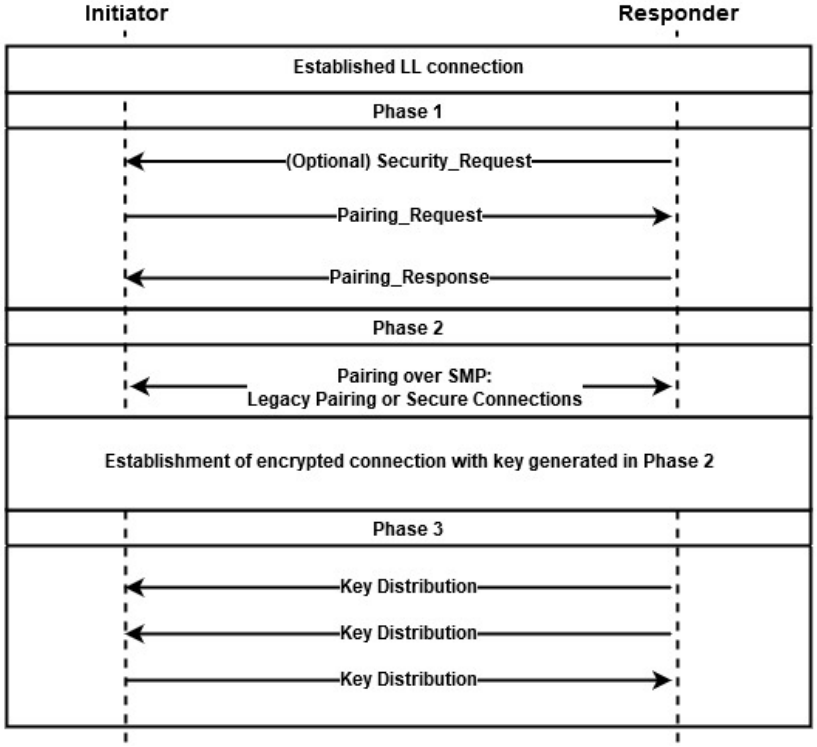

Abbildung 7.16: Bluetooth SSP/Legacy Pairing und Secure Connections (eigene Darstellung, angelehnt an [102])

7.1.4 Schwachstellen / Bestimmung der Angriffsvektoren

Im vorausgegangenen Teil wurden Sicherheitsmaßnahmen von Bluetooth dargestellt. Im nachfolgenden Teil werden nun einige Schwachstellen und die theoretisch möglichen Angriffsvektoren dargestellt, die für Bluetooth existieren.

Zudem gibt es einige Forschungsarbeiten, die sich mit der Sicherheitsanalyse von Bluetooth beschäftigt haben. Als Beispiel ist hier eine Publikation der National Institute of Standards and Technology (NIST) zu nennen, die umfassend über Bluetooth Sicherheit informiert (vgl. [100]). Laut dieser bietet Bluetooth viele Vorteile und Vorzüge, die aber mitunter Risiken in Kauf nehmen. Generell sind Bluetooth und Bluetooth Geräte anfällig für allgemeine Bedrohungen des drahtlosen Netzwerks. Hierzu lassen sich beispielsweise die Folgenden zählen:

- DoS
- Man-in-the-Middle Angriff
- Eavesdropping
- Nachrichten-Modifizierung

Nachfolgend werden einige mögliche Schwachstellen der aktuell anzutreffenden Bluetooth-Version dargestellt. Diese Betrachtung umfasst die derzeitige 5er Version von Bluetooth sowie die 4er Versionen.

7.1.4.1 SSP Association Model „Just Works“

In Kapitel 7.1.3.8 wurde der SSP Pairing-Prozess erläutert. Dieser unterstützt vier verschiedene Association Models. Wie auch die Bluetooth Spezifikation angibt, bietet das „Just Works“-Model keinen Schutz gegen „Man-in-the-Middle“ Angriffe (vgl. [88, 260]). Dies kann unter Umständen ein Angreifer ausnutzen, was letztendlich in einem nicht-authentifizierten Link Key resultieren kann. Ein Angreifer würde sich in diesem Fall in die Phase 1 des Pairing-Prozesses einklinken und könnte die ausgetauschten Nachrichten über die Ein- und Ausgabemöglichkeiten der Geräte so manipulieren, dass diese nur die „Just Works“-Methode als Pairing-Methode ableiten. Dies ist möglich, da die Nachrichten in Phase 1 über einen unverschlüsselten und nicht authentifizierten Kanal ausgetauscht werden (vgl. Kapitel 7.1.3.8, Abbildung 7.16). Durch die Auswahl von „Just Works“ als Pairing-Methode kann auch das Pairing keine Authentifizierung leisten, sodass der „Man-in-the-Middle“ weiterhin aktiv bleibt. Auch die Berechnungen der Werte in Stufe 2 bieten keinen wirksamen Schutz, da der Angreifer diese ebenfalls mit den manipulierten Werten aus Phase 1 berechnen kann. Sicherer wäre es daher, wenn Bluetooth BR/EDR Geräte, innerhalb des Pairing-Prozesses mit SSP Just Works, einen „Man-in-the-Middle“-Schutz einbauen, der nicht authentifizierte Link Keys erkennen kann und diese entsprechend ablehnt.

7.1.4.2 Authentication Requests

Laut Bluetooth Spezifikation kann die Authentifizierung nach dem Verbindungsaufbau jederzeit explizit durchgeführt werden. Hierbei kann der Link Key neu angefordert werden, falls dieser nicht vorhanden ist (vgl. [88, 1545 f.]. Abbildung 7.17 stellt den Authentication Request exemplarisch dar. Für die Authentifizierungsanfragen gibt es kein Warteintervall. Ein Angreifer könnte somit eine große Anzahl an, mit dem Link Key verschlüsselten, Challenge-Response Paketen sammeln. Unter Umständen erhält er so die Möglichkeit, Informationen zum Link Key zu erhalten und entsprechend weitere Angriffe durchzuführen.

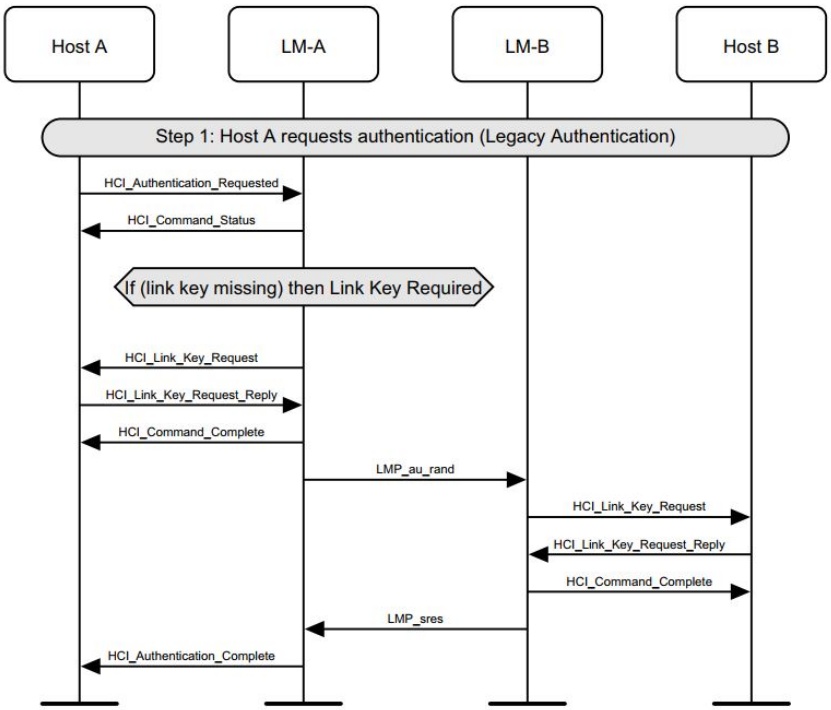

Abbildung 7.17: Bluetooth Authentication Request (vgl. [88, 1545])

7.1.4.3 Jamming

Mittels Jamming, also dem Senden von Störsignalen, lassen sich prinzipiell alle Funktechnologien in einen DoS Zustand bringen. Viele Technologien haben allerdings mittlerweile Techniken und Systeme, welche permanente Störsignale entdecken können und entsprechend reagieren (vgl. Kapitel 7.1.2.3, Abbildung 7.9). Brauer und Kollegen beschreiben in einem Artikel über das Jamming von Bluetooth LE, dass es effizienter und unauffälliger sei, Advertising-Pakete selektiv zu stören (vgl. [103, 1]). Hierbei beschreiben sie ihre Idee eines Bluetooth LE Jammers, der die Advertising-Pakete mittels der darin enthaltenen Absenderadresse stört. Bluetooth LE nutzt drei Advertising Kanäle (vgl. Kapitel 7.1.2.3, Abbildung 7.8). Der Störsender lauscht hierbei zunächst auf dem ersten Advertising Kanal, bis er Pakete erkennt, die als Access-Address (vgl. Abbildung 7.13) den Wert *0x8E89BED6* aufweisen. Diese Adresse entspricht der standardisierten Adresse für Advertising-Pakete (vgl. [88, 2692]). Ab der Position, an der die Access-Address erkannt wird, wertet der Jammer noch die nächsten 64-Bit aus. Diese umfassen den Header, das Length Field und einen Teil des Payloads. Hierdurch wird die Absenderadresse bekannt. Der entwickelte Jammer besteht aus zwei Komponenten, der Beaconerkennung und dem eigentlichen Störsender. In der Erkennungsphase dekodiert der Jammer empfangene

Beacon Frame Header, um basierend auf konfigurierten Blacklisting / Whitelisting von Bluetooth Device Addresses zu entscheiden, ob der verbleibende Frame gestört werden soll oder nicht. Ist die BD_ADDR in der Blacklist, wechselt der Jammer in die Jamming-Phase. Hierbei sendet er jeweils ein kurzes Störsignal auf dem verwendeten Kanal. Die Übertragung des Advertising-Paket wird dadurch gestört, wodurch die CRC-Prüfsumme auf Empfängerseite nicht mehr zu den Paketdaten passt. So ein Paket wird entsprechend verworfen. Wurde der Kanal gestört, wird die Übertragung gestoppt und es erfolgt ein Wechsel auf den nächsten Advertising-Kanal, da Bluetooth LE Geräte meistens nacheinander auf allen drei Advertising-Kanälen senden (vgl. Abbildung 7.18).

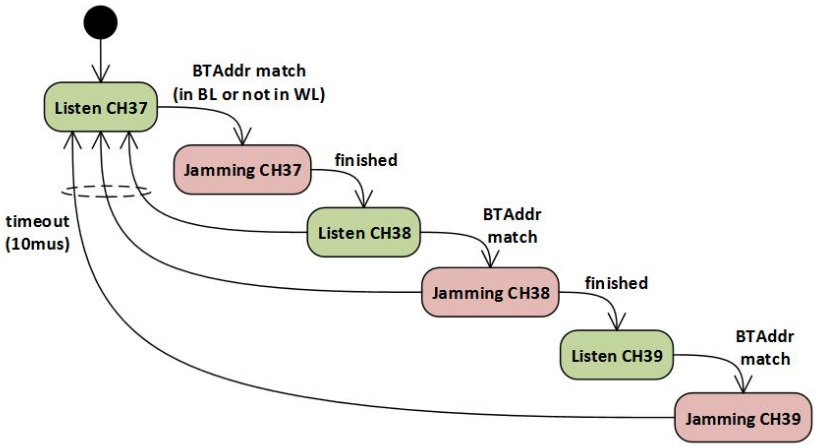

Abbildung 7.18: Bluetooth Proposed Jammer (vgl. [103, 4])

Wenn der Kanal von den Zielgeräten nicht genutzt wird oder wenn der Jammer keine Frames erkennen kann, wird die Funktion eines 10 ms Timeout genutzt. Dies 10 ms ist die maximale Zeit zwischen zwei Advertising Frames, die in aufeinanderfolgend genutzten Kanälen in einem Advertising Event senden (vgl. Abbildung 7.19, [103, 4]). Nach einem Timeout kehrt der Störsender zum ersten Advertising Kanal zurück. Tritt kein Timeout auf, wird der Störvorgang wie beim vorherigen Kanal fortgesetzt und anschließend auf dem letzten Kanal wiederholt.

7.2 Bewertung der Informationen / Risikoanalyse

Die zuvor dargestellten möglichen Angriffsvektoren von Bluetooth wurden anhand des in Kapitel 2.5.2 dargestellten „DREAD“-Modell bewertet. Das Ergebnis ist in Tabelle 7.1 dargestellt.

	SSP A. M. „Just Works“	**Authentication Requests**	**Jamming**
***D**amage Potential*	3	2	1
***R**eproducibility*	3	3	2
***E**xploitability*	1	1	1
***A**ffected Users*	1	1	1
***D**iscoverability*	2	2	2
Ergebnis	**10**	**9**	**7**
Einstufung	**Medium**	**Medium**	**Low**

Tabelle 7.1: Bewertung dargestellter Angriffsvektoren von Bluetooth

Die dargestellten Angriffsvektoren weisen nur eine *Medium* und *Low* Einstufung auf. Im Falle des beschriebenen *Jamming* Angriffs ist dies wie folgt zu begründen. Zum einen bedarf es einiger Erfahrung des Angreifers sowohl Hardware als auch Software nachzustellen, da hierzu aktuell keine näheren Informationen vorliegen. Wird der Advertising-Prozess des Beacons von dem dargestellten Round-Robin-Verfahren auf ein anderes Verfahren abgeändert, so funktioniert der aktuelle Jamming-Angriff nicht mehr.

