MASTERARBEIT

IOT & IPV6

Sicherer Einsatz von IoT-Geräten in IPv6-Netzwerken

ausgeführt am

Studiengang

Informationstechnologien und Wirtschaftsinformatik

Von: Manfred SCHLACHER

Personenkennzeichen: 1610320037

Graz, am 15. März 2018 .....

Unterschrift

EHRENWÖRTLICHE ERKLÄRUNG

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benützt und die benutzten Quellen wörtlich zitiert sowie inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

.....

Unterschrift

DANKSAGUNG

Ich widme diese Arbeit in erster Linie meiner Freundin Sandra, die mich immer wieder dazu ermutigt hat weiter zu machen und ohne die diese Arbeit nicht zustande gekommen wäre, und meinem Sohn Nuriel. Die Fertigstellung dieser Arbeit stellt gleichzeitig auch den Beginn eines neuen und aufregenden Lebensabschnittes dar, dem meine Freundin und ich bereits voller Vorfreude entgegen blicken.

Auch danke ich meinen Eltern die immer an mich glaubten und mich Zeit meines Lebens immer dabei unterstützt haben meine mir selbst auferlegten Ziele zu erreichen, und meinen Freuden, die immer Verständnis dafür hatten wenn einmal weniger Zeit für sie erübrigen konnte.

Ein ganz besonderer Dank gilt meinem Betreuer Christian Schmid, MSc, der mir in allen Bereichen dieses komplexen Themengebietes stets mit Rat und Tat zur Seite stand und Mama Mick für das erneute setzen von Tempos und Kommas.

KURZFASSUNG

Das Internet ist seit seiner Entwicklung stetig gewachsen. Durch das Aufkommen neuer Technologien stieg die Zahl der über dieses weltweit verbreitete Netzwerk kommunizierenden Endgeräte immer weiter an. Besonders das wachsende Interesse von Unternehmen ihre Systeme miteinander zu vernetzen führt dazu, dass die zu Verfügung stehende Menge freier IP-Adressen immer geringer wird. Durch den Einzug von IoT-Geräten in privaten Bereichen, welcher den Verzehr von IP-Adressen beschleunigt ist es nur noch eine Frage der Zeit bis das bestehende IPv4 flächendeckend seinem von seinem Nachfolgeprotokoll IPv6 ersetzt wird.

Durch diesen Wandel steigt zwar die Anzahl der theoretisch, zu Verfügung stehenden IP-Adressen auf circa 340 Sextillionen an, jedoch entstehen durch die von IPv6 genutzten Mechanismen und eingesetzten Protokolle auch neue Angriffsvektoren welche im Besonderen IoT-Geräte zu potentiellen Angriffszielen von Hackern werden lassen können. Auf Grund ihrer eingeschränkten Ressourcen und oft unzureichenden Versorgung mit sicherheitsrelevanten Softwareupdates durch die Hersteller oder die Endnutzer können IoT-Geräte sehr schnell zu einer ernstzunehmenden Gefahr für IPv6-Netzwerke werden. Dabei zeigt sich, dass der primäre Fokus im Besonderen auf die Absicherung der Grenzen des Netzwerkes gerichtet sein muss.

In dieser Arbeit wird auf Basis von Analysen des IPv6-Protokolles und den Merkmalen und Spezifikationen Gängiger IoT-Gerät ein einfaches Framework entwickelt welches sich an die besonderen Sicherheitsanforderungen von IoT-Geräten in IPv6-Netzwerken gerichtet ist. Das IPv6/Iot-Security Framework soll als Basis dienen um einen sicheren Einsatz von IoT-Geräten in sensiblen Netzwerken zu gewährleisten ohne dabei andere Elemente der Infrastruktur nachteilig zu beeinflussen.

ABSTRACT

Since developed, the Internet has steadily grown. With the advent of new technologies, the number of devices communicating through this worldwide network has continued to increase. Especially the growing interest of companies integrate their systems means, that the available amount of available IP addresses has become smaller and smaller. With the emergence of IoT-devices in private areas, which speeds up the consumption of IP addresses, it is only a matter of time before the existing IPv4 is replaced completely by its successor protocol IPv6.

Although this change will increase the number of theoretically available IP addresses to approximately 340 sextillion, the mechanisms and protocols used by IPv6 will also give rise to new attack vectors which, in particular, may make IoT-devices potential attack targets for hackers. Because of their limited resources and often inadequate supply of security-related software updates by manufacturers or end users, IoT-devices can quickly become a serious threat to IPv6 networks. It turns out that the primary focus must be on securing the boundaries of the network.

In this work, based on analysis of the IPv6 protocol and the characteristics and specifications of the common IoT-device, a simple framework is developed which addresses the specific security requirements of IoT-devices in IPv6 networks. The IPv6/IoT-Security Framework is designed to serve as the basis for ensuring the safe use of IoT-devices in sensitive networks without adversely affecting other elements of the infrastructure.

INHALTSVERZEICHNIS

1 EINLEITUNG

"Ein Hacker ist jemand, der versucht einen Weg zu finden,
wie man mit einer Kaffeemaschine Toast zubereiten kann."
Wau Holland

Das Internet ist seit seiner Entwicklung ein stetig wachsender Organismus. Durch die Erfindung des Smartphones und weiteren mobilen Endgeräten stieg die Zahl der sich im Internet befindlichen Geräte laut (Gartner 07.02.2017) im Jahr 2017 auf 8.3 Milliarden an. Es ist davon auszugehen, dass diese Entwicklung durch neue Technologien und betriebswirtschaftliche Umbrüche (Stichwort „Industrie 4.0“ und „Internet of Things“) in den nächsten Jahren weiter voranschreiten wird. Das derzeit im Internet gängigste Netzwerkprotokoll zur eindeutigen Identifikation von teilnehmenden Geräten, das Internet Protokoll Version 4 (IPv4), wird in naher Zukunft nicht mehr in der Lage sein, alle teilnehmenden Geräte mit einer Adresse zu versorgen.

1.1 Motivation

Mithilfe von IPv6 soll der Adressknappheit entgegengewirkt werden. Eine IPv6-Adresse besteht aus acht Blöcken zu je 16 Bit. Diese Erweiterung vergrößert den verfügbaren Adressraum auf 2^{128} (~340 Sextillionen) Adressen. Obwohl bereits ein Großteil der Endgeräte in der Lage ist, mit IPv6 -Adressen zu arbeiten, werden zum jetzigen Zeitpunkt (April 2017) immer noch bevorzugt IPv4-Adressen verwendet, weil viele Anwender und Anwenderinnen keinen direkten Nutzen in der Migration zu IPv6 sehen. Die flächendeckende Nutzung von IPv6-Adressen ist angesichts des immer größer werdenden Marktes von IoT-Geräten jedoch nur noch eine Frage der Zeit. Mit der zunehmenden Nutzung von IPv6-Adressen entstehen aber neue Herausforderungen für die administrierenden Personen. Die Hoffnung, dass IPv6-Adressen eine größere Sicherheit gegenüber Attacken bieten, ist laut Hanns Proenen (CISO at GE Europe) durch die Einführung von IPv6 nicht ad hoc gegeben. Nach Proenen wird die Einführung von IPv6 alleine zu keiner Steigerung der Sicherheit bei IoT-Geräten führen. „Tatsächlich birgt IPv6 neue Risiken welche wir verstehen müssen[…]“ (Proenen 2016).

1.2 Problemstellung

Computer, die über ein Netzwerkmodul verfügen und sich innerhalb eines Netzwerkes befinden, können über eine IP-Adresse innerhalb dieses Netzwerkes eindeutig identifiziert werden. Dies gilt sowohl für private Heimnetzwerke als auch für das Internet. Für die Identifikation und somit die Adressvergabe ist IPv4 das derzeit am weitesten verbreitete Internet Protokoll.

Eine IPv4-Adresse gliedert sich in vier Blöcke zu je acht Bit. Daraus ergibt sich eine maximale Anzahl von theoretisch 4.294.967.296 nutzbaren Adressen.

Die Anzahl der Endgeräte, die mit dem Internet verbunden sind, steigt durch die rasch voranschreitende Entwicklung von Internet of ThingsGeräten (IoT-Geräten) enorm. Dadurch ist der zur Verfügung stehende Adresspool von IPv4-Adressen in absehbarer Zeit erschöpft.

1.3 Zielsetzung

Ziel der Arbeit ist es, potentielle Sicherheitsrisiken, die durch die Verwendung von IPv6-Adressen auf IoT-Geräten auftreten, zu identifizieren. Sicherheitsrisiken, die sich durch entsprechende Konfiguration der Endgeräte eliminieren lassen, sollen zusammen mit entsprechenden Beispielkonfigurationen in einem „Best-Practice Guide“ erläutert werden.

Auf Basis dieser Zielsetzung lässt sich folgende Forschungsfrage definieren:

> „Welche Risiken entstehen durch die Verwendung von IPv6 auf IoT-Geräten bezogen auf IT-Security, und wie lassen sich diese neu entstandenen Sicherheitsrisiken durch entsprechende Konfiguration der IoT-Geräte bestmöglich minimieren?“

Unter Berücksichtigung der genannten Forschungsfrage lassen sich folgende Hypothesen formulieren, die im Zuge dieser Arbeit überprüft werden sollen:

H1: Durch den Einsatz von IPv6 auf IoT-Geräten entstehen potentielle Sicherheitslücken, die sich durch Änderung der Konfiguration minimieren lassen.

H0: Die Verwendung von IPv6 auf IoT-Geräten hat keinen sicherheitskritischen Einfluss für EndnutzerInnen.

1.4 Methodisches Vorgehen

Zur Erreichung des oben genannten Zieles und Beantwortung der daraus abgeleiteten Forschungsfrage wird ein Methodenmix von empirischen Forschungsmethoden angewandt. Dieser Methodenmix setzt sich aus folgenden Methoden zusammen:

- Inhaltsanalyse

  Auf Basis einer intensiven Recherche über IoT-Geräte, deren Einsatzbereiche und Potentiale sowie der Spezifikation von IPv6, sollen die Sicherheitsrisiken identifiziert werden, die durch den Einsatz von IPv6 in IoT-Geräten auftreten können. Dadurch könnten in einem IPv6-Netzwerk betriebene IoT-Geräte zu einem potentiellen Angriffsziel für Hacker werden. Entsprechend der Möglichkeiten der betrachteten IoT-Geräte sollen mögliche Eingriffe in die Netzwerkkonfiguration aufgezeigt werden, um anschließend die Chancen einer erfolgreichen Attacke auf ein Minimum zu reduzieren.

- Laborexperiment

  In einem darauffolgenden Laborexperiment sollen die ermittelten Konfigurationsmöglichkeiten evaluiert werden, um IoT-Geräte gegen die zuvor identifizierten Attacken unter der Verwendung von IPv6 zu schützen. Besonderes Augenmerk soll darauf gelegt werden, ob durch entsprechende Konfiguration möglicherweise neue potentielle Angriffspunkte entstehen.

Zur Erreichung des gesetzten Zieles soll eine Netzwerklandschaft aus n (Anm.: Anzahl wird basierend auf der durchgeführten Recherche determiniert) unterschiedlichen IoT-Geräten errichtet werden. Diese IoT-Geräte sollen den ermittelten Angriffsszenarien ausgesetzt werden, um zu beweisen, dass die entsprechende Konfiguration den gewünschten Sicherheitszuwachs erbringt.

- Literaturrecherche / Inhaltsanalyse

  Mittels einer ausgedehnten Literaturrecherche soll ein umfassendes Bild über die Definition „Internet of Things“ und der Spezifikation von IPv6 erstellt werden. Zusätzlich sollen derzeit am Markt befindliche IoT-Geräte auf deren Leistungsmerkmale und Konfigurationsmöglichkeiten untersucht werden.

- Auswahl von IoT- Geräten für das Laborexperiment

  Zur Durchführung des Laborexperimentes sollen n IoT-Geräte ausgewählt werden, die im späteren Verlauf den identifizierten Angriffsszenarien ausgesetzt werden.

- Identifizierung potentieller Angriffsszenarien auf Basis der gewählten IoT- Geräte

  Basierend auf der Literaturrecherche und den ausgewählten IoT-Geräten werden Angriffsszenarien abgeleitet, um unberechtigten Zugriff auf die genannten IoT-Geräte zu erlangen und eine Malware einbringen zu können.

- Suche nach geeigneten Sicherheitsmechanismen

  Zu den identifizierten Angriffsszenarien sollen Methoden gefunden werden, um im Bereich des Möglichen die IoT-Geräte so abzusichern, dass sie gegen die ausgeübten Angriffe geschützt sind.

- Laborexperiment

  In einem Laborexperiment soll an einem Beispielnetzwerk, bestehend aus den ausgewählten IoT-Geräten, evaluiert werden, ob die zuvor erarbeiteten Mechanismen und Konfigurationen die Geräte gegen die jeweilige Attacke schützen können. Gleichzeitig soll geprüft werden, ob durch die vermeintliche Absicherung der IoT-Geräte gegen die jeweilige Bedrohung neue Sicherheitsrisiken entstehen.

- Auswertung der Ergebnisse / Erstellung Best-Practice-Guide

  Basierend auf den Ergebnissen aus den Laborexperimenten, insbesondere der Frage, welche Mechanismen zur Absicherung welcher Bedrohung sorgen und ob sich neue Schwachstellen durch die eingesetzte Konfiguration auftuen, werden die ermittelten Maßnahmen priorisiert. Die priorisierten Maßnahmen fließen abschließend in einen Best-Practice-Guide ein.

2 INTERNET OF THINGS

„Internet of Things“ ist ein Begriff der erstmals um die Jahrtausendwende aufkam und in den ersten Jahren hauptsächlich durch das Auto-ID-Center am Massachusetts Institut of Technology (MIT) geprägt wurde (Ashton 2009). Bereits wenige Jahre später, Mitte der ersten Dekade des einundzwanzigsten Jahrhunderts, hat sich der Begriff im deutsch- und englischsprachigen Raum etabliert. Rasch wurde „Internet of Things“ zu einem viel diskutierten Thema in Wirtschaft und Technik.

2.1 Definition und Abgrenzung

Die Grundidee, die hinter dem „Internet der Dinge“ steckt, ist der Ansatz, durch das Zusammenschließen physischer Endgeräte durch elektronische Bauteile eine „smarte“ Vernetzung zu erreichen. Durch eindeutige Identifizierungsmechanismen sind die vernetzten Geräte dann in der Lage, untereinander zu kommunizieren und gemeinsam an der Lösungsfindung einer Problemstellung zu arbeiten. Durch diesen Schritt geht das „Internet der Dinge“ über das klassische Internet hinaus, in dem größtenteils PCs miteinander verbunden sind. Ist durch fortschreitende Miniaturisierung eine Integration von elektronischen Komponenten und Kleinstcomputern in Alltagsgegenstände bzw. Dinge zu beobachten, und sind diese Dinge bzw. ihre integrierten Computer mit dem Internet verbunden, spricht man vom „Internet of Things“ (Müller 2016). Die International Telecommunication Union (ITU) definiert „Internet of Things“ als „Eine weltweite Infrastruktur für die Informationsgesellschaft zur Bereitstellung von fortschrittlichen Diensten, basierend auf bestehenden und zukünftigen, vollständig kompatiblen Informations- und Kommunikationstechnologien (International Telecommunication Union 2016). Im Zusammenhang mit dem „Internet of Things“ wird „Things“ wie folgt definiert: „Ein Objekt der physischen Welt (physisches Ding) oder der virtuellen Welt (virtuelles Ding), welches in der Lage ist, sich in das Kommunikationsnetzwerk zu integrieren und darin identifiziert werden kann.“

Aus Sicht technischer Standardisierung kann das „Internet of Things“ als globale Infrastruktur der Informationsgesellschaft angesehen werden, die es ermöglicht, durch den Zusammenschluss physikalischer und virtueller Dinge herausragende Dienste anzubieten.

IoT-Endgeräte erweitern die bisherigen zwei Dimensionen bestehender Informations- und Kommunikationstechnologien (ICT), „Any-Place-Communication“ und „Any-Time-Communication“, um eine weitere, die sogenannte „Any-Device-Communication“. Diese dritte Dimension inkludiert nun die Möglichkeit der autonomen Kommunikation zwischen Dingen (Abbildung 1).

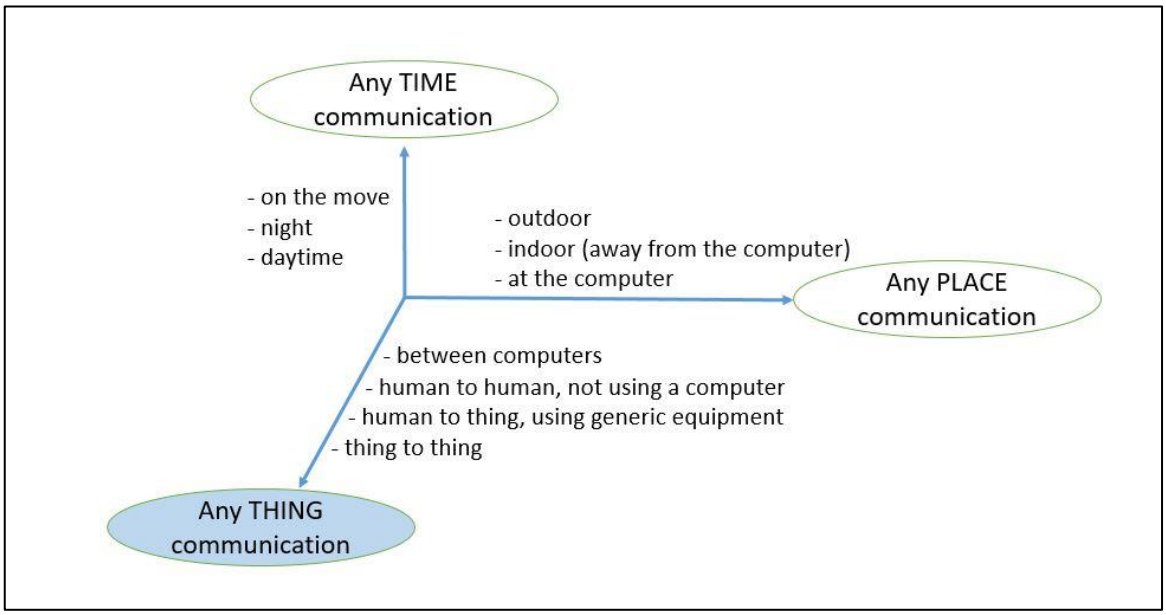


Abbildung 1:Dimensionen von ICT (International Telecommunication Union 2016).

Die ITU definiert darüber hinaus folgende fundamentalen Eigenschaften, die das „Internet of Things“ charakterisieren:

- Interkonnektivität:

  Jedes IoT-Gerät kann mit der globalen Informations- und Kommunikationsinfrastruktur verbunden werden.

- Heterogenität:

  Teilnehmende Geräte im „Internet of Things“ sind heterogen. Dies beinhaltet sowohl unterschiedliche Hardware als auch Plattformen und Netzwerkschnittstellen.

- Dynamischer Wandel:

  Der Zustand einzelner Geräte unterliegt einem ständigen Wandel. Ein Wandel sind beispielsweise wechselnde Zustände wie: „Connected“ und „Disconnected“, „Sleepmodus“ und „Awake“.

- Enorme Größenverhältnisse:

  Die Anzahl der Geräte, die am „Internet of Things“ teilnehmen und darüber kommunizieren werden, wird um ein Vielfaches größer sein als die Anzahl der Geräte, die bisher mit dem Internet verbunden sind. Das Verhältnis von Kommunikation, die autark von Geräten angestoßen wird, wird merklich jene, durch menschliche Handlungen ausgelöste Kommunikation übersteigen. Die Bearbeitung dieser Datenmengen in Echtzeit, zu Zwecken der Weiterverarbeitung, wird einen ebenso kritischen Aspekt annehmen wie die Übertragung der Daten. Dies ist darin begründet, dass einerseits die Rechenleistungen der verarbeitenden Prozessoren, andererseits auch die Kapazitäten der Übertragungskanäle dem wachsenden Bedarf gerecht werden müssen.

- Gerätebezogene Dienste:

  Dank der Technologien des „Internet of Things“ wird physischen Geräten die Möglichkeit gegeben, eigenständig innerhalb ihrer Möglichkeiten zu agieren. Dies hat zur Folge, dass sich die physische Welt weiterentwickeln wird, um die Anzahl der Möglichkeiten der interagierenden Dinge zu erhöhen. Um dies zu ermöglichen, muss sich auch die virtuelle Welt, beispielsweise Softwareprodukte, weiterentwickeln, um oben genannte physische Entwicklung zu ermöglichen.

2.2 State of the Art

IoT ist bereits weit in unser tägliches Leben vorgedrungen, da das „Internet of Things“ schon viele unserer Alltagsgegenstände mit dem Internet verbindet. „IoT“ erweitert die uns umgebenden „dummen“ Objekte mit Fähigkeiten der Kommunikation, Speicherung und Verarbeitung von Daten (Romdhani et al. 2015). Aus der Sicht von Privatpersonen sind die Bereiche „Smart-Home“ und „E-Health“ ohne Zweifel jene, die am meisten zur rasanten Verbreitung von „IoT“-Geräten beigetragen haben.

Das „Internet der Dinge“ tritt nicht als völlig neue Kategorie von Systemen in unseren Alltag ein. Es ist vielmehr der inkrementelle Entwicklungsansatz dahinter, der es in dem augenblicklichen Ausmaß wachsen lässt. Um mehr und mehr physische Geräte vernetzen zu können, werden vermehrt IoT-Bausteine in das bestehende Internet integriert. Die Technologien, die sich hinter diesen IoT-Bausteinen befinden, lassen sich in folgende Kategorien unterteilen (Romdhani et al. 2015):

- Sensortechnologien mit deren Hilfe die zur Weiterverarbeitung benötigten Daten generiert und erfasst werden. Im Bereich IoT spielen kabellose Sensoren eine entscheidende Rolle bei der Erfassung und Übertragung von Daten.

- Die Middleware-Schicht zur Verarbeitung und Weitergabe der erfassten Rohdaten. Sie stellt die Schnittstelle zwischen der physischen Schicht (Hardware) und der Anwendung (Software) dar und bietet sowohl AnwenderInnen als auch EntwicklerInnen eine Abstraktionsschicht, um eine anwender- und bedienerfreundliche Interaktion zu ermöglichen. So kann AnwenderInnen und EntwicklerInnen eine Interaktion ohne fundierte Kenntnisse der tiefer liegenden Schichten ermöglicht werden.

- Die physischen Endgeräte, die letztlich tatsächlich mit der Umwelt interagieren, werden in der Literatur als sogenannte wirkende Technologien (Aktoren) bezeichnet und können als direkte, physische Erweiterung der IoT-Anwendungen angesehen werden. IoT erweitert die uns umgebenden Objekte mit weitreichenden Fähigkeiten zur Kommunikation und Datenverarbeitung.

Diese Kategorisierung zeigt, dass IoT den AnwenderInnen mehr als eine bloße digitale Unterstützung bietet. Das „Internet of Things“ wird zukünftig vielmehr in der Lage sein, unmittelbar, in einem größeren Ausmaß als aktuell, mit unserer physischen Umwelt zu interagieren und unsere reale Welt direkt zu beeinflussen.

2.2.1 Einsatzbereiche

Das “Internet of Things“ bereichert bereits viele Bereiche unseres alltäglichen Lebens. In der praktischen und flächendeckenden Anwendung reichen oftmals bereits einige der technischen Komponenten aus die das Internet der Dinge ausmachen (Mattern, Friedemann, Flörkemeier, Christian 2010). Im Folgenden sind einige Beispiele angeführt in denen IoT bereits unser tägliches Handeln beeinflusst:

- Logistik:

  Logistikunternehmen setzen bereits seit einigen Jahren auf flächendeckende und voll automatisierte Paketverfolgung. Die Logistikunternehmen bieten ihren KundInnen dabei die Möglichkeit den aktuellen Status und Standort ihrer Pakete in Echtzeit zu erfragen. Logistikunternehmen nutzen dafür oft QR-Tags oder andere 2D-Codes, die automatisiert an den Knotenpunkten eingelesen werden und in ein zentrales System zur Weiterverarbeitung geleitet werden. PaketempfängerInnen können so den Status ihrer Lieferung jederzeit in Echtzeit über eine entsprechende Website abrufen und verfolgen.