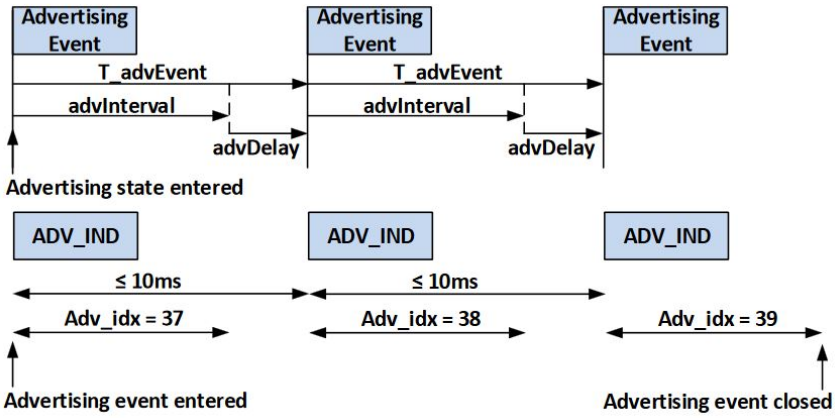

Abbildung 7.19: Bluetooth LE Advertising (vgl. [103, 2])

In Bezug auf den *SSP Association Model „Just Works“* Angriff wurde bereits bei dessen Darstellung in Kapitel 7.1.4.1 angesprochen, als Lösung des Problems einen „Man-in-the-Middle“-Schutz einzubauen. Dieser soll in der Lage sein, beim Pairing-Prozess nicht authentifizierte Link Keys zu erkennen. Der Angriff wurde als *Medium* eingestuft, da er eine bekannte Schwachstelle darstellt, die von einem versierten Angreifer ausgenutzt werden kann. Der letzte dargestellte Angriffsvektor betrifft die *Authentication Requests*. Dieser wurde ebenfalls als *Medium* eingestuft, da hier keine Einschränkungen in der Bluetooth Spezifikation festgelegt wurden. Andererseits benötigt dieser Angriffsvektor auch erfahrene und versierte Angreifer die entsprechende Kenntnisse der Materie aufweisen um diese Schwachstelle auszunutzen. Die Schwachstelle könnte dahingehend geschlossen werden, indem die Authentifizierungsanfragen entsprechend beschränkt und Warteintervalle eingeführt werden.

Bezugnehmend auf die Analyse der NIST, die generelle Kritik an der Bluetooth LE Spezifikation übt, werden insbesondere noch die folgenden Eigenschaften kritisiert (vgl. [100, 37 ff.]):

- Bluetooth LE Security Mode 1 - Stufe 1 ermöglicht die Nutzung ohne jegliche Sicherheitsmechanismen. So wird hier weder eine Verschlüsselung noch Authentifizierung vorgenommen (vgl. Kapitel 7.1.3.6).
- Statische Diffie-Hellmann-Schlüssel dürfen verwendet werden. Sinnvoller wäre hier eine neue Generierung der Schlüssel vor jeder Verbindung
- Grundsätzlich findet nur eine Authentifizierung des Gerätes statt. Nutzer werden nicht authentifiziert.

7.3 Abschlussanalyse

Nach der Recherche der Bluetooth Spezifikation und Analyse der Schwachstellen unter Zuhilfenahme existierender Forschungsarbeiten kommt man zu dem Ergebnis, dass die Sicherheitsmechanismen von Bluetooth das Ergebnis des Gleichgewichts zwischen Sicherheit und Benutzerfreundlichkeit ist. Hierbei lässt die Bluetooth Spezifikation den Herstellern und Implementierenden bewusst viel Spielraum auf Kosten der Sicherheit. Ursache hierfür ist vor allem Bluetooth LE, der in ressourcenschonenden Fällen beispielsweise in Beacons zum Einsatz kommt. Ab Bluetooth BR/EDR Version 4.1 beziehungsweise LE Version 4.2 wurde das Secure Connections Pairing eingeführt, welcher einen authentifizierten und vertraulichen Kanal zur Verfügung stellt. Die Gefahr, die dennoch bleibt, ist der Downgrade-Angriff durch einen „Man-in-the-Middle“ Angriff, welcher in Kapitel 7.1.4.1 beschrieben wurde.

In Fällen, in denen mit Bluetooth LE nur Verbindungen im Security Mode 1 - Level 1 aufgebaut werden können, weil beispielsweise keine Ein- und Ausgabemöglichkeiten existieren, müssen die Hersteller der Geräte tätig werden, um die Authentifizierung sicherzustellen (vgl. Kapitel 7.1.3.6). In Bezug auf die Praxis fällt auch auf, dass die Hersteller häufig die Spielräume der Spezifikation ausnutzen und auf Sicherheitsmaßnahmen verzichten, obwohl deren Implementierung möglich wäre.

8 Schlussbemerkung

Ziel der vorliegenden Arbeit war die Schwachstellenanalyse von Funkprotokollen am Beispiel von Smart Home Anwendungen. Die Analyse wurde hierbei auf die drei am Häufigsten anzutreffenden Protokolle eingegrenzt. Bei deren Analyse konnte festgestellt werden, dass es einen großen Unterschied macht, ob die Protokolle frei oder proprietär sind. Die Informations- und Recherchebasis für ZigBee und Bluetooth konzentrierte sich vor allem auf deren öffentlich freigegebenen Spezifikationen. Ergänzt wurden diese durch die vielfältigen Forschungsarbeiten, die zu diesen Protokollen in einschlägigen Wissenschaftsforen, wie beispielsweise ResearchGate, vorhanden sind. Für Z-Wave, als einziges proprietäres Protokoll in dieser Ausarbeitung, war die Informations- und Recherchebasis entsprechend geringer. Dies liegt vor allem an der NDA Verordnung, die den Herstellern auferlegt wurde. Freigegebene Spezifikationen des Z-Wave Public Bereich waren eher weniger hilfreich für die Ausarbeitung, da deren Inhalt eher auf die grundlegende Themen, wie beispielsweise die „Command Classes“, ausgerichtet sind. Detaillierte Beschreibungen zur Sicherheit, besonders in Bezug auf die S2 Sicherheit, ist nur sehr spärlich und oberflächlich vorhanden.

Da jede Schwachstellenanalyse mit einer Abschlussanalyse abgeschlossen wurde, werden die dort dargelegten Aussagen und Ausblicke nicht nochmals in dieser Schlussbemerkung wiederholt. Es sei daher auf die jeweiligen Abschlussanalysen verwiesen.

Literaturverzeichnis

[1] Wirtschaftswoche: *Experte warnt: Smart Home oft Einfallstor für größere Netz-Attacken.* https://www.wiwo.de/technologie/digitale-welt/experte-warnt-smart-home-oft-einfallstor-fuer-groessere-netz-attacken/23223416.html. Version: 24.10.2018, Abruf: 01.04.2019

[2] Statista (Hrsg.): *Vernetzte Geräte im Haushalt in Deutschland 2017 / Umfrage.* https://de.statista.com/prognosen/795586/umfrage-in-deutschland-zu-ueber-das-internet-steuerbaren-geraeten-im-haushalt. Version: 12.2017, Abruf: 01.04.2019

[3] Lueth, Knud L.: *Why it is called Internet of Things: Definition, history, disambiguation.* https://iot-analytics.com/internet-of-things-definition/. Version: 2014, Abruf: 04.04.2019

[4] Cole, Tim: *Interview with Kevin Ashton - inventor of IoT: Is driven by the users - SMART INDUSTRY.* https://www.smart-industry.net/interview-with-iot-inventor-kevin-ashton-iot-is-driven-by-the-users/. Version: 2018, Abruf: 05.04.2019

[5] Haller, Stephan ; Karnouskos, Stamatis ; Schroth, Christoph: The Internet of Things in an Enterprise Context. Version: 2009. http://dx.doi.org/10.1007/978-3-642-00985-3{_}2. In: Domingue, John (Hrsg.) ; Fensel, Dieter (Hrsg.) ; Traverso, Paolo (Hrsg.): *Future Internet - FIS 2008* Bd. 5468. Berlin : Springer, 2009. – DOI 10.1007/978–3–642–00985–3_2. – ISBN 978–3–642–00984–6, S. 14–28

[6] Madakam, Somayya: Internet of Things: Smart Things. In: *International Journal of Future Computer and Communication* 4 (2015), Nr. 4, S. 250–253. http://dx.doi.org/10.7763/IJFCC.2015.V4.395. – DOI 10.7763/IJFCC.2015.V4.395. – ISSN 20103751

[7] Team, Municall: *4 Fallbeispiele für die Umsetzung des Internet of Things.* https://blog.municall.de/iot_umsetzung. Version: 2018, Abruf: 12.08.2019

[8] WEGWEISER industrie: *IoT / IIoT Beispiele - Industrielles Internet der Dinge*. https://industrie-wegweiser.de/internet-der-dinge-iot/. Version: 2016, Abruf: 10.04.2019

[9] NATIONAL INTELLIGENCE COUNCIL: Six Technologies With Potential Impacts on US Interests Out to 2025. (2008). https://fas.org/irp/nic/disruptive.pdf, Abruf: 10.04.2019

[10] DOMINGUEZ, Alberto: *Interoperability in IoT; a key factor for its development*. https://pandorafms.com/blog/interoperability-in-iot/. Version: 2018, Abruf: 11.04.2019

[11] ATLAM, Hany ; WALTERS, Robert ; WILLS, Gary: Internet of Things: State-of-the-art, Challenges, Applications, and Open Issues. In: *International Journal of Intelligent Computing Research* 9 (2018), S. 928–938. http://dx.doi.org/10.20533/ijicr.2042.4655.2018.0112. – DOI 10.20533/ijicr.2042.4655.2018.0112

[12] ONEM2M: Functional Architecture. (2016). http://onem2m.org/images/files/deliverables/Release2/TS-0001-%20Functional_Architecture-V2_10_0.pdf, Abruf: 13.04.2019

[13] ITU-T: *Overview of the Internet of things: Recommendation ITU-T Y.2060*. https://www.itu.int/ITU-T/recommendations/rec.aspx?rec=11559&lang=en. Version: 2012, Abruf: 13.04.2019

[14] MINERVA, Roberto ; BIRU, Abyi ; ROTONDI, Domenico: Towards a definition of the Internet of Things (IoT). (2015). https://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Issue1_14MAY15.pdf, Abruf: 13.04.2019

[15] IOTWF ARCHITECTURE COMMITTEE: The Internet of Things Reference Model. (2014). http://cdn.iotwf.com/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.pdf, Abruf: 15.04.2019

[16] WU, Miao ; LU, Ting-Jie ; LING, Fei-Yang ; SUN, Jing ; DU, Huiying: Research on the architecture of Internet of Things. Version: 2010. http://dx.doi.org/10.1109/ICACTE.2010.5579493. In: IEEE (Hrsg.): *Conference: Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on Volume: 5* Bd. 5. 2010. – DOI 10.1109/ICACTE.2010.5579493, S. V5–484

[17] IoTWF: *The Internet of Things World Forum 2017*. https://www.iotwf.com/iotwf2017/about. Version: 2017, Abruf: 17.04.2019

[18] IoTWF Architecture Committee: A Proposed Internet of Things Reference Model. (2014). http://cdn.iotwf.com/resources/72/IoT_Reference_Model_04_June_2014.pdf, Abruf: 19.04.2019

[19] ITU-T: *ITU-T Recommendations by series*. https://www.itu.int/itu-t/recommendations/index.aspx?ser=Y. Version: 2019, Abruf: 26.08.2019

[20] Arış, Ahmet ; Oktuğ, Sema F. ; Voigt, Thiemo: Security of Internet of Things for a Reliable Internet of Services. Version: 2018. http://dx.doi.org/10.1007/978-3-319-90415-3{_}13. In: Ganchev, Ivan (Hrsg.) ; van der Mei, R. D. (Hrsg.) ; van den Berg, Hans (Hrsg.): *Autonomous control for a reliable internet of services* Bd. 10768. Cham : Springer, 2018. – DOI 10.1007/978–3–319–90415–3_13. – ISBN 978–3–319–90414–6, S. 337–370

[21] Andrea, Ioannis ; Chrysostomou, Chrysostomos ; Hadjichristofi, George: Internet of Things: Security vulnerabilities and challenges. In: IEEE (Hrsg.): *20th IEEE Symposium on Computers and Communication (ISCC)*. Piscataway, NJ : IEEE, 2015. – ISBN 978–1–4673–7194–0, S. 180–187

[22] Aziz, Tariq ; Haq, Ehsan-ul: Security Challenges Facing IoT Layers and its Protective Measures. In: *International Journal of Computer Applications* 179 (2018), Nr. 27, S. 31–35. http://dx.doi.org/10.5120/ijca2018916607. – DOI 10.5120/ijca2018916607

[23] Mpitziopoulos, Aristides ; Gavalas, Damianos ; Konstantopoulos, Charalampos ; Pantziou, Grammati: A survey on jamming attacks and countermeasures in WSNs. In: *IEEE Communications Surveys & Tutorials* 11 (2009), Nr. 4, S. 42–56. http://dx.doi.org/10.1109/SURV.2009.090404. – DOI 10.1109/SURV.2009.090404. – ISSN 1553–877X

[24] Puthal, Deepak ; Nepal, Surya ; Ranjan, Rajiv ; Chen, Jinjun: Threats to Networking Cloud and Edge Datacenters in the Internet of Things. In: *IEEE Cloud Computing* 3 (2016), Nr. 3, S. 64–71. http://dx.doi.org/10.1109/MCC.2016.63. – DOI 10.1109/MCC.2016.63

[25] Abdullah, Ibrahim ; Muntasir Rahman, Mohammad ; Chandra Roy, Mukul: Detecting Sinkhole Attacks in Wireless Sensor Network using Hop

Count. In: *International Journal of Computer Network and Information Security* 7 (2015), Nr. 3, S. 50–56. http://dx.doi.org/10.5815/ijcnis.2015.03.07. – DOI 10.5815/ijcnis.2015.03.07. – ISSN 20749090

[26] Newsome, James ; Shi, Elaine ; Song, Dawn ; Perrig, Adrian: *The Sybil Attack in Sensor Networks: Analysis & Defenses: Third International Symposium on Information Processing in Sensor Networks, April 26-27, 2004, Berkeley, California, USA*. New York N.Y. : Association for Computing Machinery, 2004 https://netsec.ethz.ch/publications/papers/newsome_shi_song_perrig_sybil.pdf. – ISBN 158113892X