- Büromaterial:

  IoT hat das Potential uns in vielen Bereichen des täglichen Lebens zu unterstützen und uns wiederkehrende, lästig erscheinende Arbeiten abzunehmen. Durch den Dienst „Instant Ink“ beispielsweise, vom US-Amerikanischen PC- und Druckerhersteller HP Inc., ist der Drucker in der Lage, den Füllstand seiner Tintenpatronen zu überwachen (HP-Inc. 2017). Unterschreitet der Füllstand eine zuvor definierte Untergrenze, wird durch das Gerät eigenständig eine Bestellung beim Zulieferunternehmen aufgegeben. Je nach Druckverhalten der BenutzerInnen entscheidet das Gerät automatisch welche Art und Größe von Tintenpatrone den Bedürfnissen der VerbraucherInnen am besten entspricht. EndnutzerInnen müssen lediglich die automatisch gelieferte Patrone in Empfang nehmen und gegen die leere Patrone austauschen.

- Smart Home:

  Ein weiterer großer Einsatzbereich von IoT zeigt sich im Bereich des familiären Zusammenlebens. Viele Herstellerunternehmen moderner Heizungsanlagen für Einfamilienhäuser bieten bereits maßgeschneiderte Lösungen zur Anbindung ihrer Systeme an das Internet. Dadurch ermöglichen sie es ihren KundInnen, die Systeme ortsunabhängig zu bedienen und zu beeinflussen. Zusätzlich sind die Geräte dadurch in der Lage, selbstständig auf sich ändernde Wetterbedingungen zu reagieren und die Heizleistung entsprechend anzupassen, ohne dass ein manueller Eingriff notwendig ist.

- Gesundheitswesen:

  Der Begriff Ambiet Assistet Living (AAL, auch altersgerechte Assistenzsysteme für ein umgebungsunterstütztes, gesundes und unabhängiges Leben) bezeichnet den Einsatz von modernen Kommunikations- und Informationstechnologien mit dem speziellen Ziel, den Alltag von SeniorInnen mit körperlichen Beeinträchtigungen zu erleichtern (Memon et al. 2014). AAL soll dazu beitragen, ihren Alltag einfacher und sicherer zu gestalten, um möglichst lange ein selbstständiges Leben führen zu können. AAL nutzt dabei das Potential von IoT, indem neben der Auswertung der Vitalwerte, weitere Funktionen implementiert werden. Beispielsweise ist es möglich, SeniorInnen mit Sturzsensoren

auszustatten, die im Anlassfall automatisch einen Notruf absetzen können. Durch das ständige Überwachen von Vitalwerten und der Protokollierung von Patientendaten ist man darüber hinaus in der Lage, Medikamentendosen tagesaktuell an den jeweiligen Gesundheitszustand des Patienten anzupassen. Die erforderlichen Dosen werden den PatientInnen von einem Spender automatisiert ausgehändigt. Durch diese und weitere Maßnahmen ist es möglich, den SeniorInnen ein hohes Maß an Eigenständigkeit zu bieten.

Genannte Beispiele zeigen wie vielfältig die Einsatzbereiche von IoT sein können und welch großes Potential sich dahinter verbirgt. Sie zeigen aber auch wie wichtig eine umfassende Absicherung von IoT-Geräten gegen unerlaubte Zugriffe ist. Diese These ist darauf begründet,dass IoT-Geräte oft in Bereichen eingesetzt werden in denen beispielsweise personenbezogene Daten oder sensible Firmeninformationen übertragen und verarbeitet werden. In unserer heutigen Zeit der Informationsgesellschaft wohnt solchen Daten ein enormes wirtschaftliches Potential inne. Daher ist sowohl dem Schutz und der Sicherung dieser Informationen als auch den Geräten, die diese Informationen verarbeiten, eine hohe Priorität zuzuschreiben.

2.2.2 Endgeräte

So unterschiedlich die Einsatzbereiche des „Internet of Things“ sind, so unterschiedlich und differenziert ist auch die äußere Erscheinung der Endgeräte mit denen wir im Kontext von IoT konfrontiert werden. Bei näherer Betrachtung sind sich IoT-Geräte im Kern jedoch sehr ähnlich. Erste Gemeinsamkeit ist, dass nur sehr begrenzte Ressourcen zur Verfügung stehen. Im Vergleich zu herkömmlichen PCs, die ausreichend Ressourcen bereitstellen, um gängige Betriebssysteme wie Linux, BSD oder deren Derivate zu unterstützen, bieten IoT-Geräte nur sehr eingeschränkt die Möglichkeit Standard-Betriebssysteme in vollem Ausmaß zu betreiben. Sehr häufig werden für IoT-Geräte „System on a Chip“-Komponenten verwendet, wie sie auch in der Mobiltelefonindustrie zu finden sind. Die Bandbreite ist weitreichend und breit gefächert. Folgende Merkmale sind jedoch charakteristisch für einen großen Teil der aktuell am Markt anzufindenden IoT-Lösungen (Halang und Unger 2016):

- Schnittstellen:

  IoT-Geräte sind einerseits mit dem Internet verbunden und nehmen andererseits über unterschiedlichste Sensoren Informationen aus der Umwelt auf. Diese Informationen werden verarbeitet, um dementsprechend über verschiedene Aktuatoren mit der Umwelt zu interagieren. Viele IoT-Geräte besitzen jedoch nur eine beschränkte Anzahl an Schnittstellen, um diverse Anforderungen umzusetzen. Dazu zählen neben Netzwerkinterfaces I/O Standards wie SPI oder I2C.

- Speicher:

  IoT-Geräte verfügen im Vergleich zu herkömmlichen PCs für den Privatgebrauch über verhältnismäßig wenig Arbeitsspeicher (RAM). Während bei Rechnern für den Privatgebrauch heutzutage bereits häufig 32GByte RAM zu finden sind, begnügen sich IoT-Geräte meist mit 512Mbyte bis 1GByte.

- Prozessoren:

  In vielen IoT-Geräten werden kostengünstige Mikrocontroller und ARM-basierte Prozessoren verbaut. ARM-basierte Prozessoren besitzen den Vorteil, dass sie sehr energieeffizient betrieben werden können, da sie nicht an das Stromnetz gekoppelt sind, sondern über Batterien oder Photovoltaikzellen mit Energie versorgt werden.

2.2.3 Betriebssysteme

Das Aufkommen des „Internet of Things“ erfordert den Einsatz von Software, die in der Lage ist, die Hardwareressourcen von Kleinstgeräten optimal zu verwalten. Um diese speziellen Anforderungen von IoT-Geräten und den zur Verfügung stehenden Hardwareressourcen gerecht zu werden, gibt es bereits Betriebssysteme, die speziell für den Einsatz von IoT-Geräten optimiert worden sind. Abbildung 2 zeigt die Verteilung der derzeit gängigsten Betriebssysteme für IoT-Geräte.

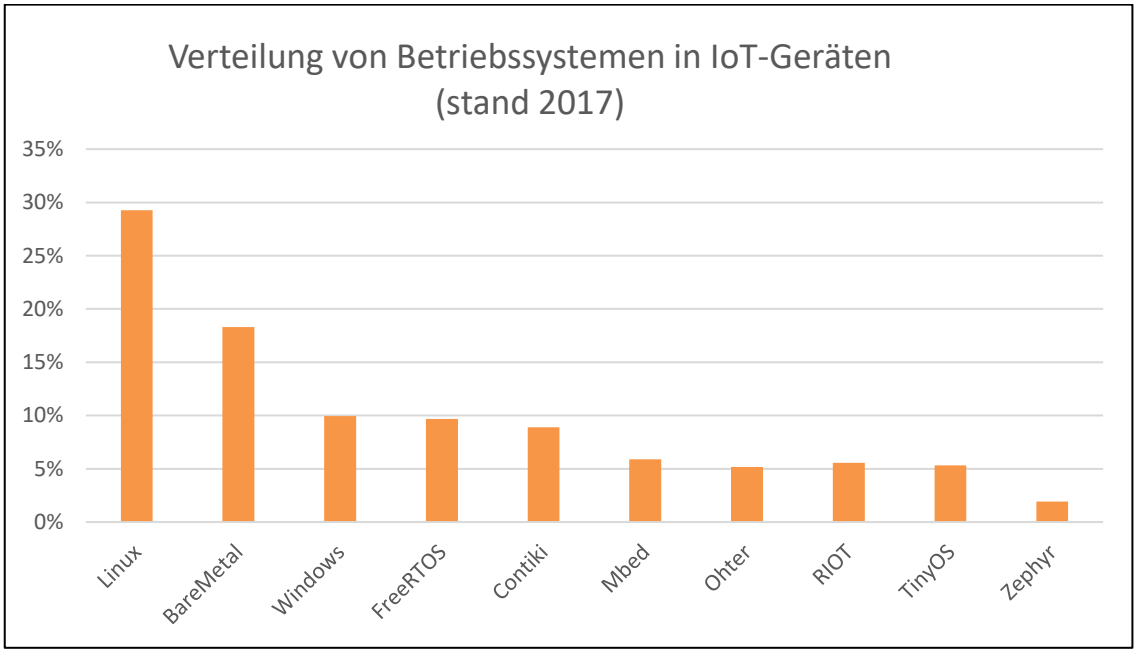


Abbildung 2: Betriebssysteme auf IoT-Geräten (in Anlehnung an Eclipse Foundation 2017).

Das am weitesten verbreitete System stellt dabei Linux dar, gefolgt von BareMetal, ein auf einem Exokernel basierendes Betriebssystem, das von „Return Infinity“ entwickelt wurde. Auch der IoT-Ableger von Windows (Windows 10 IoT Core) ist auf aktuellen IoT-Geräten weit verbreitet. Jedoch strömen vermehrt andere Distributionen auf den Markt, die sich wachsender Beliebtheit bei AnwenderInnen und EntwicklerInnen erfreuen.

Im Folgenden soll auf drei dieser Betriebssysteme näher eingegangen werden.

- ROIT:

  RIOT ist ein Linux-ähnliches Betriebssystem für IoT-Geräte (Halang und Unger 2016). Es verfügt über einen Microkernel, beherrscht Datenverschlüsselung und zeichnet sich durch einen sehr geringen Stromverbrauch und moderate Ressourcenanforderungen aus. Das Betriebssystem wurde 2013 auf der „IEEE INFOCOM“ erstmals vorgestellt und erfreut sich seither einer wachsenden Beliebtheit. RIOT verspricht eine modulare Struktur mit anwenderfreundlicher Abstraktionsschicht. Diese baut weitgehend auf einer „Portable Operating System Interface“ (POSIX) - Kompatibilität mit Unterstützung von C und C++ auf. RIOT ist auf der freien Lizenz GNU Lesser General Public License (LGPLv2) veröffentlicht und ist dahingehend bestrebt, im Bereich IoT einen ähnlich hohen Stellenwert einzunehmen wie Linux im Bereich des Internets. In diesem Zusammenhang zielt RIOT darauf ab, ein modernes, umfangreiches, evolutives und sicheres cyber-

physikalisches Ökosystem zu betreiben, das aus heterogenen IoT-Geräten, verteilten Prozessen und Anwendungen besteht, die sich nahtlos untereinander sowie mit der Cloud verbinden und Standard-Netzwerk-Stacks nutzen können (einschließlich IPv6 Interoperabilität).

- Windows 10 IoT Core:

  Auch Windows hat mit „Windows 10 IoT Core“ ein Betriebssystem auf den Markt gebracht, das speziell für den Einsatz in Kleinstgeräten optimiert wurde. Zu den Stärken von Windows 10 IoT Core zählen Microsofts Entwicklerwerkzeuge aus der Visual Studio Reihe und die darauf abgestimmten IoT-Cloud Dienste von Microsoft Azure. Dadurch ist der Einstieg in IoT-Produkte für Unternehmen, die bereits Produkte für die Windowsplattform entwickeln, einfach zu bewältigen.

- Linux Derivate:

  Mit dem Aufkommen des „Internet of Things“ haben sich auch einige Linux Derivate entwickelt, die speziell für den Einsatz in kleinen, ressourcenschwachen Endgeräten optimiert wurden. Diese stellen teilweise Weiterentwicklungen von gängigen Distributionen für mobile Endgeräte dar, wie beispielsweise Android, zum anderen sind es speziell für den Einsatz in IoT-Geräten optimierte Versionen von gängigen Systemen für PCs wie Ubuntu oder Debian. Das am weitesten verbreitete Derivat von Linux stellt augenblicklich Raspian dar (Abbildung 3), gefolgt von Ubuntu und UbuntuCore. Raspian ist ein auf Debian basierendes Betriebssystem, das gezielt für die zur Verfügung stehenden Ressourcen des Ein-Platinen-Computers RasperryPI optimiert wurde. Einer der Gründe für die weite Verbreitung des RaspberryPi ist der hohe Anteil an IoT-Geräten, die unter Raspian laufen.

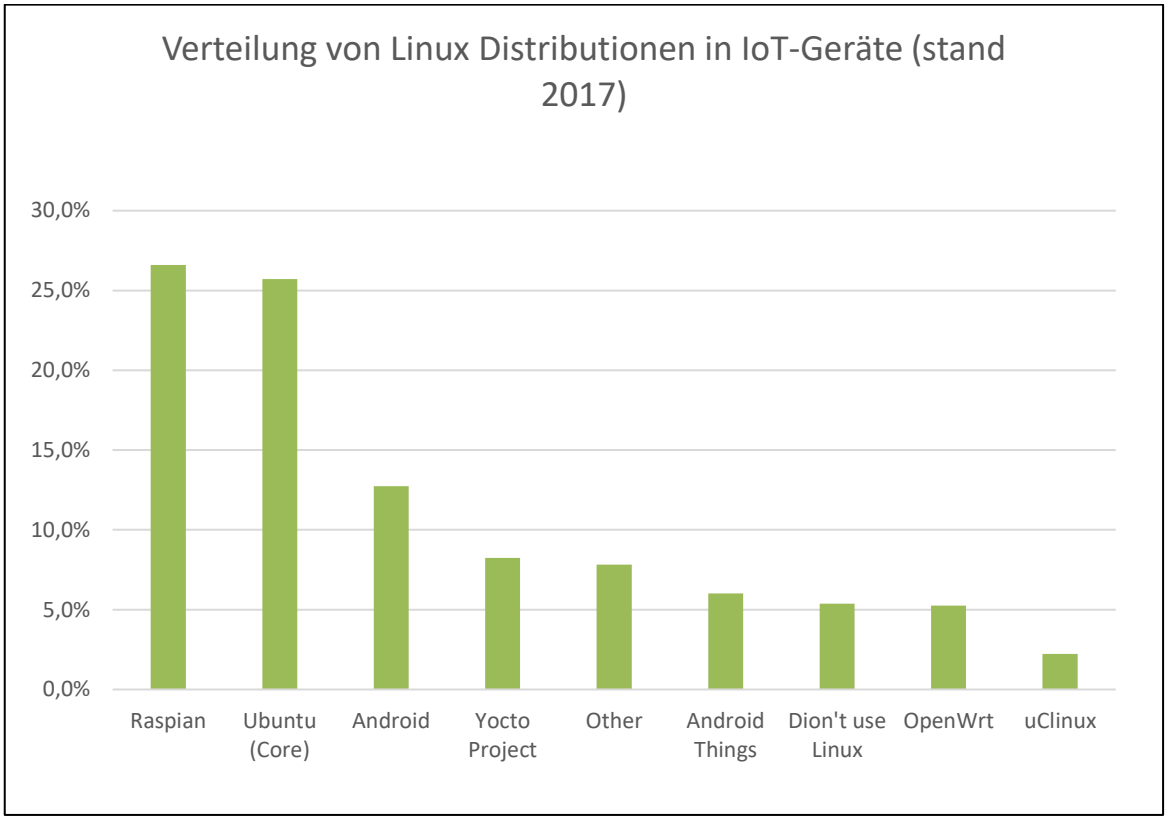


Abbildung 3: Linux Distributionen für IoT-Geräte (vgl. Eclipse Foundation 2017)

2.3 Zukünftige Entwicklung

Zum Thema IoT wird weltweit in unterschiedlichsten Einrichtungen umfassende Forschungsarbeit betrieben. Auf europäischer Ebene erfolgt dies beispielsweise im Internet Research Cluster on the Internet of Things (IERC). IERC verfolgt das Ziel, eine europaweit einheitliche Vision von IoT zu manifestieren und einen breit gefächerten, aber gefestigten Konsens über die technischen Möglichkeiten zu generieren, um diese Vision umzusetzen (IERC 2016). IERC sieht IoT-Technologien als Schlüsselindikatoren des digitalen Binnenmarktes, die einen erheblichen Einfluss auf die Schaffung von Arbeitsplätzen und das Wachstum haben werden. Diese Prognose zeigt wiederum, wie wichtig eine umfassende Absicherung von IoT-Geräten gegenüber fremden Zugriffen ist und welch großes Augenmerk auf IoT-Security gelegt werden sollte, insbesondere in Anbetracht zukünftiger Entwicklungen und Veränderungen im Internet.

3 INTERNET PROTOKOLL VERSION 6

Seit der Geburtsstunde des Internets bildet das Internet Protokoll Version 4 (IPv4) die Basis zur Übertragung einzelner Datenpakete zwischen zwei Netzwerkknoten. IPv4 wurde in den frühen 70er Jahren entwickelt. Ziel war es, staatliche und universitäre Netzwerke in den USA miteinander zu verbinden. Ging man zu Beginn der Entwicklung davon aus, dass eine Adressierung der einzelnen Knoten mit 32bit eine ausreichend große Anzahl an Adressen bereit stellen würde, um alle im Netz befindlichen Geräte eindeutig zu identifizieren, so wurde schnell deutlich, dass die rasante Entwicklung des Internets und dessen schnelles Wachstum einen größeren Adresspool unumgänglich machten. Abbildung 4 verdeutlicht, dass bei aktuell 3,42 Milliarden InternetnutzerInnen die maximal mögliche Anzahl an IPv4-Adressen von 4.294.967.296 bald erreicht ist. Die steigende Anzahl der NutzerInnen und das rasante Wachstum des Internets haben sehr schnell zur Erweiterung des bestehenden IPv4-Protokolls geführt**,** um den drohenden Problemen Herr zu werden. Es wurden Verfahren wie zum Beispiel Network Address Translation (NAT) entwickelt. Durch NAT wurde es möglich**,** mehrere private IP Adressen hinter einer einzigen öffentlichen Adresse zu verwenden (Hagen 2016). Obwohl viele administrierende Personen in NAT einen Sicherheitszuwachs sehen, sei angemerkt, dass NAT kein Sicherheitsfeature ist. In den Anfängen des Internets wurde primär nur dazu entwickelt, einem raschen Ende an verfügbaren IPv4-Adressen entgegen zu wirken. Dass es durch die Verwendung von NAT möglich ist, die Topologie eines privaten Netzwerkes hinter einer öffentlichen Adresse zu verbergen, ist lediglich als ein positiver Nebenaspekt anzusehen.

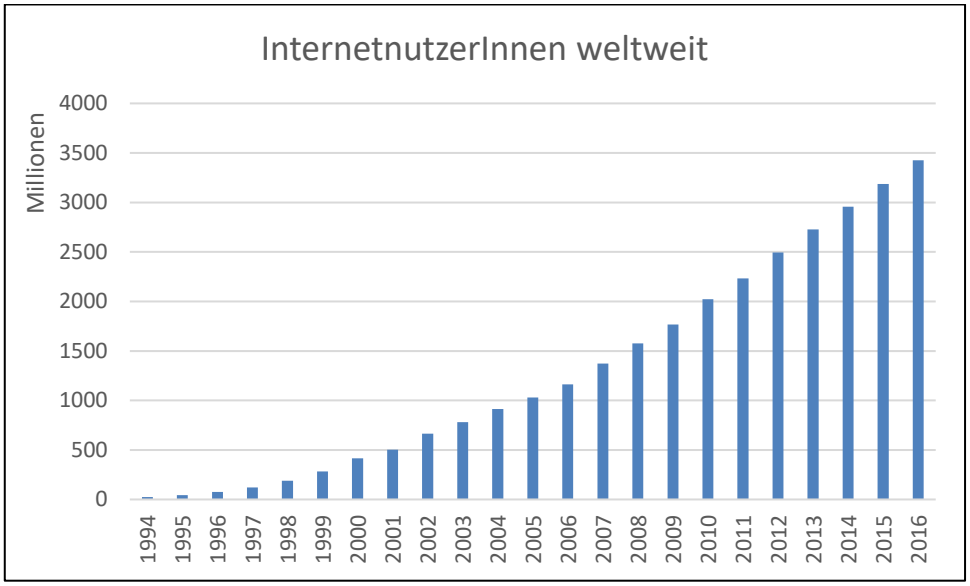


Abbildung 4: Entwicklung von InternetnutzerInnen weltweit (Quelle: internetlivestats.com)

Die aktuelle Entwicklung jedoch, hin zu einem „Internet der Dinge“, beschleunigt diesen Vorgang eines Engpasses an IP Adressen noch weiter, da jede, der in der Abbildung dargestellten InternetnutzerInnen, mehrere internetfähige Endgeräte betreiben wird. Die drohende Erschöpfung von IP-Adressen ist einer der Gründe, der die Entwicklung eines neuen Netzwerkprotokolls nötig machte.

3.1 Grundlagen zu IPv6

Im folgenden Abschnitt wird auf die Grundzüge von IPv6 näher eingegangen. Besonderes Augenmerk soll hierbei auf eine detaillierte Darstellung der Sicherheitsaspekte, die in IPv6 implementiert sind, gelegt werden, da diese die Grundlage für das weitere Vorgehen in dieser Arbeit darstellen.

3.1.1 Entstehung

IPv6 ist eine Evolution von IPv4 und wurde basierend auf den umfangreichen Erfahrungen mit IPv4 entwickelt. Das weltweit älteste IPv6-Netzwerk war das sogenannte 6Bone, das mehr als 1000 Hosts in über 50 Ländern der Erde miteinander verband.

Bei der Entwicklung von IPv6 wurde besonders darauf geachtet, Bewährtes in das neue Protokoll mit aufzunehmen und bekannte Einschränkungen von IPv4 zu beheben (Hagen 2016). Dies liegt darin begründet, dass man zur Zeit der Entwicklung von IPv4 nicht in der Lage war, sich vorzustellen, dass das Internet zu einem weltumspannenden Netzwerk heranwachsen würde, dessen teilnehmende Geräte nicht nur Computer, sondern auch tragbare Endgeräte und auch, durch das „Internet of Things“, Gegenstände des täglichen Lebens sind. In diesem Sinne soll IPv6 auch das Protokoll sein, das in der Lage ist, der Wachstumsrate des Internets und den Anforderungen zukünftiger Dienste gewachsen zu sein.

Neben der Zurverfügungstellung eines um ein Vielfaches vergrößerten Adressraumes, wurde IPv6 auch dafür entwickelt, weitere Nachteile und Schwächen von IPv4 auszugleichen. Bei der Entwicklung von IPv6 wurden daher folgende Ziele verfolgt (Badach und Hoffmann 2015):

- Vergrößerung und hierarchische Strukturierung

  IPv6-Adressen sind im Vergleich zu IPv4-Adressen viermal so lang, also 128 statt bisher 32 Bit. Hierbei bilden die ersten Bits den Adressbereich der Adresse (Scope). Nachfolgende Bits werden zum Zweck des Routings verwendet. Die letzten 62 Bits dienen letztlich der Identifikation des Netzwerkinterfaces.

- Verbesserung der Header Strukturen

  Der Header des neuen Internet Protokolls wurde gegenüber der Vorversion deutlich verbessert. Besonderer Wert wurde auf eine klare Unterteilung von notwendigen und optionalen Angaben gelegt. Mit IPv6 ist es nun möglich, optionale Angaben nur bei Bedarf in speziellen Headern (sog. Extension Header) zu übertragen.

- Gültigkeitsbereich von IPv6-Adressen

  Für jede IPv6-Adresse wird ein eigener Gültigkeitsbereich definiert (der sogenannte Scope). Einer öffentlichen Adresse, wie sie in IPv4 verwendet wird, entspricht dabei eine IPv6-Adresse mit globalem Scope. *Unique-Local Unicast Adresses* besitzen nur Gültigkeit im lokalen Linksegment und sind daher ähnlich den privaten Adressen von IPv4. Dagegen stellen sogenannte *Link-Lical Unicast Adresses* technische Adressen dar**,** die für den Ablauf des IPv6-Protokolls notwendig sind. Darüber hinaus werden in IPv6 auch spezielle Adresstypen definiert**,** um verschiedenste Fälle zu unterstützen, beispielsweise solche in denen IPv4 und IPv6 gemeinsam eingesetzt werden.

- Autokonfiguration

  IPv6-fähige Rechner sind in der Lage, die Netzwerkkonfiguration pro Interface nach dem Plug-and-Play-Prinzip eigenständig einzurichten. Um diese Funktion zu gewährleisten, muss sich jede IP-Instanz einen „Interface Identifier“ (Interface-ID) merken. Dies führt jedoch zu Änderungen der Socket-Schnittstelle.

- Verbesserung der Sicherheit

  Um eine verbesserte Sicherheit gegenüber IPv4 zu gewährleisten, wurden zwei Extension-Headers vorgesehen: „Encapsulation Payload Security“ und „Authetication Header“. Die Tatsache, dass diese Extension-Header auch in IPv4 eingesetzt werden können, hat zur Entstehung des ergänzenden Protokolls IPSec (IP Security) in IPv4 geführt.