[27] ITU-T: *X.200 : Information technology - Open Systems Interconnection - Basic Reference Model: The basic model*. https://www.itu.int/rec/T-REC-X.200-199407-I. Version: 1994, Abruf: 25.04.2019

[28] Khan, Rukhsar: *1. Netzwerktechnik 1*. https://www.airnet.de/cr1-gfe/de/html/index.xml. Version: 17.03.2015, Abruf: 25.04.2019

[29] Braden, Robert: *Requirements for Internet Hosts - Communication Layers*. https://tools.ietf.org/html/rfc1122. Version: 1989, Abruf: 26.04.2019

[30] Wendzel, Steffen: Grundlagen der Netzwerktechnik. Version: 2018. http://dx.doi.org/10.1007/978-3-658-22603-9{_}2. In: Wendzel, Steffen (Hrsg.): *IT-Sicherheit für TCP/IP- und IoT-Netzwerke*. Wiesbaden : Springer Vieweg, 2018. – DOI 10.1007/978–3–658–22603–9_2. – ISBN 978–3–658–22602–2, S. 7–77

[31] IBM: TCP/IP Tutorial and Technical Overview. (2006). https://www.redbooks.ibm.com/pubs/pdfs/redbooks/gg243376.pdf, Abruf: 01.05.2019

[32] Anderson, Mike ; Tech Briefs (Hrsg.): *Understanding Network Topology: A guided tour of interconnected networks*. https://www.techbriefs.com/component/content/article/tb/supplements/st/features/28536. Version: 2018, Abruf: 03.05.2019

[33] Seebo® Interactive LTD.: *IoT Connectivity for Industry 4.0 Explained: Navigating IoT Protocols*. https://www.seebo.com/iot-connectivity/. Version: 2019, Abruf: 03.05.2019

[34] Garzke, René ; Der Tagesspiegel (Hrsg.): *Die Telekom registriert bis zu 46 Millionen Cyberangriffe pro Tag*. https://www.tagesspiegel.de/wirtschaft/netzkriminalitaet-die-telekom-registriert-bis-zu-46-

millionen-cyberangriffe-pro-tag/24375956.html. Version: 2019, Abruf: 03.06.2019

[35] Kraif, Ursula (Hrsg.): *Duden - das große Fremdwörterbuch: Herkunft und Bedeutung der Fremdwörter*. 4., aktualisierte Aufl. Mannheim : Dudenverl., 2007 http://deposit.d-nb.de/cgi-bin/dokserv?id=2902254&prov=M&dok_var=1&dok_ext=htm. – ISBN 9783411041640

[36] Bundesamt für Sicherheit in der Informationstechnik: Leitfaden für die Informationssicherheitsrevision (IS-Revision). (2016). https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Sicherheitsberatung/Pentest_Webcheck/Leitfaden_Penetrationstest.pdf?__blob=publicationFile&v=10, Abruf: 02.06.2019

[37] Bundesamt für Justiz ; Bundesministerium der Justiz und für Verbraucherschutz (Hrsg.): *§ 202a StGB - Einzelnorm*. https://www.gesetze-im-internet.de/stgb/__202a.html. Version: 23.05.2019, Abruf: 23.05.2019

[38] Cole, Eric: *Hackers beware*. 1st ed. Indianapolis Ind. : New Riders, 2002 https://doc.lagout.org/security/Hackers%20Beware.pdf. – ISBN 0735710090

[39] Giles, Lionel: *Sun Tzu on the Art of War: THE OLDEST MILITARY TREATISE IN THE WORLD: Translated from the Chinese*. Allandale Online Publishing, 2000 https://sites.ualberta.ca/~enoch/Readings/The_Art_Of_War.pdf. – ISBN 1–903328–03–9

[40] Hamburg, Daniel: *IT-Security: Schwachstellenanalyse vs. Penetrationstest*. https://www.computerwoche.de/a/schwachstellenanalyse-vs-penetrationstest,3068790. Version: 2014, Abruf: 11.06.2019

[41] Luber, Stefan ; Schmitz, Peter: *Definiton Pentest: Was ist ein Penetrationstest?* https://www.security-insider.de/was-ist-ein-penetrationstest-a-667683/. Version: 2017, Abruf: 11.06.2019

[42] Common Criteria for Information Technology Security Evaluation: Part 3: Security assurance components: CCMB-2017-04-003. (2017), Nr. Version 3.1 - Revision 5. https://www.commoncriteriaportal.org/files/ccfiles/CCPART3V3.1R5.pdf, Abruf: 15.06.2019

[43] Rouse, Margaret: *Definiton Angriffsvektor*. https://www.computerweekly.com/de/definition/Angriffsvektor. Version: 2016, Abruf: 17.06.2019

[44] Microsoft Corporation: *IoT-Sicherheitsarchitektur: Richtlinien und wichtige Aspekte*. https://docs.microsoft.com/de-de/azure/iot-fundamentals/iot-security-architecture. Version: 2018, Abruf: 17.06.2019

[45] Liew, Chun: *The Smart Home radio protocols war - IoT Now - How to run an IoT enabled business*. https://www.iot-now.com/2015/08/10/35653-the-smart-home-radio-protocols-war/. Version: 2015, Abruf: 19.06.2019

[46] Olawumi, Olayemi ; Haataja, Keijo ; Asikainen, Mikko ; Vidgren, Niko ; Toivanen, Pekka: Three practical attacks against ZigBee security: Attack scenario definitions, practical experiments, countermeasures, and lessons learned. In: IEEE (Hrsg.): *2014 14th International Conference on Hybrid Intelligent Systems (HIS)*. Piscataway, NJ : IEEE, 2014. – ISBN 978–1–4799–7633–1, S. 199–206

[47] Hussain, Abid ; Saqib, Nazar A. ; Qamar, Usman ; Zia, Muhammad ; Mahmood, Hassan: Protocol-aware radio frequency jamming in Wi-Fi and commercial wireless networks. In: *Journal of Communications and Networks* 16 (2014), Nr. 4, S. 397–406. http://dx.doi.org/10.1109/JCN.2014.000069. – DOI 10.1109/JCN.2014.000069. – ISSN 1229–2370

[48] Pöttner, Wolf-Bastian ; Wolf, Lars: IEEE 802.15.4 packet analysis with Wireshark and off-the-shelf hardware. (2010). https://www.ibr.cs.tu-bs.de/papers/poettner-inss2010-sniffer.pdf, Abruf: 21.06.2019

[49] Meier, J. D. ; Mackman, Alex ; Dunner, Michael ; Vasireddy, Srinath ; Escamilla, Ray ; Murukan, Anandha ; Microsoft Corporation (Hrsg.): *Threat Modeling*. https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff648644(v=pandp.10). Version: 2010, Abruf: 22.06.2019

[50] Eckert, Claudia: *IT-Sicherheit: Konzepte - Verfahren - Protokolle*. 9. München : Oldenbourg Wissenschaftsverlag GmbH, 2014. – ISBN 978–3–486–77848–9

[51] Boyce, Robert: Vulnerability Assessments: The Pro-active Steps to Secure Your Organization. (2001). https://www.sans.org/reading-

room/whitepapers/threats/vulnerability-assessments-pro-active-steps-secure-organization-453, Abruf: 26.06.2019

[52] Bundesamt für Sicherheit in der Informationstechnik: BSI - Studie Penetrationstests. (2003). https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Studien/Penetrationstest/penetrationstest.pdf?__blob=publicationFile&v=3, Abruf: 16.05.2019

[53] SmartHomeDB: *Smart Home Products*. https://www.smarthomedb.com/. Version: 2019, Abruf: 24.06.2019

[54] Blendien, Ken: *Schwachstellenanalyse kryptografischer Verfahren am Beispiel von WPA-2 und WPA-3*. Wismar, Hochschule Wismar, Diss., 06.06.2019

[55] Zigbee Alliance: *The Zigbee Alliance Celebrates 15 Years and A Decade of Standards*. https://zigbee.org/the-zigbee-alliance-celebrates-15-years-and-a-decade-of-standards/. Version: 2017, Abruf: 28.06.2019

[56] Zigbee Alliance: *Zigbee 3.0*. https://zigbee.org/zigbee-for-developers/zigbee-3-0/. Version: 2018, Abruf: 28.06.2019

[57] Zigbee Alliance: *ZigBee Specification*. https://zigbee.org/download/zigbee-pro-2015-spec/?wpdmdl=8451&refresh=5d6eda13c2e341567545875. Version: 2015, Abruf: 29.06.2019

[58] Shams, Soodi: *ADLT - Mesh Network Lighting Control System - Zigbee*. https://adlt.com.au/mesh-networks-for-lighting-control/. Version: 2016, Abruf: 29.06.2019

[59] Zigbee Alliance: *Zigbee Light Link*. https://zigbee.org/zigbee-for-developers/applicationstandards/zigbee-light-link/. Version: 2014, Abruf: 02.07.2019

[60] Zigbee Alliance: *Application Standards*. http://zigbee.org/zigbee-for-developers/applicationstandards/. Version: 2014, Abruf: 02.07.2019

[61] Zigbee Alliance: ZigBee Cluster Library Specification. (2016). http://www.zigbee.org/wp-content/uploads/2014/10/07-5123-06-zigbee-cluster-library-specification.pdf, Abruf: 04.07.2019

[62] NXP Semiconductors: ZigBee 3.0 Stack User Guide. (2018). https://www.nxp.com/docs/en/user-guide/JN-UG-3113.pdf, Abruf: 05.07.2019

[63] Zigbee Alliance: *Zigbee: Securing the Wireless IoT*. https://zigbee.org/download/securingthe-wireless-iot/?wpdmdl=7248&refresh=5d7209cb350f71567754699. Version: 2017, Abruf: 06.07.2019

[64] NXP Laboratories UK: ZigBee 3.0 Devices User Guide. (2016). https://www.nxp.com/docs/en/user-guide/JN-UG-3114.pdf, Abruf: 08.07.2019

[65] Farahani, Shahin: *ZigBee wireless networks and transceivers*. Amsterdam and Boston : Newnes/Elsevier, 2008 http://www.chiaraburatti.org/uploads/teaching/ZigBee-Libro.pdf. – ISBN 0750683937

[66] Zillner, Tobias: ZigBee Exploited - The Good, the Bad and the Ugly. (2015). https://www.blackhat.com/docs/us-15/materials/us-15-Zillner-ZigBee-Exploited-The-Good-The-Bad-And-The-Ugly-wp.pdf, Abruf: 08.07.2019

[67] Guzmann, Aaron ; Gupta, Aditya: *IoT Penetration Testing Cookbook*. Birmingham : Packt Publishing Ltd., 2017. – ISBN 978–1–78728–057–1

[68] Yang, Qing ; Huang, Lin: *Inside Radio: An Attack and Defense Guide*. Beijing : Publishing House of Electronics Industry, 2018. – ISBN 978–981–10–8446–1

[69] Cao, Xianghui ; Shila, Devu M. ; Cheng, Yu ; Yang, Zequ ; Zhou, Yang ; Chen, Jiming: Ghost-in-ZigBee: Energy Depletion Attack on ZigBee-Based Wireless Networks. In: *IEEE Internet of Things Journal* 3 (2016), Nr. 5, 816–829. http://dx.doi.org/10.1109/JIOT.2016.2516102, Abruf: 09.09.2019. – DOI 10.1109/JIOT.2016.2516102

[70] Grün, Frank-Oliver ; digitalzimmer (Hrsg.): *ZigBee 3.0 kommt - was bedeutet das?* https://www.digitalzimmer.de/artikel/wissen/zigbee-3-0-was-bedeutet-das/. Version: 2018

[71] Zigbee Alliance: *ZigBee Light Link Standard*. https://zigbee.org/download/standard-zigbee-light-link/?wpdmdl=2132&refresh=5d776a6d99cf01568107117. Version: 2012

[72] Morgner, Philipp ; Mattejat, Stephan ; Benenson, Zinaida ; Müller, Christian ; Armknecht, Frederik: Insecure to the touch. In: Noubir, Guevara (Hrsg.): *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. New York : Association for Computing Machinery, July 2017. – ISBN 9781450350846, 230–240

[73] M45T3R 0F H4RDC0R3S: *ZigBee ZLL master key für Philips Hue*. http://moh-computer.de/zigbee-zll-master-key-fuer-philips-hue/. Version: 2015, Abruf: 10.07.2019

[74] JACKSON, Chris: A Deep Dive into Z-Wave. (2017). https://www.openhabfoundation.org/documents/2017-10_Chris_Jackson_A_Deep_Dive_into_Z-Wave.pdf, Abruf: 12.07.2019

[75] BADENHOP, Christopher W. ; GRAHAM, Scott R. ; RAMSEY, Benjamin W. ; MULLINS, Barry E. ; MAILLOUX, Logan O.: The Z-Wave routing protocol and its security implications. In: *Computers & Security* 68 (2017), 112–129. http://dx.doi.org/10.1016/j.cose.2017.04.004. – DOI 10.1016/j.cose.2017.04.004. – ISSN 0167–4048

[76] RF WIRELESS WORLD: *z-wave tutorial | z-wave basics | tutorial section*. https://www.rfwireless-world.com/Tutorials/z-wave-tutorial.html. Version: 2019, Abruf: 14.07.2019

[77] ITU-T: *G.9959 : Short range narrow-band digital radiocommunication transceivers - PHY, MAC, SAR and LLC layer specifications*. https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-G.9959-201501-I!!PDF-E&type=items. Version: 2015, Abruf: 15.07.2019

[78] BADENHOP, Christopher ; FULLER, Jonathan ; HALL, Joseph ; RAMSEY, Benjamin ; RICE, Mason: Evaluating ITU-T G.9959 Based Wireless Systems Used in Critical Infrastructure Assets. Version: 2015. http://dx.doi.org/10.1007/978-3-319-26567-4{_}13. In: RICE, Mason (Hrsg.) ; SHENOI, Sujeet (Hrsg.): *Critical infrastructure protection IX* Bd. 466. Cham : Springer, 2015. – DOI 10.1007/978–3–319–26567–4_13. – ISBN 978–3–319–26566–7, 209–227