- Default Minimum-MTU-Size von 1280 Byte

  Per Definition verlangt ein IPv6-Netz, dass die Data-Link-Schicht in der Lage sein muss, IPv6-Pakete bis zu einer Größe von 1280 Byte ohne jegliche Einschränkung zu übermitteln (RFC 2460 1998). Dieser Wert lag bei IPv4 lediglich bei 576 Byte. Das bedeutet, dass daher auch UDP-Nachrichten, wie zum Beispiel bei DNS im Einsatz, deutlich größer sein können.

3.1.2 IPv6 Datagramm

Die EntwicklerInnen hinter IPv6 verfolgten bei ihrer Arbeit unter anderem das Ziel, den Header leichtgewichtiger und verständlicher zu gestalten. Folgender Abschnitt zeigt die einzelnen Teile des IPv6-Headers und dessen Funktionen. Abbildung 5 zeigt den grundsätzlichen Aufbau des IPv6-Headers (RFC 2460 1998).

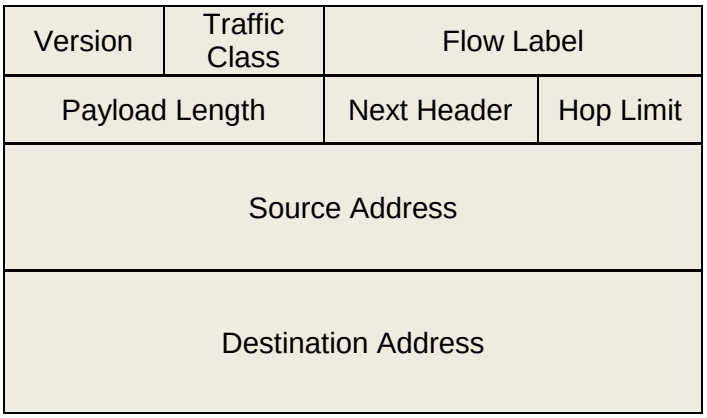


Abbildung 5: Header eines IPv6-Paketes.

- Version:

  Das erste Feld beinhaltet die Versionsnummer des Protokolls. Das Feld hat eine Länge von 4 Bit (0.5 Byte) und bei IPv6-Header einen Standardwert von 6.

- Traffic Class:

  Die darauffolgenden 8 Bit (1 Byte) bilden die Traffic Class (auch Priority Field genannt). Durch dieses Feld ist es der Quelle möglich, die Priorität seiner Pakete zu klassifizieren. Die Werte der Priorität sind in zwei Bereiche unterteilt:

  0-7: Datenverkehr der sich bei Überlastung verlangsamt
  8-15 Datenverkehr mit konstanter Übertragungsrate (z.B. für Echtzeitanwendungen)

- Flow Label:

  Dieser neu eingeführte Teil des Headers, mit einer Länge von 20 Byte, dient zur Flusskennzeichnung von Paketen. Damit werden Pakete gekennzeichnet, die einer besonderen Behandlung durch IPv6 Router bedürfen. Der Standardwert des Feldes beträgt 0. Empfängt ein Router ein Paket mit einem Flow Label ungleich 0, so werden diese entsprechend speziell behandelt.

- Payload Length:

  Durch das 16 Bit (2 Byte) große Feld Payload Length wird die Länge der Nutzdaten ohne Header angegeben. Im Unterschied zu IPv4, wo dieses Feld auch die 40 Byte des Header beinhaltet, wird in IPv6 die Länge des Headers von dieser Angabe ausgenommen.

- Next Header:

  Dieses 8 Bit umfassende Feld gibt den Typ des Headers an, der dem IPv6-Header direkt folgt. Um den IPv6-Header gegenüber IPv4 zu vereinfachen, wurden diverse Erweiterungsheader definiert. Dieses Feld gibt an, welcher Erweiterungsheader folgt. Ist es der letzte IP-Header, so ist dem Feld zu entnehmen, welchen Handler des Transportprotokolls das Paket passiert (z.B. UDP, TCP).

- Hop Limit:

  Dieser Bereich entspricht dem Feld “Time to Life” eines IPv4-Headers. Es gibt die Anzahl an Knoten an, welche das Paket passieren kann, bevor es verworfen wird. Bei jedem „Hop“ wird der Wert um eins verringert. Erreicht der Wert dieses Feldes 0, so wird das Paket nicht mehr weitergeleitet. Die Länge dieses Feldes beträgt 8 Bit (1 Byte).

- Source Address:

  Dieses Feld enthält die 128 Bit (16 Byte) lange IPv6-Adresse des Absenders.

- Destination Address:

  Dieses Feld enthält die 128 Bit (16 Byte) lange IPv6-Adresse des beabsichtigten Empfängers.

In einigen Fällen sind zusätzliche Informationen über das zu übertragende Paket notwendig. Für diese Fälle wurden eigens Erweiterungs-Header geschaffen, um diese Zusatzinformationen bereitzustellen. Aktuell (stand 2017) sind sechs Erweiterungs-Header definiert:

- Hop-by-Hop Options:

  Dieser Header enthält Informationen über das Paket, die von jedem Knoten entlang des Pfades untersucht werden müssen. Der Hop-by-Hop-Header wird durch den Wert 0 des Next-Header Feldes im IP-Header identifiziert.

- Routing (Type 0):

  Der Routing-Header wird von einer IPv6-Quelle verwendet, um einen oder mehrere Knotenpunkte aufzulisten, die entlang des Weges vom Paket besucht werden müssen. Diese Funktion ähnelt der Option „Loose Source und Record Route“ von IPv4. Der Routing-Header wird durch den Wert 43 des Next-Header Feldes im IP-Header identifiziert.

- Destination Option:

  Der Destination-Option-Header enthält Informationen, die nur von den Zielknoten des Paketes untersucht werden müssen. Der Destination-Option-Header wird durch den Wert 60 des Next-Header Feldes im IP-Header identifiziert.

- Fragment:

  Der Fragment-Header wird genutzt, um Pakete von einer IPv6-Quelle zu übertragen, die größer sind als die Maximum Transfer Unit (MTU). Im Unterschied zu IPv4 wird die Fragmentierung in IPv6 nur von Quellknoten durchgeführt, nicht jedoch von Routern entlang des Transportpfades. Der Fragment-Header wird durch den Wert 44 des Next-Header Feldes im IP-Header identifiziert.

- Authentication:

  Auf diesen Extension Header wird gesondert im nächsten Abschnitt detailliert eingegangen.

- Encapsulation Security Payload:

  Auf diesen Extension Header wird gesondert im nächsten Abschnitt detailliert eingegangen.

3.1.3 Limitation von IPv6

Bei der Entwicklung von IPv6 wurde besonders darauf geachtet, bekannte Einschränkungen aus IPv4 zu überwinden. Hauptaugenmerk wurde dabei auf einen erweiterten Adressraum und einfachere Konfigurierbarkeit gelegt. Obwohl IPv6 durch seine 128 Bit-Adressierung einen schier unerschöpflichen Pool an Adressen bereitstellt, sind unter dem neuen Protokoll einige Adressbereiche reserviert und stehen somit nicht direkt zur Verfügung. Folgender Abschnitt gibt einen Einblick in reservierte und spezielle Adressbereiche von IPv6.

3.1.3.1. Spezielle Adressbereiche

- 0:0:0:0:0:0:0:0 oder ::/128

  Diese „nicht spezifizierte Adresse“ darf keinem Knoten zugeteilt werden. Dieser Wert gibt das Nichtvorhandensein einer IPv6-Adresse an. Diese Adresse findet sich im Source-Address-Field eines initial von einem Host gesendeten IPv6 Paketes, das noch keine IPv6-Adresse erlernt hat (RFC 4291 2006).

  Eine nicht spezifizierte Adresse darf nicht als Zieladresse in einem IPv6 Paket oder IPv6 Routing Header verwendet werden. IPv6 Pakete mit einer nicht spezifizierten Quelladresse werden von einem Router nicht weitergeleitet.

- 0:0:0:0:0:0:0:1 oder ::1/128

  Diese Adresse stellt die unter IPv6 verwendete Loopback-Adresse dar. Jedes an diese Adresse gesendete Paket soll den Host nicht verlassen, sondern selbst an diesen geschickt werden. Deswegen darf diese Adresse keiner physikalischen Schnittstelle zugewiesen werden. Sie wird wie eine Link-Local Adresse behandelt und kann als Link-Local-Unicast Adresse einer virtuellen Schnittstelle angesehen werden (RFC 4291 2006).

  Ein Paket mit dieser Zieladresse darf niemals außerhalb eines Knotens gesendet werden und wird von keinem Router weitergeleitet. Auf Schnittstellen empfangene Pakete mit dieser Adresse als Zieladresse werden daher fallen gelassen.

- 0:0:0:0:0:FFFF:a.b.c.d/96 *oder* ::FFFF:a.b.c.d/96

  Diese Sonderform der IPv6-Adressen sind IPv4-Adressen, die als IPv6-Adressen dargestellt werden. Diese Adressen kommen bei manchen IPv6 Serverprogrammen zum Einsatz, die in einem reinen IPv4Netzwerk arbeiten. Dem 96 Bit langem Präfix folgt dabei die 32 Bit lange IPv4-Adresse. Diese Adressen finden in der Praxis häufig bei der sogenannten 6to4-Tunneling Anwendung.

- Anycast-Adressen

  Eine Anycast-Adresse adressiert eine bestimmte Gruppe von Rechnern in einem Netzwerk, beispielsweise alle Router. Sendet ein Netzwerkknoten eine Nachricht über diese Adresse, antworten hingegen nicht alle im Netz vorhandenen Router, sondern nur derjenige, der per Routing-Tabelle am nächsten oder am besten erreichbar ist.

  IPv6 verwaltet Unicast- und Multicast-Adressen in getrennten Adressbereichen. Anycast-Adressen sind hingegen Teil des jeweiligen Unicast-Bereichs und unterscheiden sich daher syntaktisch nicht von ihnen. Allerdings können sie nie als Quelladresse dienen, sondern nur als Ziel einer Übertragung. (RFC 2373 1998) definiert beispielsweise eine Anycast-Adresse für Router in einem Netz, die aus dem jeweiligen Netzwerk-Präfix besteht und deren Host-Teil mit Nullen aufgefüllt ist. (RFC 2526 1999) beschreibt weitere Anycast-Adressen, die beispielsweise bei Mobile-IPv6 Verwendung finden.

3.1.3.2. Reservierte Adressbereiche

Eine Vielzahl an Adressbereichen ist für die derzeitige Verwendung nicht bestimmt, da sie bereits von diversen Institutionen oder für zukünftige Funktionalitäten reserviert wurden. Viele Adressbereiche sind beispielsweise von der Internet Engineering Task Force (IETF) reserviert.

Tabelle 1 zeigt die von der IETF reservierten Adressbereiche.

0100::/8	0200::/7	0400::/6
0800::/5	1000::/4	4000::/3
6000::/3	8000::/3	A000::/3
C000::/3	E000::/4	F000::/5
F800::/6	FE00::/9	FEC0::/10

Tabelle 1: Reservierte IPv6-Adressbereiche

Adressen des Bereiches 2000::/3 sind global gültige und eindeutige Unicast-Adressen. Sie werden im Internet geroutet und sind bislang die einzigen erhältlichen IPv6 Präfixe.

Adressen des Bereiches 2001:D88::/32 sind reservierte Adressen, die ausschließlich zu Dokumentationszwecken zur Verfügung stehen.

Adressen des Bereiches 2001:0000::/32 sind reserviert für sogenanntes Toredo-Tunneling gemäß (RFC 3849 2004). Toredo ist ein Dienst, der es Knoten, die sich hinter einer oder mehreren IPv4-Adressen (NAT) befinden, ermöglicht, IPv6 Konnektivität zu erhalten.

Adressen aus dem Bereich 2001:10::/28 stellen keine IPv6-Adressen dar, sondern sind „Overlay Routable Cryptographic Hash Identifiers“ (ORCHID). Diese Pseudoadressen werden nicht geroutet und sollten niemals im öffentlichen Netz in Erscheinung treten. ORCHID sind experimentale IPv6 ähnliche Identifikationsnummern für Programmierschnittstellen (API).