[79] VESTERNET: *How Z-Wave Controllers & Devices Work*. https://www.vesternet.com/pages/how-z-wave-controllers-devices-work. Version: 2012, Abruf: 16.09.2019

[80] KOHRS, Kevin: Introduction to Z-Wave: An Introductory Guide to Z-Wave Technology. (2013). http://library.ademconet.com/MWT/fs2/L5210/Introductory-Guide-to-Z-Wave-Technology.pdf, Abruf: 15.07.2019

[81] ANIMUS HOME TEAM: *Understanding S0/S2 and the newly discovered Z-Wave hack ("Z-shave")*. https://blog.animushome.com/2018/05/

30/understanding-s0s2-and-the-newly-discovered-z-wave-hack-z-shave/. Version: 2018, Abruf: 28.07.2019

[82] ABR: Introduction to the Z-Wave Security Ecosystem. (2016). https://cdn.shopify.com/s/files/1/0066/8149/3559/files/z-wave-security-white-paper.pdf, Abruf: 01.08.2019

[83] Tierney, Andrew: *Z-Shave. Exploiting Z-Wave downgrade attacks | Pen Test Partners*. https://www.pentestpartners.com/security-blog/z-shave-exploiting-z-wave-downgrade-attacks/. Version: 2018, Abruf: 02.08.2019

[84] Hoskins, Katie: Security Vulnerabilities in Z-Wave Home Automation Protocol. (2016). http://www.cs.tufts.edu/comp/116/archive/fall2016/khoskins.pdf, Abruf: 02.08.2019

[85] Behrang, Fouladi ; Sahand, Ghanoun: Security Evaluation of the Z-Wave Wireless Protocol. (2013). https://sensepost.com/cms/resources/conferences/2013/bh_zwave/Security%20Evaluation%20of%20Z-Wave_WP.pdf, Abruf: 03.08.2019

[86] Buron, Jakob: *Security and backwards compatibility, and how we designed Z-Wave to maintain both*. https://www.embedded-computing.com/guest-blogs/security-and-backwards-compatibility-and-how-we-designed-z-wave-to-maintain-both. Version: 2018, Abruf: 05.08.2019

[87] Khandelwal, Swati: *Z-Wave Downgrade Attack Left Over 100 Million IoT Devices Open to Hackers*. https://thehackernews.com/2018/05/z-wave-wireless-hacking.html. Version: 2018, Abruf: 05.08.2019

[88] Bluetooth SIG, Inc.: *Bluetooth Core Specification: v5.1*. https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=457080. Version: 2019

[89] Lee, Lori: *Alibaba Builds a Smart Home Ecosystem with Bluetooth Mesh | Bluetooth Technology Website*. https://www.bluetooth.com/blog/alibaba-builds-a-smart-homeecosystem-with-bluetooth-mesh/. Version: 2019, Abruf: 01.09.2019

[90] Bluetooth SIG, Inc.: Bluetooth Market Update 2019. (2019). https://3pl46c46ctx02p7rzdsvsg21-wpengine.netdna-ssl.com/wp-content/uploads/2018/04/2019-Bluetooth-Market-Update.pdf, Abruf: 08.08.2019

[91] Bluetooth SIG, Inc.: *Radio Versions | Bluetooth Technology Website*. https://www.bluetooth.com/bluetooth-technology/radio-versions/. Version: 27.08.2019, Abruf: 10.08.2019

[92] TAIYO YUDEN: *The expanding application of Bluetooth® low energy*. https://www.yuden.co.jp/or/solutions/ble/. Version: 2019, Abruf: 11.08.2019

[93] Infotip Service GmbH: *Bluetooth - InfoTip Kompendium*. https://kompendium.infotip.de/bluetooth.html. Version: 2016, Abruf: 13.08.2019

[94] ITWissen: *Bluetooth-Protokollstack*. https://www.itwissen.info/Bluetooth-Protokollstack-Bluetooth-protocol-stack.html. Version: 2016, Abruf: 13.08.2019

[95] infotainmenttechnologynews: *Bluetooth Protocol stack/layers*. https://infotainmenttechnology.wordpress.com/2017/01/27/bluetooth-protocol-stacklayers/. Version: 2017, Abruf: 15.08.2019

[96] Goebel, Hartmut: *Scatternet*. https://commons.wikimedia.org/wiki/File:BluetoothScatternet-de.svg. Version: 2019, Abruf: 15.08.2019

[97] Elektronik Kompendium: *Bluetooth Mesh*. https://www.elektronik-kompendium.de/sites/kom/2210201.htm. Version: 2019, Abruf: 15.08.2019

[98] Digi-Key: *Lösungen für Bluetooth Mesh: 1. Teil | DigiKey*. https://www.digikey.de/de/articles/techzone/2018/mar/designing-bluetooth-low-energy-smart-applications-part-1. Version: 2018, Abruf: 17.08.2019

[99] Hodgdon, Charles ; Ericsson Technology Licensing AB (Hrsg.): *Adaptive Frequency Hopping for Reduced Interference between Bluetooth® and Wireless LAN*. https://www.design-reuse.com/articles/5715/adaptive-frequency-hopping-for-reduced-interference-between-bluetooth-and-wireless-lan.html. Version: 2003, Abruf: 17.08.2019

[100] Padgette, John ; Bahr, John ; Batra, Mayank ; Holtmann, Marcel ; Smithbey, Rhonda ; Chen, Lily ; Scarfone, Karen: *Guide to Bluetooth Security*. http://dx.doi.org/10.6028/NIST.SP.800-121r2

[101] Wells, Christopher J.: *Bluetooth*. http://www.technologyuk.net/telecommunications/communication-technologies/bluetooth.shtml. Version: 2009, Abruf: 18.08.2019

[102] Ren, Kai: *Bluetooth Pairing Part 1 -Pairing Feature Exchange | Bluetooth Technology Website*. https://www.bluetooth.com/blog/bluetooth-pairing-part-1-pairing-feature-exchange/. Version: 2016, Abruf: 19.08.2019

[103] Bräuer, Sebastian ; Zubow, Anatolij ; Zehl, Sven ; Roshandel, Mehran ; Mashhadi-Sohi, Soroush: *On Practical Selective Jamming of Bluetooth LowEnergy Advertising*. Piscataway, NJ : IEEE, 2016 https://www2.informatik.hu-berlin.de/~zubow/ble_jamming_cscn.pdf. – ISBN 9781509038633

[104] Ali, Shakeel ; Heriyanto, Tedi: BackTrack 4: Assuring Security by Penetration Testing: Chapter No. 2 - "Penetration Testing Methodology". (2011). https://pdfs.semanticscholar.org/fdf6/3f03483aec82b3540ec681241111eb0b4c36.pdf, Abruf: 20.08.2019

[105] Attify: *AttifyOS - Distro to assess the security of IoT Devices — Attify IoT Security and Penetration Testing Training*. https://www.attify.com/attifyos. Version: 2017, Abruf: 17.09.2019

[106] Attify: *APIMote (for ZigBee sniffing and transmission)*. https://www.attify-store.com/collections/frontpage/products/apimote-for-zigbee-sniffing-and-transmission. Version: 2019

[107] AVAST Software s.r.o.: *Was ist Ransomware und wie entfernt man sie | Avast*. https://www.avast.com/de-de/c-ransomware. Version: 2016, Abruf: 21.01.2019

[108] Bundesamt für Sicherheit in der Informationstechnik: *BSI - Radio Frequency Identification*. https://www.bsi.bund.de/DE/Themen/DigitaleGesellschaft/RadioFrequencyIdentification/radiofrequencyidentification_node.html. Version: 2019, Abruf: 12.08.2019

[109] Czagan, Dawid: *Qualitative Risk Analysis with the DREAD Model*. https://resources.infosecinstitute.com/qualitative-risk-analysis-dread-model/#gref. Version: 2014, Abruf: 02.07.2019

[110] Domingue, John (Hrsg.) ; Fensel, Dieter (Hrsg.) ; Traverso, Paolo (Hrsg.): *Lecture Notes in Computer Science*. Bd. 5468: *Future Internet - FIS 2008: First Future Internet Symposium, FIS 2008, Vienna, Austria, September 29 - 30, 2008 ; revised selected papers*. Berlin : Springer, 2009.

http://dx.doi.org/10.1007/978-3-642-00985-3. http://dx.doi.org/10.1007/978-3-642-00985-3. – ISBN 978–3–642–00984–6

[111] Fields Associates Ltd. (Hrsg.): *Black Box v/s White Box Testing | Fields Penetration Testing*. http://www.fields-penetrationtesting.co.uk/black_box_white_box.html, Abruf: 10.06.2019

[112] Ganchev, Ivan (Hrsg.) ; van der Mei, R. D. (Hrsg.) ; van den Berg, Hans (Hrsg.): *Lecture Notes in Computer Science*. Bd. 10768: *Autonomous control for a reliable internet of services: Methods, models, approaches, techniques, algorithms, and tools*. Cham : Springer, 2018. http://dx.doi.org/10.1007/978-3-319-90415-3. http://dx.doi.org/10.1007/978-3-319-90415-3. – ISBN 978–3–319–90414–6

[113] GreyCampus: *Sniffing and its Types*. https://www.greycampus.com/opencampus/ethical-hacking/sniffing-and-its-types. Version: 2019, Abruf: 25.06.2019

[114] Hagel, Jens: *Was ist ein Portscan?* https://www.hagel-it.de/it-service/was-ist-ein-portscan.html. Version: 2015, Abruf: 25.06.2019

[115] ID, F. C.: *FCC ID Search*. http://fccid.io/. Version: 2019, Abruf: 19.07.2019

[116] IEEE (Hrsg.): *2014 14th International Conference on Hybrid Intelligent Systems (HIS): 14 - 16 Dec. 2014, [Hawally], Kuwait*. Piscataway, NJ : IEEE, 2014 . – ISBN 978–1–4799–7633–1

[117] IEEE (Hrsg.): *Conference: Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on Volume: 5*. 2010

[118] IEEE (Hrsg.): *20th IEEE Symposium on Computers and Communication (ISCC): 20th IEEE Symposium on Computers and Communication (ISCC) took place 6-9 July 2015 in Lanarca, Cyprus*. Piscataway, NJ : IEEE, 2015 . – ISBN 978–1–4673–7194–0

[119] ITWissen: *DMZ (demilitarized zone)*. https://www.itwissen.info/DMZ-demilitarized-zone-Demilitarisierte-Zone.html. Version: 2010, Abruf: 23.06.2019

[120] Mahmood, Haider: *Application Threat Modeling using DREAD and STRIDE*. https://haiderm.com/application-threat-modeling-using-dread-and-stride/. Version: 2017, Abruf: 14.07.2019

[121] Melgares, Ricky: *IEEE 802.15.4/ZigBee Security Research Toolkit*. https://github.com/riverloopsec/killerbee. Version: 2018, Abruf: 05.09.2019

[122] Microsoft Corporation: *The STRIDE Threat Model*. https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20). Version: 2009, Abruf: 02.07.2019

[123] mikrocontroller.net: *AVR Raven – Mikrocontroller.net*. https://www.mikrocontroller.net/articles/AVR_Raven. Version: 2010, Abruf: 06.09.2019

[124] Noubir, Guevara (Hrsg.): *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. New York : Association for Computing Machinery, July 2017 . – ISBN 9781450350846

[125] Rice, Mason (Hrsg.) ; Shenoi, Sujeet (Hrsg.): *IFIP Advances in Information and Communication Technology*. Bd. 466: *Critical infrastructure protection IX: 9th IFIP 11.10 international conference, ICCIP 2015, Arlington, VA, USA, March 16-18, 2015 : revised selected papers / Mason Rice, Sujeet Shenoi (eds.)*. Cham : Springer, 2015. http://dx.doi.org/10.1007/978-3-319-26567-4. http://dx.doi.org/10.1007/978-3-319-26567-4. – ISBN 978–3–319–26566–7

[126] Rouse, Margaret ; Kramer, David: *Definition Buffer Overflow*. https://www.computerweekly.com/de/definition/Buffer-Overflow. Version: 2016, Abruf: 23.06.2019

[127] Rouse, Margaret: *Definition Denial of Service (DoS)*. https://www.computerweekly.com/de/definition/Denial-of-Service-DoS. Version: 2013, Abruf: 23.06.2019

[128] Rouse, Margaret ; Rosencrance, Linda: *Definition IP Spoofing*. https://searchsecurity.techtarget.com/definition/IP-spoofing. Version: 2018, Abruf: 24.06.2019

[129] Rouse, Margaret: *Definition Trojaner*. https://www.computerweekly.com/de/definition/Trojaner. Version: 2006, Abruf: 25.06.2019

[130] Saindane, Manish S.: Penetration testing – A Systematic Approach. (2015). http://www.infosecwriters.com/Papers/MSaindane_Pentest.pdf, Abruf: 11.06.2019

[131] Schonscheck, Oliver: *IT-Security & Cyber-Risk-Versicherung: Unternehmen gegen Hacker versichern*. https://www.computerwoche.de/a/unternehmen-gegen-hacker-versichern,3213043,2. Version: 2015, Abruf: 02.06.2019

[132] Secondis GmbH: *Bedrohungsmodellierung (Threat Modelling) / Risk Assessments*. https://www.secodis.com/bedrohungsanalysen/. Version: 2019, Abruf: 02.07.2019

[133] Silicon Laboratories Inc.: UG103.2: ZigBee Fundamentals. (2019). https://www.silabs.com/documents/public/user-guides/ug103-02-fundamentals-zigbee.pdf, Abruf: 05.09.2019

[134] Stewart, Ryan: *Latest Bluetooth hacking techniques expose new attack vectors for hackers | Cyware Hacker News*. https://cyware.com/news/latest-bluetooth-hacking-techniques-expose-new-attack-vectors-for-hackers-a16cfb5e. Version: 2019, Abruf: 14.09.2019

[135] Texas Instruments: *CC2531EMK CC2531 USB Evaluation Module Kit | TI.com*. http://www.ti.com/tool/CC2531EMK#. Version: 2009, Abruf: 12.09.2019

[136] Texas Instruments: *SmartRF Protocol Packet Sniffer*. http://www.ti.com/tool/PACKET-SNIFFER. Version: 2019, Abruf: 17.09.2019

[137] Wendzel, Steffen (Hrsg.): *IT-Sicherheit für TCP/IP- und IoT-Netzwerke: Grundlagen, Konzepte, Protokolle, Härtung*. Wiesbaden : Springer Vieweg, 2018. http://dx.doi.org/10.1007/978-3-658-22603-9. http://dx.doi.org/10.1007/978-3-658-22603-9. – ISBN 978–3–658–22602–2

[138] Wikipedia (Hrsg.): *Common Criteria for Information Technology Security Evaluation*. https://de.wikipedia.org/w/index.php?oldid=186459182. Version: 09.06.2019, Abruf: 15.06.2019

[139] Wikipedia (Hrsg.): *Federal Communications Commission*. https://de.wikipedia.org/w/index.php?oldid=189592257. Version: 2019, Abruf: 19.07.2019

[140] Wikipedia (Hrsg.): *Intrusion Detection System*. https://de.wikipedia.org/w/index.php?oldid=188067782. Version: 17.06.2019, Abruf: 23.06.2019

[141] Wikipedia (Hrsg.): *Smart Home*. https://de.wikipedia.org/w/index.php?oldid=184678143. Version: 13.01.2019, Abruf: 13.01.2019

Abbildungsverzeichnis

1.1 Vernetzte Geräte im Haushalt in Deutschland 2017, gekürzte Fassung 8

2.1 Definition Internet der Dinge 11
2.2 IdD-Referenzmodell nach dem IoTWF Architekturausschuss 14
2.3 IdD-Referenzmodell nach der ITU-T Recommendation Y.2060 14
2.4 IdD-„High-Level-Architektur“ 15
2.5 OSI-Referenzmodell nach der ITU-T Recommendation X.200 21
2.6 TCP/IP-Referenzmodell nach dem IETF RFC 1122 im Vergleich mit dem OSI-Referenzmodell 24
2.7 Netzwerktopologien im IdD-Umfeld 26
2.8 StGB: §202a Ausspähen von Daten 28
2.9 Definition Penetrationstest 29

3.1 Schwachstellenanalyse in Verbindung mit dem Prozess des Penetrationstests 36

4.1 Statistik der genutzten Protokoll in Smart Home Geräten 38

5.1 ZigBee Protokollstack 42
5.2 ZigBee Referenzmodell und Einordnung in das TCP/IP-Modell 44
5.3 ZigBee logische Geräte nach der ZDO Definition 44
5.4 ZigBee Netzwerktopologien 46
5.5 ZigBee Centralized Security versus Distributed Security 49
5.6 ZigBee Frame mit Sicherheit auf der NWK Ebene 51
5.7 ZigBee Frame mit Sicherheit auf der APS Ebene 51
5.8 ZigBee Auxiliary Header Format 51
5.9 ZigBee Daten Authentifizierung mittels MIC 53
5.10 ZigBee Auxiliary Header Format und die CCM* Nonce 54
5.11 ZigBee Secure Network Join Prozedur 56
5.12 ZigBee Übertragung des Network Key beim Network Join 60
5.13 ZigBee Ablauf eines Replay Angriffs 61
5.14 ZigBee Insecure Rejoin Prozedur 63
5.15 ZigBee Touchlink Commissioning 64
5.16 ZigBee Resetting a device to factory new 66
5.17 Wireshark mit geöffneter out.dump 70
5.18 Philips Hue App für iPhone 71

6.1 Z-Wave Referenzmodell und Einordnung in das TCP/IP-Modell . . . 75
6.2 Z-Wave Frames 77
6.3 Z-Wave Mesh-Topologie mit Controller und Slaves 78

6.4 Z-Wave Routing 80
6.5 Z-Wave S0 und S2 Key Exchange 84
6.6 Z-Wave Node Info eines S2 Gerätes 85
6.7 Z-Wave S2 Downgrade via Timing 86
6.8 Z-Wave S2 Downgrade via Jamming 86

7.1 Bluetooth Nutzungsbeispiele 91
7.2 Bluetooth Protokollstack 92
7.3 Bluetooth Piconet mit einer aktiven Verbindung 94
7.4 Bluetooth Piconet mit der maximalen Anzahl an aktiven Verbindungen 94
7.5 Bluetooth Piconet mit passiven Slaves 95
7.6 Bluetooth Scatternet mit aktiven und passiven Slaves 95
7.7 Bluetooth Mesh 96
7.8 Bluetooth Frequenzspektrum mit Low Energie-Kanalaufteilung 96
7.9 Bluetooth Funkübertragung mittels FHSS 98
7.10 Bluetooth Basic Rate Packet Format 98
7.11 Bluetooth Header Format 99
7.12 Bluetooth Enhanced Data Rate Packet Format 100
7.13 Bluetooth Low Energy Packet Format 100
7.14 Bluetooth Device Address 103
7.15 Bluetooth Link Key Generierung mit PIN 108
7.16 Bluetooth SSP/Legacy Pairing und Secure Connections 111
7.17 Bluetooth Authentication Request 113
7.18 Bluetooth Proposed Jammer 114
7.19 Bluetooth LE Advertising 116

A.1 Klassifikation von Penetrationstests 148
A.2 Schlüsselkonzept für die Durchführung einer Schwachstellenanalyse . 155
A.3 Phasen eines Penetrationstests 156

B.1 APIMote (for ZigBee sniffing and transmission) 158
B.2 CC2531 USB Dongle 159
B.3 AVR Raven 159

Listings

5.1 KillerBee zbid Befehl 68
5.2 KillerBee zbstumbler Befehl 68
5.3 KillerBee zbdump Befehl 69
5.4 KillerBee zbconvert Befehl 70
5.5 KillerBee zbreplay Befehl 70

Tabellenverzeichnis

2.1 Bewertungstabelle des „DREAD“-Modells zur Priorisierung von Bedrohungen 34

3.1 Klassifizierung der Vorgehensweise für die Phase der aktiven Eindringversuche 37

5.1 Bewertung dargestellter Angriffsvektoren von ZigBee 67

6.1 Bewertung dargestellter Angriffsvektoren von Z-Wave 87

7.1 Bewertung dargestellter Angriffsvektoren von Bluetooth 115

Glossar

Common Criteria for Information Technology Security Evaluation (CC) Internationaler Standard zur Prüfung und Bewertung der Sicherheitseigenschaften von IT-Produkten (vgl. [138]).

Demilitarisierte Zone (DMZ) Firewall-Technologie zum Schutz eigener Netzwerke gegenüber öffentlich zugänglichen Netzwerken, sowie zum Schutz von firmeninternen Systemen, die sowohl von öffentlichen Netzwerken als auch aus dem internen Unternehmensnetzwerk zugänglich sein müssen (vgl. [119]).

Denial of Service (DoS) Beeinträchtigt die Funktionalität von Diensten oder Servern und schränkt die Verfügbarkeit dieser für Benutzer und Unternehmen ein (vgl. [127]).

Intrusion Detection System (IDS) System zur Erkennung von Angriffen, die sich gegen Computersysteme oder Netzwerke richten (vgl. [140]).

Internet der Dinge (IdD) Kann als eine Erweiterung des Internets und anderer Netzwerkverbindungen zu verschiedenen Sensoren und Geräten - oder „Dingen“ - beschrieben werden, die selbst einfachen Objekten wie Glühbirnen, Schlössern und Lüftern ein höheres Maß an Rechen- und Analysefähigkeiten bieten.

Radio Frequency Identification (RFID) Verfahren zur automatischen Identifizierung von Objekten per Funk. Der Einsatz von RFID-Systemen ist grundsätzlich überall dort sinnvoll, wo eine automatische Identifikation, Erkennung, Registrierung, Lagerung, Überwachung oder Transport erforderlich ist (vgl. [108]).

Target of Evaluation (TOE) Prüfgegenstand bei IT-Sicherheitsrichtlinien.

Buffer-Overflow Liegt dann vor, wenn ein Programm oder Prozess versucht, mehr Daten in einen Puffer (temporärer Datenspeicher) zu speichern, als im Puffer vorhanden ist. Da Puffer für eine begrenzte Datenmenge konzipiert sind, fließen überschüssige Daten, die gespeichert werden müssen, in angrenzende Puffer. In diesem Fall werden die dort abgelegten Daten überschrieben und beschädigt (vgl. [126]).

IP-Spoofing Erstellen und versenden von IP-Paketen mit gefälscher Quell-IP-Adresse. Hierdurch wird die Identität des Absenders verborgen beziehungsweise eine fremde Identität übernommen (vgl. [128]).

Port-Scanning Gezielter Versuch, offene Ports und damit Dienste in IT-Systemen zu eruieren (vgl. [114]).

Ransomware Auch als Scareware bezeichnet. Schränkt den Zugriff auf ein Computersystem ein und verlangt die Zahlung eines Lösegelds, damit die Einschränkung wieder behoben wird (vgl. [107]).

Smart Home Dient als Oberbegriff für [...] Systeme in Wohnräumen und -häusern, in deren Mittelpunkt eine Erhöhung von Wohn- und Lebensqualität, Sicherheit und effizienter Energienutzung auf Basis vernetzter und fernsteuerbarer Geräte [...] sowie automatisierbarer Abläufe steht (vgl. [141]).

Sniffing Prozess der Überwachung und Erfassung aller Datenpakete, die durch ein bestimmtes Netzwerk laufen. Hacker nutzen Sniffer, um Datenpakete mit sensiblen Informationen wie Passwort, Kontoinformationen usw. zu empfangen (vgl. [113]).

Trojaner Programm, in das schädlicher beziehungsweise bösartiger Code eingebettet ist. Wenngleich das Programm für die Außenwelt als harmlos erscheinen mag, so ist es in der Lage, die Kontrolle über den Computer zu erlangen und schwere Schäden zu verursachen (vgl. [129]).

Abkürzungsverzeichnis

ACL Asynchronous Connection-Oriented Logical.

AES Advanced Encryption Standard.

AFH Adaptive Frequency Hopping Spread Spectrum.

AMP Alternate MAC und Physical Layer Erweiterungen.

APL Application Layer.

APS Application Support Sub-Layer.

APSDE APS Data Entity.

APSME APS Management Entity.

ARQN Automatic Repeat Request Scheme.

BLE Bluetooth Low Energy.

BR Basic Rate.

BSI Bundesamt für Sicherheit in der Informationstechnik.

CC Common Criteria for Information Technology Security Evaluation.

CRC Cyclic Redundancy Check.

CSMA/CA Carrier Sense Multiple Access/Collision Avoidance.

DCF Daintree Capture File.

DDoS Distributed Denial of Service.

DH Diffie-Hellman.

DMZ Demilitarisierte Zone.

DoS Denial of Service.

DPSK Differential Phase-Shift Keying.

ECDH Elliptic Curve Diffie-Hellman.

EDR Enhanced Data Rate.

EHR End Header.

ESCE External Security Control Entity.

eSCO Enhanced SCO.

EUI-48 48-Bit Extendend Unique Identifier.

FCC Federal Communications Commission.

FEC Forward Error Correction.

FHSS Frequency Hopping Spread Spectrum.

FIPS Federal Information Processing Standard.

FLiRS Frequently Listening Routing Slave.

FSK Frequency Shift Keying.

GFSK Gaussian Frequency Shift Keying.

GPS Global Positioning System.

HEC Header-Error-Check.

ICACTE International Conference on Advanced Computer Theory and Engineering.

IdD Internet der Dinge.

IDS Intrusion Detection System.

IEEE Institute of Electrical and Electronics Engineers.

IEEE-SA IEEE Standards Association.

IETF Internet Engineering Task Force.

IoT Internet of Things.

IoTWF IoT World Forum.

IP Internet Protocol.

ISCC IEEE Symposium on Computers and Communication.

ISM Industrial, Scientific and Medical.

ISO International Organization for Standardization.

ITU International Telecommunication Union.

ITU-T ITU - Telecommunication Standardization Sector.

L2CAP Logical Link Control and Adaption Protocol.

LAN Local Area Network.

LAP Lower Address Part.

LC Link Control.

LE Low Energy.

LMP Link Manager Protocol.

LTK Long Term Key.

MAC Medium Access Control.

MFR MAC Footer.

MHR MAC Header.

MIC Message Integrity Code.

MIT Massachusetts Institute of Technology.

MLME Medium Access Control Sub-Layer Management Entity.

MPDU MAC Protocol Data Unit.

MSDU MAC Service Data Unit.

NAP Non-significant Address Part.

NDA Non-Disclosure Agreement.

NIC National Intelligence Council.

NIST National Institute of Standards and Technology.

NLDE Network Layer Data Entity.

NLME Network Layer Management Entity.

NPDU Network Level PDU.

NRZ Non Return to Zero.

NWK Network Layer.

OBEX Object Exchange Protocol.

OOB Out of Band.

OSI Open Systems Interconnection.

PAN Personal Area Network.

PDU Protocol Data Unit.

PIN Personal Identification Number.

PPDU Physical Protocol Data Unit.

PPP Point-to-Point Protocol.

PSDU Physical Service Data Unit.

RFC Request for Comments.

RFCOMM Radio Frequency Communication.

RFID Radio Frequency Identification.

RTC Real-Time Clock.

SCO Synchronous Connection Oriented.

SDP Service Discovery Protocol.

SEQN Sequential Numbering Scheme.

SFD Start Frame Delimiter.

SHR Start Header.

SIG Bluetooth Special Interest Group.

SPAN Singlecast Pre-Agreed Nonce.

SRD Short Range Devices.

SSP Secure Simple Pairing.

StGB Strafgesetzbuch.

STK Short Term Key.

SVP Senior Vice President.

TC Trust Center.

TCP Transmission Control Protocol.

TCS Telephony Control Specification Binary.

TDD Time Division Duplex.

TOE Target of Evaluation.

UAP Upper Address Part.

UDP User Datagram Protocol.

WAP Wireless Application Protocol.

WPA Wi-Fi Protected Access.

WPAN Wireless Personal Area Network.