Adressen mit dem Präfix FC00::/7 sind reserviert als Unique-Local-Unicast Adressen. Diese Adressen sollen per Definition nur in lokalen, abgegrenzten Netzen eingesetzt werden. Laut (RFC 4193 2005) sollen solche Adressen weder von Routern noch von einer Firewall in das globale Internet durchgereicht werden.

Der Präfix FF00::/8 definiert in IPv6 verwendete Multicast Adressen. Multicast Adressen sind Adressen, die eine Gruppe von NetzwerkteilnehmerInnen adressieren. Das Präfix einer Multicast Adresse gibt unter anderem darüber Auskunft, ob eine Adresse dauerhaft oder dynamisch zugewiesen wurde und welche Reichweite sie hat. Beispielsweise ist die Adresse „FF02::2“ gültig für alle Router eines Standortes.

3.2 Sicherheitsmechanismen von IPv6

Durch das rasche Wachstum des Internets und die Vielzahl der Einsatzgebiete wurde schnell klar, dass Sicherheit höchste Priorität im Netzwerkprotokoll der Zukunft haben wird. IPv6 war somit die ideale Gelegenheit, neue Sicherheitselemente in den Nachfolger von IPv4 zu implementieren. Neben diesen neuen Funktionen beeinflusst das neue Protokoll die Sicherheit auf indirekte Weise. Bisher ist nicht gänzlich sicher, ob dieser Einfluss die Gesamtsicherheit steigert oder nicht (Badamchizadeh und Chianeh 2006). Im folgenden Abschnitt wird auf allgemeine Konzepte, die IPv6 sicherer machen sollen, eingegangen. Danach wird besonderes Augenmerk auf IPSec und die damit verbundenen Technologien gelegt.

3.2.1 Allgemein

Ein radikaler Unterschied zu IPv4 bildet die bereits erwähnte Adressierungsstruktur. Bereits durch die enorme Größe des möglichen Adress-Pooles lassen sich Zugewinne im Bereich Netzwerksicherheit erzielen. Dies ist durch die derzeit verwendeten „break-in“ Techniken von Schadsoftware begründet. Für gewöhnlich werden Netzwerke nach potentiellen Angriffszielen abgesucht. Durch die begrenzte Anzahl an verfügbaren IP-Adressen in einem IPv4-Netzwerk ist dies für AngreiferInnen ohne großen Aufwand rasch zu bewältigen, da sich der zu durchsuchende Adressbereich nur in Größenordnungen von einigen Hundert bis Tausend IP-Adressen bewegt. Dieser Adressbereich lässt sich durch Scannen mittels „Brute-Force“-Methoden, wobei alle möglichen Adressen des Host-Teiles einer IPv4-Adresse durchprobiert werden, sehr rasch abarbeiten. Eine IPv6-Adresse besteht aus dem Präfix und dem Interface Identifier. Dieser Interface Identifier, welcher den Host-Teil einer IPv6-Adresse bildet, besteht immer aus 64 Bit. Daraus ergibt sich, dass das kleinste mögliche lokale Netzwerk aus 2^64 nutzbaren Adressen

besteht. Das bedeutet ~18.5 Trillionen verfügbare Adressen je Netzwerk. Dieser riesige Bereich an möglichen IP-Adressen macht einen Brute-Force Angriff mit heute verfügbaren Ressourcen nahezu unmöglich.

Diesem vermeintlichen Sicherheitszuwachs durch den enormen Adressbereich wird jedoch durch eine Eigenschaft von IPv6 zur einfacheren Konfiguration entgegengewirkt. Um den administrierenden Personen die Arbeit mit IPv6 zu erleichtern, werden IPv6-Adressen automatisch konfiguriert. Dies erfolgt auf Basis der MAC-Adresse des Netzwerkinterfaces. Abbildung 6 zeigt die Erstellung einer IPv6-Adresse aus der MAC-Adresse mittels „Modified EUI-64“ (Extended Unique Identifier).

Im ersten Schritt wird dabei die 48 Bit lange MAC-Adresse in zwei 24 Bit lange Teile geteilt. Dabei bildet der erste Teil, der „Organizationally Unique Identifier“ (OUI), die ersten 24 Bit der modifizierten EUI-63 Adresse und der zweite Teil, der „Network Interface Controler“ (NIC), die letzten 24 Bit.

Die fehlenden 16 Bit in der Mitte der zu bildenden EUI-64 Adresse werden mit folgendem Bitmuster belegt: 1111 1111 1111 1110 (Hexadezimal: FFFE).

Nach diesen beiden Schritten befindet sich die Adresse nun im UEI-64-Format. Um eine modifizierte UEI-64-Adresse daraus zu generieren, wird im letzten Schritt das siebte Bit von links invertiert.

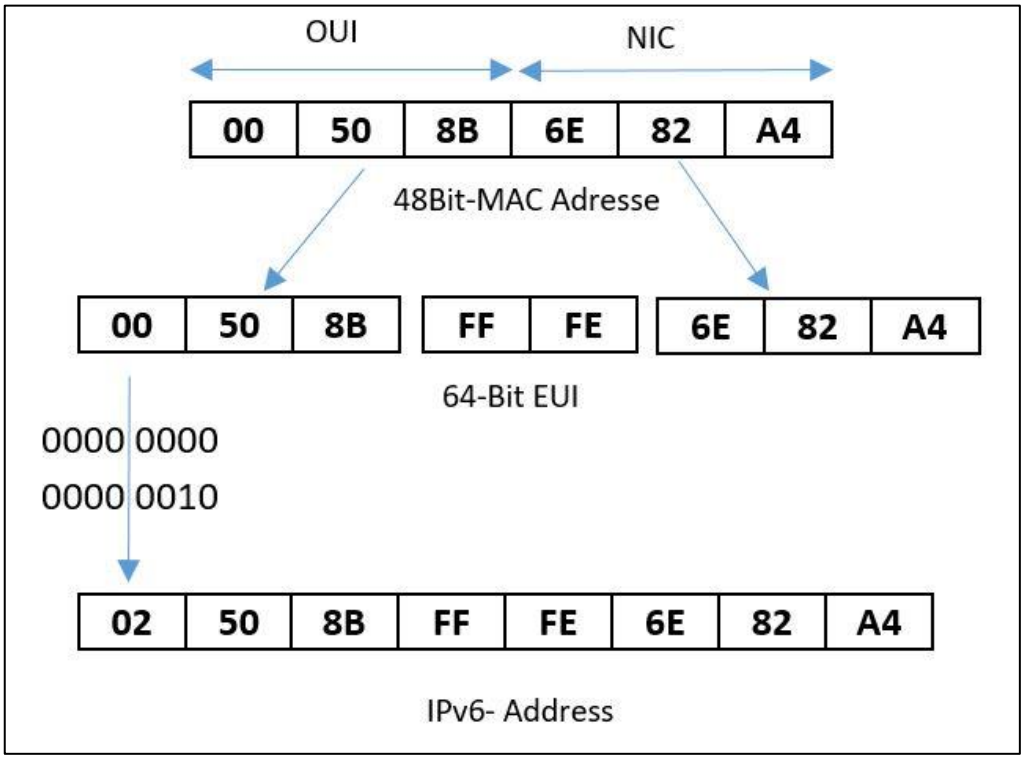


Abbildung 6: Generierung einer IPv6-Adresse mittels "Modified EUI-64"

Das siebte Bit wird gekippt, weil man bei der Festlegung des Adressraumes für MAC-Adressen einen Adressbereich definiert hat, der vom Hersteller frei befüllt werden kann und somit nicht zugewiesen wird. Das zweite Bit des ersten Bytes enthält die Information, ob es sich um eine MAC-Adresse handelt, die von der IEEE zugewiesen, oder vom Hersteller wahllos festgelegt wurde. Ist der Wert des Bits „0“, so wurde die Adresse von der IEEE vergeben und ist „global unique“. Hat das Bit den Wert „1“, ist die Adresse frei gewählt und ist „local unique“. Es ist also nicht gewährleistet, dass die Adresse weltweit einzigartig ist.

Durch diesen Mechanismus ist es AngreiferInnen möglich, den Adressbereich eines IPv6-Netzwerkes drastisch einzugrenzen, wenn man über die verwendete Hardware Bescheid weiß, da der OUI-Teil der MAC-Adresse von unterschiedlichen Herstellern bekannt ist. Trotzdem bleibt der Adressbereich mit 2^24 (16,8 Millionen) möglichen Variablen noch beträchtlich groß, was eine Brute-Force-Attacke erschwert, jedoch mit entsprechender Rechenleistung nach wie vor zu bewerkstelligen ist. Obwohl es möglich ist, durch den großen Adressraum diverse Arten von Angriffen wirksam zu verhindern, wird der Zuwachs an Sicherheit nur durch Verschleierung erreicht und sollte daher nicht als wirksame Prävention angesehen werden. So ist es möglich mittels „Eavesdropping“, dem gezielten Belauschen des Netzwerkverkehrs, an entsprechende Informationen zu gelangen.

Während der große Adressbereich in der Lage ist, einige Attacken zu erschweren oder gar zu verhindern, kann der generierte Mehrwert auch auf Kosten von zum Einsatz kommenden Gegenmaßnahmen gehen. So ist es beispielsweise möglich, dass der breite Adressraum die

Arbeit von Sicherheits-Scannern und Intrusion-Dedection-Systemen (IDS) erschwert und die Wahrscheinlichkeit der Erkennung von korrumpierten Hosts drastisch verringert.

Da IPv6 genügend Adressen bereitstellt, um theoretisch jedem Sandkorn der Erde mehrere Adressen zuzuteilen, ist die Anwendung von „Network Address Translation“ (NAT) in IPv6 nicht mehr nötig. NAT wird in IPv4-Netzen dazu verwendet, um mehrere private IP-Adressen hinter einer öffentlichen IP-Adresse zu verbergen. Diese Netzwerkadressübersetzung ermöglicht es, das zur Neige gehen von IPv4-Adressen hinauszuzögern und genügend Adressen für interne Netzwerke bereitzustellen. Trotz des Nachteils, dass NAT die End-zu-End-Konnektivität aufbricht, bietet NAT den Vorteil, dass das interne Netz vor der Außenwelt verborgen und Zugriffsversuche von außen unterbunden werden können. Manche sehen darin einen großen Vorteil von NAT, und den Wegfall dieser Netzadressübersetzung als großen Verlust welchen IPv6 mit sich bringt. Diese Funktion kann jedoch auch durch moderne Firewalls ausgeübt werden, ohne die Nachteile, die NAT mit sich bringt.

3.2.2 Internet Protocol Secure

Internet Protocol Secure (IPSec) stellt den offensichtlichsten Teil des Sicherheitszuwachses in IPv6 dar. IPSec beschreibt dabei jedoch generelle Sicherheitsmechanismen, die sowohl unter IPv6 als auch IPv4 verwendet werden können. Der Unterschied liegt darin, dass IPSec in IPv4 nachträglich installiert werden muss, während es unter IPv6 meist ein integrierter Bestandteil der Basis-Protokolle ist und somit bei jeder IPv6-Implementierung zur Verfügung steht. (Hagen 2016).

Die ursprüngliche Spezifikation von IPv6 schrieb IPSec zwingend für jeden IPv6-Stack vor. Diese Regelung wurde jedoch gelockert, da dies für Geräte mit geringen Ressourcen (IoT-Geräte) nicht immer möglich ist.

IPSec ist nicht als ein einziger Standard zur Erhöhung der Sicherheit der Netzwerkkommunikation anzusehen. IPSec ist vielmehr eine Sammlung (Framework) an offenen Standards, die das Ziel verfolgen, Daten vertraulich zwischen Endgeräten zu übertragen und dabei Datenintegrität und Datenauthentifizierung zu etablieren.

Folgende Elemente gehören zum IPSec Framework:

- Eine allgemeine Beschreibung von Sicherheitsanforderungen und -mechanismen auf Vermittlungsebene (Networklayer)

- Ein Protokoll für die Verschlüsselung von Daten (Encapsulation Security Payload)

- Ein Protokoll für die Authentisierung (Authentication Header)

- Eine Definition für den Gebrauch kryptographischer Algorithmen für die Verschlüsselung und die Authentisierung

- Eine Definiton von Security Policies und Security Associations für die Kommunikationspartner

- Effektives Schlüsselmanagement (Key Management) zum Austausch der für eine sichere Übertragung notwendigen Schlüsselpaare.

Die Beschreibungen der Sicherheitsanforderungen verfolgen im Wesentlichen folgende Ziele:

- Authentizität:

  Die Identität von AbsenderInnen oder EmpfängerInnen kann von der jeweiligen Gegenseite bestätigt werden. AngreiferInnen ist es somit nicht möglich, sich eine falsche Identität anzueignen.