WSN Wireless Sensor Network.

ZDO ZigBee Device Object.

ZLL ZigBee Light Link.

A Penetrationstests

A.1 Ziele von Penetrationstests

Orientiert man sich an der im Kapitel 2.4.1 abgeleiteten Definition (vgl. Abbildung 2.9) eines Penetrationstest, so bietet dieser einen Überblick des aktuellen Sicherheitsstatus eines – allgemein ausgedrückt – Betrachtungsobjektes. Die Absicht eines solchen Tests ist es, die Machbarkeit eines Angriffs und die Auswirkungen eines erfolgreichen Angriffs festzustellen. Damit dieser erfolgreich durchgeführt werden kann und „den Erwartungen des Auftraggebers entspricht, ist eine klare Zielvereinbarung unbedingt notwendig“ (vgl. [52, 10]). In einer Konzept-Studie des BSI zur Durchführung von Penetrationstests werden die Ziele eines solchen Tests in die folgenden Gruppen eingeteilt (vgl. [52, 10 f.]):

- *Erhöhung der Sicherheit der technischen Systeme*
  Penetrationstests werden mit dem Ziel durchgeführt, die Sicherheit von IT-Systemen wie Firewalls, Routern und Servern zu verbessern. Die organisatorische aber auch personelle Infrastruktur wird hierbei nicht dediziert geprüft. Zum Schutz der Daten werden verschiedene Sicherheitsmechanismen wie Intrusion Detection System (IDS)s, Firewalls und Kryptographie eingesetzt. Die Häufigkeit und Schwere von Einbrüchen in das Netzwerk, Datendiebstahl und Angriffe durch bösartigen Code, Hacker, unzufriedene Mitarbeiter sowie die mit Sicherheitsverletzungen und Datendiebstahl verbundenen Risiken und Kosten nehmen jedoch weiter zu. Penetrationstests helfen, solche Probleme zu lösen. Als Beispiel ist hier das Eruieren nicht benötigter offene Ports oder verwundbarer Versionen von Webanwendungen und Betriebssystemen zu nennen (vgl. [52, 11]).
- *Identifikation von Schwachstellen*
  Die Identifizierung von Schwachstellen ist das eigentliche Ziel von Penetrationstests. Der Einsatz von solchen Tests hilft nicht nur, das Sicherheitsrisiko zu verstehen, sondern auch, Risikoprobleme zu priorisieren, zusammen mit einer Bewertung ihrer Auswirkungen und oft mit einem Vorschlag zur Minderung.

Das bei der Prüfung identifizierte Risiko kann nach Schweregrad priorisiert werden (vgl. [52, 11]). Außerdem können diese Bemühungen im Kontext eines Unternehmens zu einer effizienten Budgetzuweisung in Bezug auf Fragen der Informationssicherheit führen.

- *Bestätigung der IT-Sicherheit durch einen externen Dritten*
  Eine unvoreingenommene Sicherheitsanalyse und ein Penetrationstest können die internen Sicherheitsressourcen dort konzentrieren, wo sie am dringendsten benötigt werden. Darüber hinaus liefert ein unabhängiges Sicherheitsaudit den Nachweis der Sorgfaltspflicht im rechtlichen Kontext zum Schutz von Online-Vermögenswerten und minimiert so, in Bezug auf Unternehmen, den potenziellen Verlust des Unternehmenswertes. Diese unabhängigen Audits werden zunehmend zu einer Voraussetzung für den Abschluss einer Cybersicherheitsversicherung und binden Unternehmen als Versicherungsnehmer an Pflichten hinsichtlich der IT-Sicherheit (vgl. [131]).
- *Erhöhung der Sicherheit der organisatorischen und personellen Infrastruktur*
  Neben dem Testen der technischen Infrastruktur kann ein Penetrationstest auch die Management- und Mitarbeiterinfrastruktur testen, um beispielsweise Eskalationsverfahren zu überwachen, während der Umfang und/oder die Aggressivität der Tests Schritt für Schritt erhöht wird. Social Engineering-Techniken, wie z.B. die telefonische Anforderung von Passwörtern, können eingesetzt werden, um das allgemeine Sicherheitsbewusstsein und die Wirksamkeit von Sicherheitsrichtlinien und Nutzungsvereinbarungen zu beurteilen (vgl. [52, 11]).

A.2 Ansätze für Penetrationstests

Obwohl es verschiedene Arten von Penetrationstests gibt, hängt die Durchführung solcher Tests in der Regel von den Anforderungen des jeweiligen Unternehmens ab, unabhängig davon, ob der Schwerpunkt dabei auf die Simulation eines Angriffs durch einen Insider oder einer externen Quelle liegt. Die beiden weit verbreiteten Ansätze sind Black-Box- und White-Box-Tests. Der Hauptunterschied zwischen den beiden Ansätzen besteht in der Kenntnis der Implementierungsdetails, welche dem Prüfer über die zu testenden Systeme zur Verfügung gestellt werden (vgl. [130, 2]). Diese beiden Ansätze sowie eine Kombination aus beiden – dem Grey-Box-Test – werden nachfolgend näher erläutert:

- *Black-Box-Test*
  Der Black-Box-Test ist auch unter dem Begriff „externes Testen“ oder „Remote Penetration Testing“ bekannt. Bei diesem Ansatz wird durch den Tester ein Angriff simuliert, welcher keinerlei Kenntnisse über die zu untersuchende Infrastruktur hat, indem er realistische Angriffstechniken (z.B. Social Engineering, Network Scanning, Remote Access, Trojaner usw.) einsetzt. So werden beispielsweise den Testern nur die Webseite oder die IP-Adresse des Unternehmens zur Verfügung gestellt. Die Tester simulieren daher alle Hacking-Techniken, die bekannte und unbekannte Schwachstellen im Netzwerk offen legen können. Vorrangiges Ziel des Black-Box-Penetrationstests ist der Nachweis der Integrität des Netzwerks eines Unternehmens und die proaktive Reduktion von Risiken von externen wie auch internen Angriffen (vgl. [111]).

- *White-Box-Test*
  Der White-Box-Test wird auch als „interner Test“ bezeichnet. Bei diesem Ansatz simulieren Tester einen Angriff als jemand, der über vollständige Kenntnisse der zu testenden Infrastruktur verfügt, oft einschließlich Betriebssystem-Details, IP-Adress-Schema und Netzwerk-Layouts, Quellcode und möglicherweise sogar einige Passwörter. Das Hauptziel des White-Box-Penetrationstests ist es, die Integrität des Unternehmensnetzwerks zu überprüfen und die Risiken der Systeme und Netzwerke durch eine interne Instanz, wie z.B. einen unzufriedenen Mitarbeiter oder Besucher, proaktiv zu reduzieren (vgl. [111][130, 2]).

- *Grey-Box-Test*
  Die Kombination beider Arten von Penetrationstests bietet einen umfassenden Einblick in die interne und externe Sicherheitssicht. Diese Kombination wird als Grey-Box-Test bezeichnet. Der Hauptvorteil bei der Konzeption und Anwendung eines Grey-Box-Ansatzes besteht in den Vorteilen beider oben genannten Ansätze. Voraussetzung dafür ist jedoch, dass ein Prüfer mit begrenzten Kenntnissen des internen Systems den bestmöglichen Ansatz zur Beurteilung der Gesamtsicherheit trifft. In diesem Zusammenhang sind die Testmethoden des Grey-Box-Ansatzes vergleichbar mit denen des Black-Box-Ansatzes an sich, können aber helfen, bessere Entscheidungen und Prüfmöglichkeiten zu treffen, da der Prüfer bereits über die zugrundeliegende Technologie im Bilde ist und diese entsprechend kennt (vgl. [104, 9]).

A.3 Klassifizierung von Penetrationstests

Um einen effizienten und effektiven Penetrationstest mit kalkulierbarem Risiko sicherzustellen, ist es wichtig, eine entsprechende Klassifizierung vorzunehmen. Dabei werden Unterscheidungsmerkmale wie der Umfang der zu prüfenden Systeme, die Aggressivität bei der Durchführung der Tests usw., an die jeweilige Zielsetzung des Tests angepasst. Dadurch erhält jeder Penetrationstest einen individuellen Charakter. In Abbildung A.1 ist eine Klassifikation von möglichen Penetrationstests in Anlehnung an die vom BSI veröffentlichte Studie (vgl. [52, 13]) dargestellt.

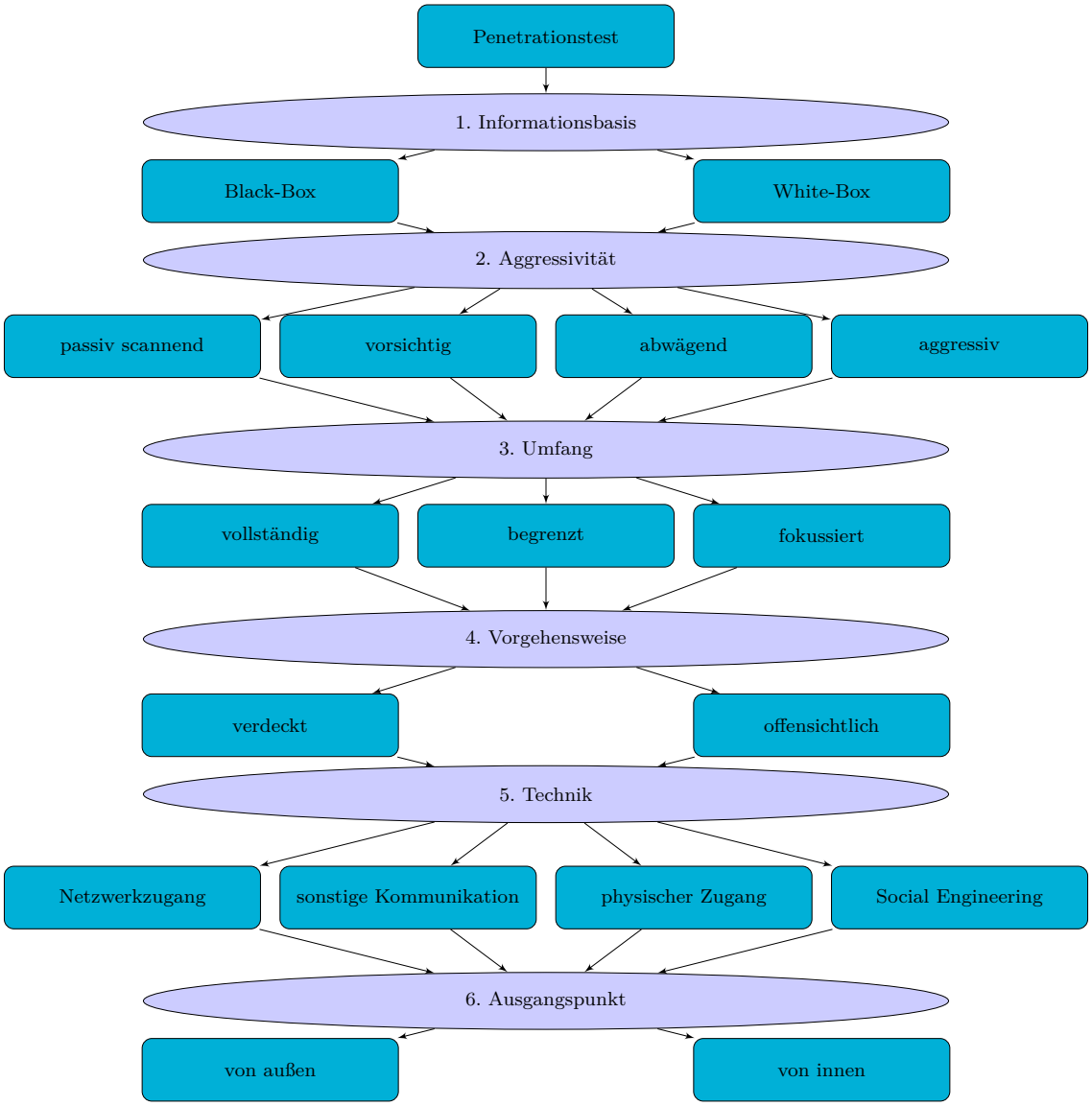

Abbildung A.1: Klassifikation von Penetrationstests (in Anlehnung an [52, 13])

In Abhängigkeit der Ziele des jeweiligen Kunden ist auf Basis der in Abbildung

A.1 genannten Auswahlkriterien ein entsprechender Penetrationstest abzustimmen. In diesem Zusammenhang ist zu berücksichtigen, dass sich nicht alle Kombinationsmöglichkeiten auf sinnvolle Tests abbilden lassen. Der Einsatz eines aggressiven Tests kann in der Regel zum Beispiel sehr zeitnah erkannt werden und ist somit nicht in Kombination mit einem verdeckten Ansatz sinnvoll. Gleichermaßen ist auch ein offensichtlicher Penetrationstest ungeeignet für den Einsatz von Social Engineering Techniken bei der Beschaffung vertraulicher Informationen aus dem Umfeld der bereits vorab gewarnten Mitarbeiter.

Die sechs Kriterien und ihre möglichen Werte werden in den nachstehenden Kapiteln kurz dargestellt (vgl. [52, 13 ff.]).

A.3.1 Informationsbasis

Ausgehend von der Menge an Informationen, die dem Penetrationstester vor der Durchführung seiner Tests über das Zielsystem zur Verfügung stehen, wird zwischen Black-Box- und White-Box-Tests differenziert:

- Bei einem **Black-Box**-Test verfügen die Tester im Vorfeld nicht über Informationen zur Infrastruktur des Zielsystems. Hierbei sind diese als Simulation eines realistischen Angriffs durch einen Außenstehenden anzusehen. Der Hacker muss in diesem Fall die für ihn nötigen Informationen in öffentlich zugänglichen Datenquellen recherchieren.
- Bei einem **White-Box**-Test hingegen verfügen die Tester über umfassende Kenntnisse der zu testenden Systeme respektive der Zielinfrastruktur. Weiterhin handelt es sich bei diesem Test um eine Simulation eines Angriffs durch einen Insider, der sich im Besitz von Systemkenntnissen befinden könnte. Vorrangiges Ziel hierbei ist es, dem Tester alle Informationen zur Verfügung zu stellen, die er benötigt, um einen Einblick in das System zu erhalten und den Test auf Basis von vorab erarbeiteten Erkenntnissen abzuarbeiten.