- Integrität (Unverfälschtheit)

  Jede allfällige Änderung der empfangenen Datenpakete und Informationen wird entdeckt. AngreiferInnen können keine Datenpakete manipulieren, ohne dass es unbemerkt bleibt.

- Vertraulichkeit

  Die übertragenen Daten können weder gelesen noch verändert werden, wodurch gewährleistet wird, dass Dritte keine Möglichkeit haben Daten zu manipulieren oder einzusehen.

- Obligation

  Eine Aktivität wie das Senden, Empfangen oder Löschen von Daten darf von keinem der KommunikationspartnerInnen abgestritten werden.

Um diese Ziele durch IPSec zu erreichen, werden zwei Mechanismen eingesetzt. Einerseits wird durch Verschlüsselung die Vertraulichkeit gewahrt, andererseits werden sichere Checksummen genutzt, um die Integrität von Daten zu gewährleisten.

Darüber hinaus bietet IPSec die Möglichkeit Daten in zwei unterschiedlichen Modi zu übertragen, per „Transport Mode“ oder mittels „Tunnel Mode“. Der Transportmodus bietet eine sichere Verbindung zwischen zwei Endpunkten, da er die Nutzlast der IP-Pakete kapselt, während der Tunnelmodus das gesamte IP-Paket kapselt, um einen virtuellen "sicheren Sprung" zwischen zwei Gateways bereitzustellen. Letzteres wird verwendet, um ein traditionelles Virtual Private Network (VPN) zu bilden, wobei der Tunnel im Allgemeinen einen sicheren Tunnel über ein nicht vertrauenswürdiges Internet erzeugt. Abbildung 7 zeigt die im Folgenden näher beschriebenen Mechanismen, Tunnel- und Transportmodus, im Vergleich zu herkömmlichem IP-Traffic ohne IPSec Nutzung, einerseits mit Authentication Header (AH) und andererseits mit Encapsulated Security Payload (ESP).

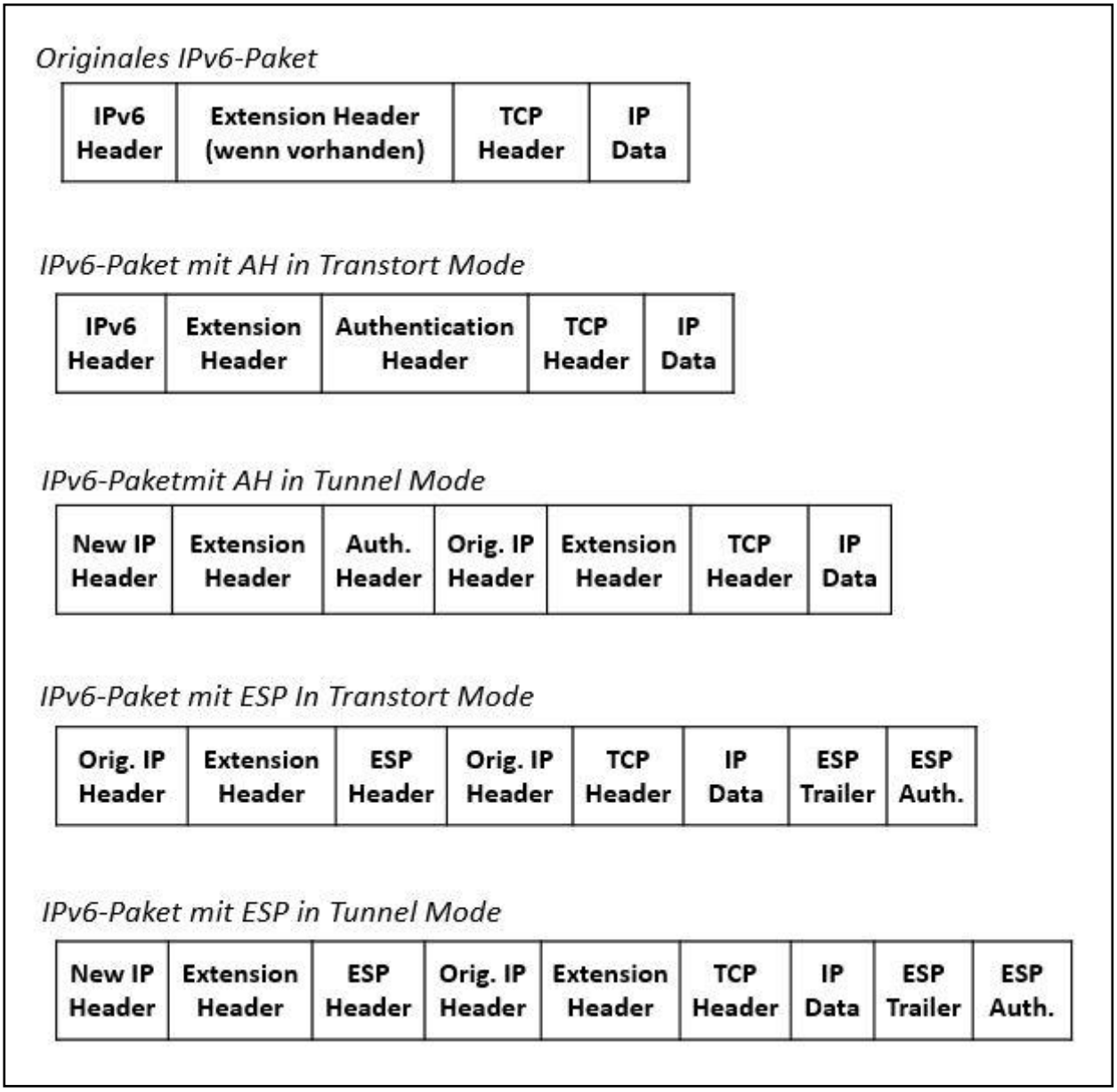


Abbildung 7: IPSec gesicherte IP-Pakete in verschiedenen Transportmodi.

Spezifikationen für IPSec definieren Protokolle für einen Authentication-Header (AH) und den Encapsulation Security Payload Header (ESP). Unter IPv6 kommen diese beiden Header als Erweiterungsheader zur Anwendung. Im folgenden Abschnitt soll gezielt auf die beiden Header und die beiden Übertragungsmodi eigegangen werden (Doraswamy und Harkins 2003).

3.2.2.1. Transport Mode

Der Transport Mode von IPSec wird verwendet, um eine „End-to-End“-Konversation zwischen zwei Hosts zu schützen. Dieser Schutz wird entweder durch Authentifizierung oder Verschlüsselung gewährleistet, ist jedoch kein Tunnel-Protokoll. Der Transportmodus wird von dem Host verwendet, der die IP-Pakete generiert. Im Transportmodus werden die Sicherheits-Header vor dem Header der Transportschicht (z. B. TCP, UDP) hinzugefügt, bevor der IP-Header dem Paket vorangestellt wird. Mit anderen Worten, ein dem Paket hinzugefügtes AH deckt das Hashing des TCP-Headers und einiger Felder des End-to-End-IP-Headers ab, und ein ESP-Header deckt die Verschlüsselung des TCP-Headers und der Daten ab, nicht aber das Ende IP-Header.

3.2.2.2. Tunnel Mode

Der Tunnelmodus ist der im Augenblick wohl meist verbreitetste Modus. Er wird verwendet, wenn der End-to-End-IP-Header bereits an das Paket angehängt und eines der Enden der sicheren Verbindung nur ein Gateway ist. In diesem Modus werden die AH- und ESP-Header verwendet, um das gesamte Paket einschließlich des End-to-End-Headers abzudecken, und ein neuer IP-Header wird an das Paket angehängt, das nur den Sprung zum anderen Ende der sicheren Verbindung abdeckt. Dies können natürlich mehrere IP-Hops sein.

3.2.2.3. Encapsulation Security Payload

Encapsulation Security Payload ist jener Teil von IPSec der Vertraulichkeit, Datenintegrität und Quellauthentizität eines IP-Paketes garantiert (RFC 4303 2005). Dies wird durch Einfügen eines neuen Headers, den Encapsulation Security Payload Header (ESP-Header), garantiert. Dieser Erweiterungsheader wird hinter dem IP-Header und möglichen Erweiterungs-Headern, aber vor den zu schützenden Daten eingeschoben. Darüber hinaus wird ein ESP-Trailer hinter den zu schützenden Daten angefügt, um das Paket abzuschließen.

3.2.2.4. Authentication Header

Wie der ESP-Header, so stellt auch der Authentication-Header (AH) Datenintegrität und Quellauthentizität der Daten bereit (RFC 4302 2005). Der Schutz der Datenintegrität erstreckt sich sowohl über das gesamte IP-Paket als auch über den IP-Header. Darüber hinaus bietet er einen Schutz gegen sogenannte „Replay Attacken“. Die Verwendung eines AH garantiert jedoch keine Vertraulichkeit während der Übertragung der Daten. Ein AH bietet Authentifizierung für möglichst viele IP-Header sowie für Protokolldaten der oberen Ebene. Aus diesem Grund ist der Authentication-Header auch einfacher aufgebaut als der ESP-Header. Einige IP-Header-Felder können sich jedoch im Transit ändern und der Wert dieser Felder ist, wenn das Paket bei dem Empfänger, der Empfängerin ankommt, möglicherweise von dem Absender, der Absenderin nicht vorhersehbar (z.B. „Hop-Limiter“). Die Werte solcher Felder können nicht durch AH geschützt werden. Im Vergleich zum ESP Protokoll kommt AH ohne einen IP-Paket abschließenden Trailer aus.

3.2.2.1. Security Association

Um eine ordnungsgemäße IPSec Datenübertragung zu ermöglichen, ist es notwendig eine Möglichkeit bereitzustellen, um, über dem normalen abhörbaren Medium hinaus, Informationen zum verwendeten Verschlüsselungsalgorithmus und dem genutzten Schlüssel auszutauschen. Solch eine Vereinbarung zwischen zwei kommunizierenden Einheiten wird als „Security Association“ (SA) bezeichnet. Das bedeutet, eine Security Association beschreibt wie KommunikationsteilnehmerInnen Sicherheitsmechanismen anwenden, um miteinander kommunizieren zu können (Doraswamy und Harkins 2003).

Eine solche IPSec-SA ist einfach direktional bindend. Das heißt, sie definiert Sicherheitsmechanismen nur für eine Richtung, somit entweder für eingehende Datenpakete, die empfangen werden, oder für ausgehende Pakete, die an andere Knoten übertragen werden. Solche Security Associations werden durch einen Security Parameter Index (SPI) identifiziert.

Die SA wird in einer Datenbank an jedem Endpunkt gehalten, indiziert durch die äußere Zieladresse, das sogenannte IPsec-Protokoll (AH oder ESP) und den Wert des Sicherheitsparameterindex. Die Auswahl von SA kann manuell erfolgen (Pre-Shared Keys), ist jedoch vorzugsweise mit Internet Key Exchange (IKE, IKEv2) automatisiert. IKE verwendet Diffie-Hellman-Techniken zum Erstellen eines gemeinsamen geheimen Verschlüsselungsschlüssels, der zum Aushandeln von SA-Daten verwendet wird. Für den Schlüsselaustausch hängt IKE von einer Public Key Infrastructure (PKI) ab, die noch nicht weit verbreitet ist. Das Framework und die Syntax für den Schlüsselaustausch ist Internet Security Association und Key Management Protocol (ISAKMP) (RFC 7296 2014).

3.2.2.2. Internet Key Exchange

IKE ist eine Komponente von IPsec, die zur gegenseitigen Authentifizierung und zum Erstellen und Verwalten von Sicherheitsvereinbarungen (SAs) verwendet wird (RFC 7296 2014). IKE führt die gegenseitige Authentifizierung zwischen zwei Parteien durch und richtet eine IKE-Security Association ein, die gemeinsam genutzte geheime Informationen enthält, die wiederum zum effizienten Einrichten von SAs zur Nutzung des Encapsulating Security Payload-Headers und/oder der Nutzung eines Authentication Headers notwendig sind. Der Ablauf einer solchen IKE-Kommunikation wird im folgenden Absatz beschrieben (Abbildung 8).