A.3.2 Aggressivität

Penetrationstests können mit unterschiedlicher Intensität und Aggressivität durchgeführt werden. Dies kann zu einer schnellen und frühzeitigen Erkennung von Angriffen führen. Die Aggressivität bei der Durchführung von Penetrationstests kann in eine der vier nachfolgend definierten Metriken eingeteilt werden:

- Bei der niedrigsten Ausprägung - **passiv scannend** - werden die erkannten Schwachstellen infolge der geringfügigen Interaktion mit dem Zielsystem nicht ausgenutzt.
- Auf der zweiten Stufe - **vorsichtig** - nutzt der Tester nur solche Schwachstellen, deren Verwendung den Betrieb des Zielsystems nicht beeinträchtigen. Der Einsatz von bekannten Standardkennwörtern oder der Versuch, auf Verzeichnisse auf einem Webserver zuzugreifen, sind exemplarische Beispiele für einen vorsichtigen Angriff.
- Bei der nächsthöheren Variante - **abwägend** - wird bei deren Verwendung versucht, Schwachstellen zu nutzen, die möglicherweise Systemausfälle mit sich bringen können. Hierzu zählt beispielsweise das automatische Testen von Passwörtern und die Ausnutzung bekannter Buffer-Overflows in klar identifizierten Zielsystemen.
- Die höchste Stufe - **aggressiv** - bildet einen am stärksten wahrnehmbaren Angriff, dessen Ausführung eine erhebliche Menge an Netzwerkverkehr erzeugt. Der Penetrationstest ist hierbei bestrebt, alle potenziellen Schwachstellen auszunutzen. Als Beispiele für solche aggressiven Angriffe können Buffer-Overflows gegen Zielsysteme und Denial of Service (DoS)-Angriffe als gezielte Überlastung von Systemen genannt werden. Aggressive Tests können schnell identifiziert werden, sodass diese in Kombination mit einer offensichtlichen Vorgehensweise (vgl. Kapitel A.3.4) nicht besonders geeignet sind.

A.3.3 Umfang

Der Umfang von Penetrationstests ist sorgfältig zu definieren, um die in das Testumfeld einzubeziehenden Geräte, Netzwerke und Dienste zu präzisieren. Hinsichtlich des Umfangs des Tests sind drei Kategorien zu unterscheiden: vollständig, begrenzt und fokussiert. Die Komplexität des Tests in Kombination mit den damit verbundenen Kosten variiert hierbei je nach Auswahl des Umfangs. Besonderer Wert sollte hierbei darauf gelegt werden, dass bei „einem erstmaligen Penetrationstest [...] grundsätzlich eine vollständige Überprüfung empfehlenswert (vgl. [52, 14])“ ist.

- Ein **vollständiger** Test untersucht systematisch das Gesamtsystem. In diesem Zusammenhang ist zu berücksichtigen, dass selbst bei einem vollständigen Test bestimmte Systeme (d.h. ausgelagerte wie auch extern gehostete Systeme) unter Umständen nicht getestet werden dürfen.

- Bei einem Penetrationstest mit **begrenztem** Zugriff wird nur der Teil eines Systems überprüft, welcher ein logisches Ganzes bildet. So lassen sich unter anderem alle Systeme in einer demilitarisierten Zone (DMZ) aber auch Systeme, welche eine Betriebs- oder Funktionseinheit bilden, testen.
- Bei einem **fokussierten** Ansatz hingegen konzentriert man sich auf lediglich einen Teil eines Systems oder aber auch nur auf einen Dienst innerhalb des Systems und prüft diesen. Dieser Prüfumfang ist etwa nach einer Änderung oder Erweiterung der Systemlandschaft sinnvoll. Mit einem solchen Test können jedoch nur Aussagen zu dem Teil eines Systems oder Dienstes getroffen werden, der untersucht wurde; nicht aber über die gesamte Sicherheit des Systems.

A.3.4 Vorgehensweise

Penetrationstests können durch die Vorgehensweise der Tester charakterisiert werden. Hierbei gibt es zwei Arten die unterschieden werden, namentlich genannt sind dies verdeckte und offene Ansätze.

- **Verdeckte** Ansätze bedienen sich Techniken, die in der Regel nicht als Angriffe identifiziert werden und so ihre Aktivitäten verschleiern. Üblicherweise sollten Penetrationstests von sekundären Sicherheitssystemen, wie Organisations- und Personalstrukturen, sowie bestehenden Eskalationsverfahren verdeckt durchgeführt werden. Im Rahmen der ersten Erhebung sollten nur Methoden angewandt werden, die nicht unmittelbar als Angriffsversuche auf die Systeme identifizierbar sind, mit dem Ziel, Systemwarnungen zu minimieren.
- Die Verwendung eines **offensichtlichen** White-Box-Tests ist dann empfehlenswert, wenn der verdeckte Ansatz zu keinem Resultat geführt hat. In diesem Fall kann ein umfassender Port-Scan erforderlich werden, welcher in enger Abstimmung mit dem für das System zuständigen internen Mitarbeitern durchgeführt werden sollte. Die internen Mitarbeiter können Teil des Teams sein, welches den offensichtlichen White-Box-Test durchführt. Dies verschafft den Testern die nötige Zeit, auf unerwartete Problemsituationen zeitnah reagieren zu können.

A.3.5 Technik

Es gibt eine Vielzahl an Techniken, die im Rahmen eines Penetrationstests Anwendung finden können. Bei einem konventionellen Penetrationstest erfolgen die Angriffe

auf die Systeme lediglich über das Netzwerk. Des Weiteren besteht auch die Möglichkeit, die Systeme durch andere Kommunikationsnetze, physischen Zugriff aber auch Social Engineering Techniken anzugreifen.

- Penetrationstests auf Basis des **Netzwerkzugangs**, auch bekannt als IP-basierte Penetrationstests, sind die gängigsten Vorgehensweisen, welche einen typischen Hackerangriff simulieren. Durch den Einsatz von netzwerkbasierten Angriffen versucht der Tester, Schwachstellen in Betriebssystemen, Netzwerkprotokollen und Anwendungssystemen auszunutzen. Diese Art des Angriffs beinhaltet beispielsweise DoS-Angriffe, Buffer-Overflows, IP-Spoofing, Sniffing und Port-Scanning.
- Neben IP-basierten Penetrationstests können auch Techniken zum Erkennen von Schwachstellen in **sonstigen Kommunikationsnetzen** eingesetzt werden. Hierzu zählen zum Beispiel drahtlose Netze der mobilen Kommunikation auf Basis der IEEE Norm 802.11 aber auch der IEEE Norm 802.15 für Funktechniken wie Bluetooth oder ZigBee.
- Sicherheitssysteme wie Firewalls, bei denen sich die Konfiguration in der Regel auf einem sehr hohen Sicherheitsniveau befindet, sind inzwischen zumeist flächendeckend im Einsatz, wodurch ein Angriff durch Überwindung dieser Systeme nicht mehr oder nur mit sehr hohem Aufwand realisierbar ist. In der Praxis ist es so oftmals effektiver, die gesuchten Daten durch Umgehen dieser Systeme mittels direktem **physischen Zugriff** zu erlangen. Hierunter versteht man z.B. den direkten Datenzugriff mit Hilfe eines nicht passwortgeschützten Arbeitsplatzes nach unberechtigtem Zutritt zu Gebäuden und/oder Serverräumen.
- Nicht selten wird der Mensch als das schwächste Glied in der Kette der Sicherungssysteme erachtet. Dies erklärt auch, weshalb **Social Engineering** Techniken zumeist sehr erfolgreich in der Praxis zum Einsatz kommen. Social Engineering ist die Kunst der Ausnutzung menschlicher Schwächen, mit dem Ziel, aufschlussreiche Informationen über Systeme zu erlangen. Hierbei eignet sich diese Technik am Besten, wenn spezifische Richtlinien und Verfahren eingeführt wurden, um den Grad deren Umsetzung und Akzeptanz zu evaluieren.

A.3.6 Ausgangspunkt

Ein umfassender Penetrationstest beinhaltet auch die Definition des Ausgangspunktes, von dem aus der Test initiiert wird. Typische Ausgangspunkte sind hierbei

zumeist Firewalls, Fernzugriffsdienste, Webserver aber auch drahtlose Netzwerke.

- Im Rahmen eines Penetrationstests, welcher von einer externen Umgebung aus durchgeführt wird, versucht der Tester die Sicherheit von **außen** zu durchbrechen. Der Fokus richtet sich hierbei im Speziellen auf ein mit dem Internet verbundenes Netzwerk. Derartige Tests versetzen den Tester in die gleiche Position wie die eines Angreifers und vermitteln so ein Gesamtbild eines Angriffs, wie er erwartet werden könnte.
- Bei einem Penetrationstest, welcher von der **inneren** Umgebung aus durchgeführt wird, verfügt der Tester über einen Zugang zur internen IT-Infrastruktur sowie über einen Basiszugriff auf die IT-Systeme. Im Rahmen interner Tests können beispielsweise die Auswirkungen eines Fehlers in der Konfiguration der Firewall sowie physische Zugriffe auf IT-Systeme evaluiert werden, um so einen Angriff von Personen mit Zugriff zum internen Netzwerk nachzustellen.

A.4 Methodologie zur Durchführung von Schwachstellenanalysen und Penetrationstests

Im Rahmen dieses Kapitels wird eine fünfstufige Methodologie zur Durchführung von Penetrationstests vorgestellt, welche sich an der Studie des BSI (vgl. [52, 45 f.]) orientiert. Ein wesentlicher Bestandteil der Methodik ist hierbei ein systematisches Vorgehen zur Durchführung von Penetrationstests, woraus sich individuelle Handlungspläne für spezifische Penetrationstests als Erweiterung der Schwachstellenanalyse ableiten lassen.

A.4.1 Anforderungen an die Methodologie

Ein methodisches Vorgehen beschreibt und strukturiert das Durchführen eines Penetrationstests im Rahmen eines Auftrags. Dementsprechend ist es unerlässlich, sich auf die Ziele des Kunden zu konzentrieren respektive sicherzustellen, dass die Orientierung dabei nicht außer Acht gelassen wird (vgl. [52, 44]). Darüber hinaus sollte die in Kapitel 2.4.2 beschriebene Gegenüberstellung zwischen einer Schwachstellenanalyse und einem Penetrationstest nicht vernachlässigt werden. Ziel dieser Arbeit ist es, Schwachstellen in Funkprotokollen am Beispiel von Smart Home Anwendungen zu analysieren. In diesem Zusammenhang werden die Schwachstellen eines definierten Funkprotokolls, im Falle dieser Ausarbeitung die der ZigBee Spezifikation, mittels eines Penetrationstests detailliert analysiert mit dem Ziel, diese entsprechend auszunutzen. Für die verbleibenden Funkprotokolle wird eine analytische Betrachtung

mittels Schwachstellenanalyse durchgeführt. Im Rahmen der folgenden Ausführungen soll ein Hintergrund zu den späteren Kapiteln dieser Arbeit geschaffen werden. Zur Erreichung des angestrebten Ziels der Schwachstellenanalyse in Kombination mit Penetrationstests ist es notwendig, dass sowohl aus theoretischer als auch aus praktischer Sicht die richtige Methodik und der geeignete Workflow definiert werden.

Zur Identifizierung von Schwachstellen in den zu untersuchenden Funkprotokollen empfiehlt sich die Anwendung des sogenannten „Threat Modeling“, z. Dt. Bedrohungsmodellierung, welches eine „wertvolle Grundlage für Risikoanalysen, Penetrationstests und andere Arten von Sicherheitsanalysen (vgl. [132])“ darstellt. Hierbei kann zur Identifikation, Klassifizierung und Modellierung von Bedrohungen auf die Definition von Angriffsvektoren zurückgegriffen werden. Für die Bewertung der identifizierten Bedrohungen und Risiken bietet sich das als „DREAD“ bezeichnete Modell an (vgl. [109]). Eine nähere Darstellung und Erläuterung des „DREAD“-Modells erfolgt im Kapitel 2.5.2.

A.4.2 Schlüsselkonzept der Schwachstellenanalyse

In Abbildung A.2 wird das zyklische Schlüsselkonzept für die Durchführung einer Schwachstellenanalyse nach Robert Boyce veranschaulicht (vgl. [51, 3]). Bei der Konzeption eines Verfahrens ist es empfehlenswert, von einem übergeordneten Niveau aus zu starten und auf die Definition der spezifischen Details hinzuarbeiten. Das Schlüsselkonzept dient somit als Ausgangslage für die Schwachstellenanalyse und soll in die Phasen des Penetrationstests, welcher sich – wie bereits erwähnt – an der Studie des BSI (vgl. [52, 45 f.]) orientiert, eingebunden werden. Hierdurch soll eine einheitliche Vorgehensweise geschaffen werden, welche sowohl für die Schwachstellenanalyse als auch für die tiefer gehende Analyse mittels Penetrationstest in dieser Ausarbeitung zur Anwendung kommen soll. Nachfolgend sind die Phasen des Schlüsselkonzeptes kurz dargestellt (vgl. [51, 4]).

- *Phase 1 - Durchführung des Assessments*
  Im Mittelpunkt dieser Phase stehen zwei wesentliche Zielsetzungen – die Planung und Durchführung der Schwachstellenanalyse. Zu den planungsrelevanten Komponenten zählen beispielsweise die Erfassung aller relevanten Informationen sowie die Definition des Umfangs der Tätigkeiten. Hierauf aufbauend wird dann die eigentliche Schwachstellenanalyse durchgeführt.

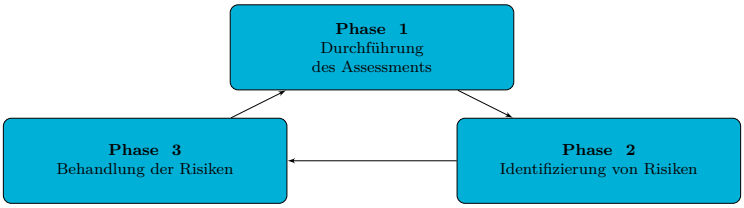

Abbildung A.2: Schlüsselkonzept für die Durchführung einer Schwachstellenanalyse (in Anlehnung an [51, 3])

- *Phase 2 - Identifizierung von Risiken*
  Bei dieser Phase kann es sich um eine Vielzahl an verschiedenen Aufgaben handeln. So können beispielsweise die aus dem Assessment resultierenden Daten ausgewertet werden, um die Verantwortlichkeit in Bezug auf die Probleme zu ermitteln. Durch die Möglichkeit der Datenüberprüfung können unternehmensweite Risikoanalysen und -trends durchgeführt werden.

- *Phase 3 - Behandlung der Risiken*
  Im Rahmen dieser Phase wird versucht, die in der vorangegangenen Phase identifizierten Risiken zu beheben. Vor Beginn der Behebung der Risiken muss zunächst eine Analyse durchgeführt werden, um beispielsweise festzustellen, ob ein Dienst, welcher ein Problem verursacht hat, tatsächlich benötigt wird. Ist der Dienst notwendig, sollte das System entsprechend aktualisiert werden. Bei Nichtbenutzung des Dienstes kann dieser hingegen deaktiviert werden.

A.4.3 Phasen eines Penetrationstests

Basierend auf den vorangegangenen Ausführungen werden die fünf Phasen eines Penetrationstests nach der Studie des BSI nachstehend kurz dargestellt (vgl. [52, 45 f.]). Die verschiedenen Phasen laufen hierbei in chronologischer Reihenfolge ab und sind in Abbildung A.3 dargestellt.

- *Phase 1 - Vorbereitung*
  Es bedarf einer umfangreichen Planung und Vorbereitung, um Penetrationstests zu einem erfolgreichen Abschluss zu bringen. Dabei werden die Ziele, der Umfang, die gesetzlichen Einschränkungen und die zeitlichen Planung für den Auftrag in dieser Phase definiert und formuliert. In einem Unternehmen ist das Ziel eines Penetrationstests beispielsweise, aufzuzeigen, welche ausnutzbaren Schwachstellen im Netzwerk vorhanden sind. Der Rahmen hierfür kann durch

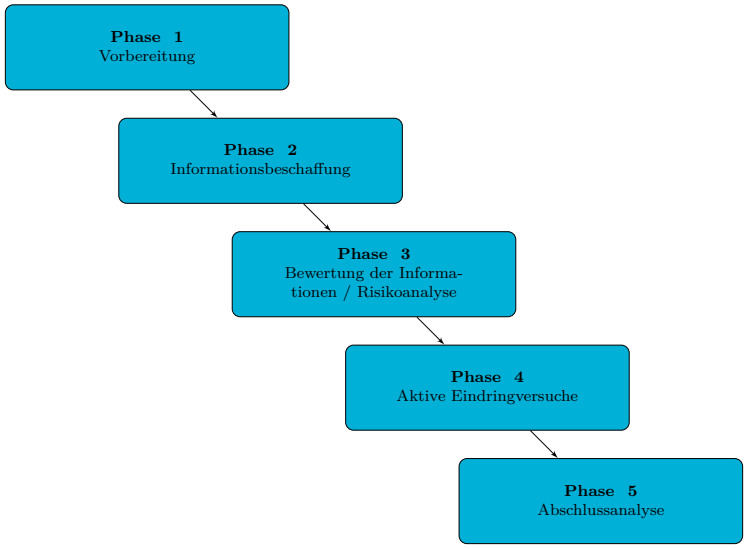

Abbildung A.3: Phasen eines Penetrationstests (in Anlehung an [52, 45 f.])

die Identifizierung von bestehenden Sicherheitsrichtlinien, Industriestandards und Best Practices usw. festgelegt werden. Im Allgemeinen umfasst diese Phase somit alle Aktivitäten, die vor Beginn des eigentlichen Penetrationstests durchgeführt werden müssen.

- *Phase 2 - Informationsbeschaffung*
  Nach der Definition der Ziele, des Umfangs, der rechtlichen Rahmenbedingungen und der zeitlichen Planung erfolgt nun die Durchführung der eigentlichen Überprüfung. Hierbei kann die Überprüfung als Phase der Informationsbeschaffung verstanden werden und wird „auch als passiver Penetrationstest bezeichnet (vgl. [52, 45])“.

  Im Rahmen dieser Phase verfolgt der Penetrationstester das Ziel, so viele öffentlich zugängliche Informationen wie möglich auf technischer wie auch auf nicht-technischer Grundlage zu ermitteln. Dabei wird angestrebt, einen möglichst umfassenden und detaillierten Gesamtüberblick über die installierten Systeme zu erhalten, der auch potenzielle Angriffspunkte und bekannte Sicherheitsmängel mit einbezieht. Der Penetrationstester muss hierbei in der Lage sein, den Sicherheitsstatus innerhalb eines Systems oder Netzwerks so zu erfassen, sodass alle potenziellen Schwachstellen identifiziert werden können.

- *Phase 3 - Bewertung der Informationen / Risikoanalyse*
  Vor der Durchführung der zum Teil sehr langwierigen Testschritte des aktiven Angriffs ist es notwendig, die gewonnenen Informationen zu analysieren und

auszuwerten, um ein effizientes, transparentes und vor allem wirtschaftliches Vorgehen zu gewährleisten. Dabei sind die vereinbarten Ziele des Penetrationstests, die mögliche Anfälligkeit der Systeme und der geschätzte Aufwand zur Evaluierung der potenziellen Sicherheitslücken im Rahmen der darauf folgenden aktiven Eindringversuche zu berücksichtigen. Auf der Grundlage dieser Evaluation erfolgt dann die Selektion der Ziele die in Phase 4, den aktiven Eindringversuchen, umgesetzt werden. Hierbei können beispielsweise nur diejenigen Systeme aus der Reihe der identifizierten Systemen selektiert werden, die aufgrund ihrer Konfiguration beziehungsweise der identifizierten Anwendungen/Dienste als potentielle Schwachstellen bekannt sind.

- *Phase 4 - Aktive Eindringversuche*
  Die Phase der aktiven Eindringversuche gilt als eine der spannendsten und zugleich anspruchsvollsten in der Durchführung von Penetrationstests. In diesem Rahmen werden nach Identifizierung und Analyse der Schwachstellen geeignete Angriffsmethoden ausgewählt und geeignete Ziele im Hinblick auf mögliche Angriffe festgelegt. Nach der Identifizierung geeigneter Ziele erfolgt der eigentliche Angriff. Im Erfolgsfall wird die Schwachstelle verifiziert und bestätigt. In der Folge werden dann weitere Versuche unternommen, um höhere Privilegien zu erlangen. An dieser Stelle wird erst deutlich, inwieweit die im Rahmen der Informationsbeschaffung identifizierten vermeintlichen Schwachstellen tatsächliche Risiken darstellen. Nach Abschluss dieser Phase verfügt der Penetrationstester in der Regel über ein umfassendes Wissen der Sicherheitsstärken und -schwächen des Zielsystems oder Netzwerks, welches in der Abschlussanalyse dokumentiert und erörtert wird.

- *Phase 5 - Abschlussanalyse*
  Die Abschlussanalyse kann zum einen parallel zu den vorangegangenen Phasen als auch am Ende der aktiven Eindringversuche erfolgen. Es ist ratsam, dass die Berichte eine Evaluierung der Schwachstellen auf der Grundlage potenzieller Risiken sowie Empfehlungen zur Minimierung etwaiger Schwachstellen und Risiken beinhalten. Im Rahmen dieser Berichtsphase ist die Transparenz der Tests und der dabei aufgezeigten Schwachstellen zu gewährleisten. Grundsätzlich bietet diese Abschlussanalyse die Möglichkeit, die gesamte Sicherheitslage der Systeme oder des Netzwerks nachzuvollziehen. Die Erkenntnisse sowie die daraus abgeleiteten Risiken in Bezug auf die IT-Sicherheit sollten in einem Abschlussgespräch mit dem Auftraggeber nach Abschluss des Penetrationstests detailliert erläutert werden.

B Einrichtung des Penetrationstestlabors

In diesem Kapitel wird die Einrichtung des Penetrationstestlabors hinsichtlich der Hard- und Software beschrieben. Das Penetrationstestlabor kommt für die Phase der aktiven Eindringversuche in der Schwachstellenanalyse des Protokolls der ZigBee Spezifikation zum Einsatz.

B.1 Hardware zur Funkanalyse

B.1.1 APIMote (for ZigBee sniffing and transmission)

Die APIMote Hardware ist in Abbildung B.1 dargestellt. Entwickelt wurde diese von Attify, einer auf IoT und Mobile Security spezialisierten Firma. APIMote ist bereits mit der KillerBee-Firmware vorinstalliert, sodass mit dem passenden Betriebssystem sowie der Software direkt mit der Evaluierung der IEEE 802.15.4 / ZigBee Sicherheit begonnen werden kann.

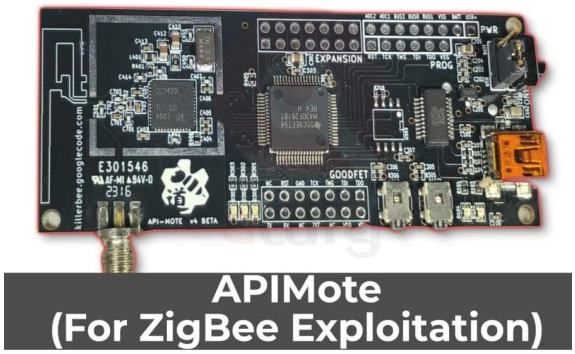

Abbildung B.1: APIMote (for ZigBee sniffing and transmission) (vgl. [106])

B.1.2 CC2531 USB Dongle

Der CC2531 USB Dongle wurde von Texas Instruments auf Basis des CC2531 entwickelt. In Abbildung B.2 ist er entsprechend dargestellt. Der Dongle ist in der Lage, ZigBee Datenpakete zu erfassen und im Texas Instruments SmartRF Packet Sniffer darzustellen (vgl. [136]).

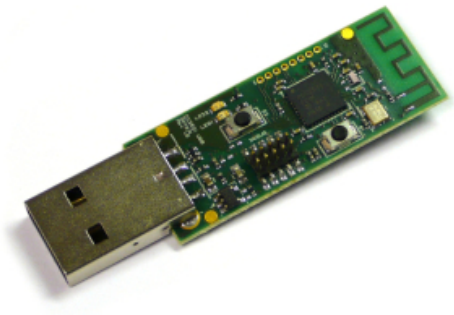

Abbildung B.2: CC2531 USB Dongle (vgl. [135])

B.1.3 AVR Raven

Das AVR Raven Starterkit ermöglicht die Entwicklung, Demonstration und das Debugging einer Reihe von drahtlosen Anwendungen inkl. IEEE 802.15.4, 6LoWPAN und ZigBee Netzwerken. Das Kit beinhaltet zwei Raven-Boards, die als End Devices fungieren können, sowie einen USB-Stick, der einen Controller simulieren kann.

Abbildung B.3: AVR Raven (vgl. [123])

B.2 Betriebssystem und Angriffswerkzeuge zur Funkanalyse

B.2.1 AttifyOS1.3

AttifyOS ist eine Penetrationstest-Distribution zur Beurteilung der Sicherheit von IdD-Geräten. Die Distribution basiert auf LUbuntu und enthält bereits vorkonfigurierte Tools (vgl. [105]). Dies bietet den Vorteil, dass keine Zeit mit der Installation, Konfiguration und Einrichtung verschiedener Tools verbracht werden muss, die für den Penetrationstest erforderlich sind. Für diese Master-Thesis wird das Killerbee / Attify ZigBee Framework genutzt, welches entsprechend auf die APIMote Hardware aufsetzt (vgl. Kapitel B.1.1). AttifyOS wird als virtuelle Maschine über Oracle VirtualBox auf Windows 10 betrieben.

B.2.2 Killerbee / Attify ZigBee Framework

Das Killerbee / Attify ZigBee Framework ist ein auf GoodFET basierendes Framework, welches das Prüfen von Sicherheitslücken im IEEE 802.15.4/ZigBee Protokoll ermöglicht. Hierbei bildet KillerBee eine vereinfachte Schnittstelle, welche es ermöglicht die Paketerfassung über Formate wie libpcap oder das Daintree SNA Format zu erfassen. KillerBee stellt mehrere Tools zur Verfügung, die entwickelt wurden um ZigBee als auch IEEE 802.15.4 Netzwerke angreifen zu können (vgl. [121]).

- **zbid:**
  Identifiziert verfügbare Schnittstellen, die von KillerBee und den zugehörigen Tools verwendet werden können.
- **zbwireshark:**
  Ähnlich wie zbdump, aber mit einer Named Pipe für die Echtzeit-Erfassung und -Ansicht in Wireshark.
- **zbdump:**
  Ein TCP-Dump-ähnlicher Vorgang zum Erfassen von IEEE 802.15.4-Frames in eine libpcap- oder Daintree SNA-Paketerfassungsdatei.
- **zbreplay:**
  Implementiert einen Replay-Angriff. Hierbei liest das Tool aus einer bestimmten Daintree DCF- oder libpcap-Paketerfassungsdatei und sendet die Frames erneut. ACK-Frames werden nicht erneut übertragen.
- **zbstumbler:**
  Aktives ZigBee und IEEE 802.15.4 Netzwerkerkennungswerkzeug. Es sendet

hierbei Beacon Request Frames während des Channel Hoppings aus, zeichnet auf und zeigt zusammengefasste Informationen über erkannte Geräte an. Diese kann es auch als Ergebniss in einer CSV-Datei protokollieren.

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die hier vorliegende Arbeit selbstständig, ohne unerlaubte fremde Hilfe und nur unter Verwendung der aufgeführten Hilfsmittel angefertigt habe.

Ort, Datum Unterschrift