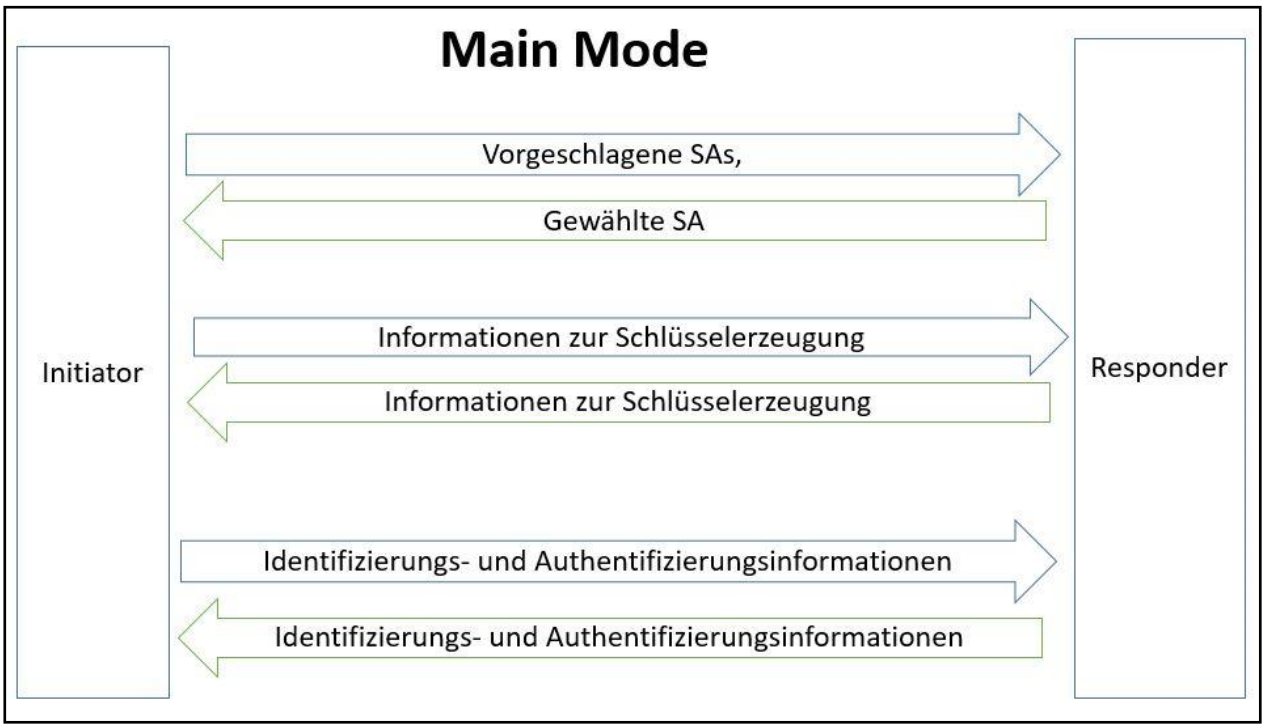


Abbildung 8: Ablauf einer IKE-Session im Main Mode

Eine IKE-Sitzung beginnt damit, dass der Initiator dem Antwortenden einen Vorschlag oder mehrere Vorschläge sendet. Die Vorschläge definieren, welche Verschlüsselungs- und Authentifizierungsprotokolle akzeptabel sind, wie lange Schlüssel aktiv bleiben sollen, und ob beispielsweise eine „Perfect Forward Secrecy“ (PFS) erzwungen werden soll. Diese, zu Deutsch „perfekte Vorwärts-Geheimhaltung“ liegt vor, wenn die verwendeten Schlüssel einer Session nach der Beendigung der Sitzung nicht mehr aus den geheimen Langzeitschlüsseln rekonstruiert werden können. Mehrere Angebote zu unterschiedlichen Verschlüsselungsalgorithmen können

in einem Angebot gesendet werden. Der genaue Ablauf einer IKE-Sitzung verläuft per Definition wie folgt:

Der erste Austausch zwischen den Knoten legt die grundlegende Sicherheitsrichtlinie fest und der Initiator schlägt die Verschlüsselungs- und Authentifizierungsalgorithmen vor, die er verwenden möchte. Der antwortende Knoten wählt den passenden Vorschlag und sendet ihn an den Initiator. Der nächste Austausch beinhaltet öffentliche Schlüssel und andere Daten im Zuge des Diffie-Hellman-Verfahrens. Alle weiteren Verhandlungen werden innerhalb der IKE SA verschlüsselt. Die dritte Exchange authentifiziert die Sitzung. Sobald die IKE-SA eingerichtet ist, beginnt die IPSec-Verhandlung.

Dieser Datenaustausch stellt den sogenannten „Main Mode“ des IKE-Protokolls dar. Der Informationsaustausch ist nach sechs Messages beendet. Hierbei sind Nachricht fünf und sechs bereits verschlüsselt übertragen worden.

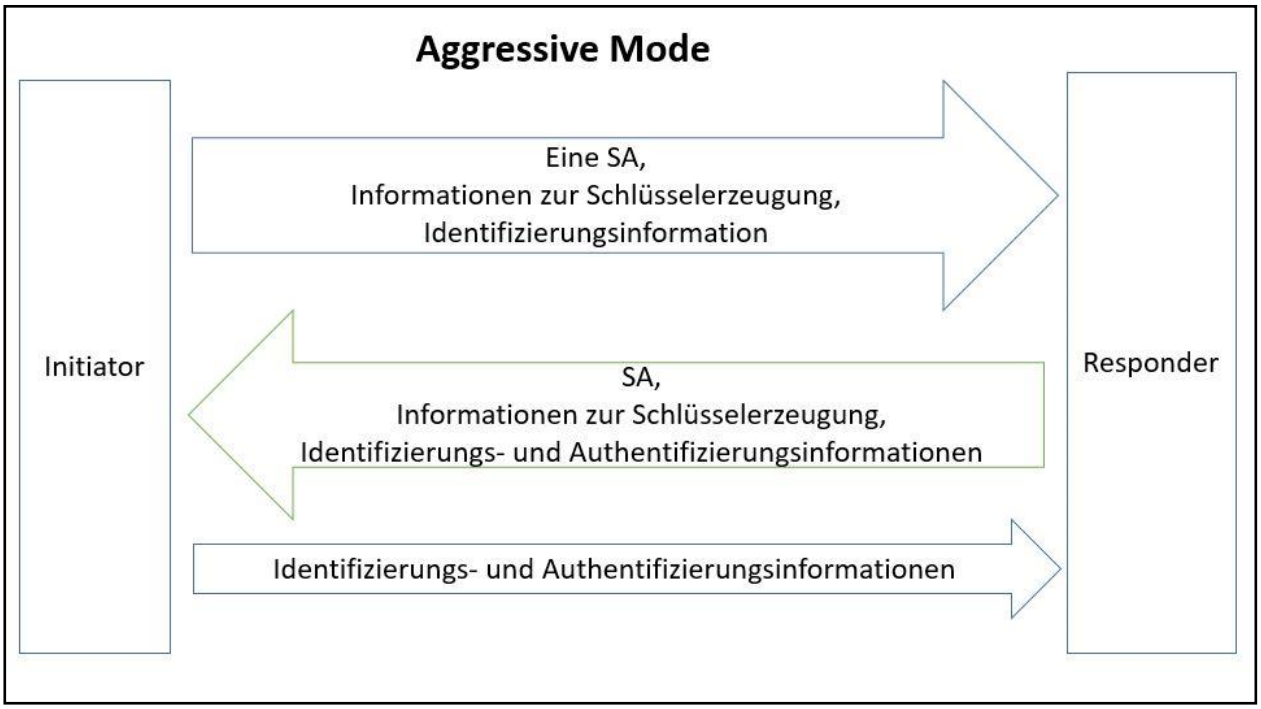


Abbildung 9: Ablauf einer IKE-Session im Aggressive Mode.

Ein effizienterer Modus, der sogenannte „Aggressive Mode“ reduziert die IKE-SA-Verhandlung auf lediglich drei Pakete, wobei alle für die SA erforderlichen Daten vom Initiator übergeben werden (RFC 7296 2014). Der Responder sendet den Vorschlag, das Schlüsselmaterial und die ID und authentifiziert die Sitzung im nächsten Paket. Der Initiator antwortet durch

Authentifizierung der Sitzung. Die Verhandlung ist schneller und die Initiator- und Responder-ID wird im Klartext übergeben (Abbildung 9).

Hierbei beinhaltet die erste Message des „Aggressive Mode“ alle Informationen der Messages 1 und 3 des „Main Mode“ und zusätzliche Identifizierungsinformationen aus Nachricht 5 des Main Modes.

Die Informationen der Nachrichten 2, 4 und 6 des „Main Mode“ werden im „Aggressive Mode“ in Nachricht 2 übertragen.

Nachricht 3 beinhaltet schließlich nur noch Authentifizierungsinformationen wie sie in Nachricht 5 des „Main Mode“ übertragen werden.

3.2.3 Potentielle Schwächen von IPv6

Durch die Tatsache, dass die Ursprünge von IPv6 bis in die 90er Jahre des letzten Jahrhunderts zurückreichen, wurden von SicherheitsforscherInnen bereits zahlreiche Schwächen von IPv6 aufgezeigt und potentielle, darauf abzielende Angriffsszenarien vorgestellt. Im folgenden Abschnitt soll auf die bekanntesten Risiken und möglichen Angriffsvektoren, die bei Verwendung von IPv6 bestehen, eingegangen werden.

Obwohl in IPv6-Netzwerken viele Sicherheitsrisiken ebenso präsent sein werden wie sie es bereits in IPv4-Netzwerken sind, bietet IPv6 den AnwenderInnen auch neue Funktionen, welche sich auf die System- und Netzwerksicherheit auswirken, und darüber hinaus potentiellen Einfluss auf Richtlinien und Prozeduren haben werden. Dies wird beispielsweise im Besonderen bei der Implementierung und Umsetzung umfangreicher IPv6-Netzwerke der Fall sein (IPv6Now 2017). In der Einführungsphase von IPv6 wird das neue Protokoll häufig parallel zu IPv4 betrieben. Da IPv4 und IPv6 normalerweise völlig unabhängig voneinander über dieselbe Layer2-Infrastruktur betrieben werden, ist es nötig, zahlreiche zusätzliche und separate IPv6-Sicherheitsmechanismen zu implementieren. Dazu müssen häufig viele Bereiche (zB. Firewall) überholt und neu implementiert werden. Auf solche Sonderfälle wird im folgenden Abschnitt ebenso eingegangen, wie auf Fälle, die bei reiner Verwendung von IPv6 auftreten.

Folgend werden Sicherheitsauswirkungen erläutert, die bei der Einführung von IPv6 in Netzwerken berücksichtigt werden müssen, besonders wenn parallel zu IPv6, weiterhin IPv4 betrieben wird.

- ICMP und Multicast

  Die übliche IPv4-Praxis, ICMP-Pakete (Internet Communication Messaging Protocol) als vermeintliche Sicherheitsmaßnahme zu blockieren, ist in IPv6-Nettzwerken nicht anzuwenden, da IPv6 für Fehlermeldungen, Pfad-MTU-Erkennung, Multicast-Gruppenverwaltung und Neighbor Discovery von ICMPv6 abhängt. Beispielsweise werden in IPv6 zu große Pakete von Routern nicht wie in IPv4 fragmentiert, was den Durchsatz erheblich verbessert. Wenn ein Paket zu groß ist, um weitergeleitet zu werden, verwirft der Router das Paket und sendet dem Host eine ICMPv6-Packet-too Big-Nachricht, die die MTU des nächsten „Hop“ enthält. Der Host verwendet jetzt die niedrigere MTU und überträgt das Paket erfolgreich erneut (Hogg und Vyncke 2009).

  IPv6 stützt sich auch auf die Multicastverfügbarkeit, die sich auf Firewalls, Angriffserkennungs- und Zugriffssteuerungsregeln auswirken wird (RFC 4884 2007) und einen Wandel des Sicherheitsdenkens vieler administrierender Personen erfordert, da ein Blockieren von ICMP in IPV6 einige schwer zu diagnostizierende Probleme verursachen kann.

- Automatisches Tunneling

  Tunneling bedeutet, dass Pakete eines Protokolls von Paketen eines zweiten Protokolls für den Transport über ein Netzwerk des zweiten Typs eingekapselt werden (Abbildung 10). Tunnel sind eine wesentliche IPv6-Übergangstechnik, um die Zeitspanne bis zur vollständigen Nutzung von IPv6 aller teilnehmenden Geräte des Internets zu überbrücken. Einige Betriebssysteme erstellen jedoch automatisch ein IPv6-Netzwerk, wenn ein Client mit einem Server verbunden ist, zum Beispiel verschiedene Windows-Versionen. Potentiell unerwünschte neue Pfade zu Hosts können automatisiert eingerichtet werden, und Firewalls sind möglicherweise auf dieses neue Szenario nicht vorbereitet.

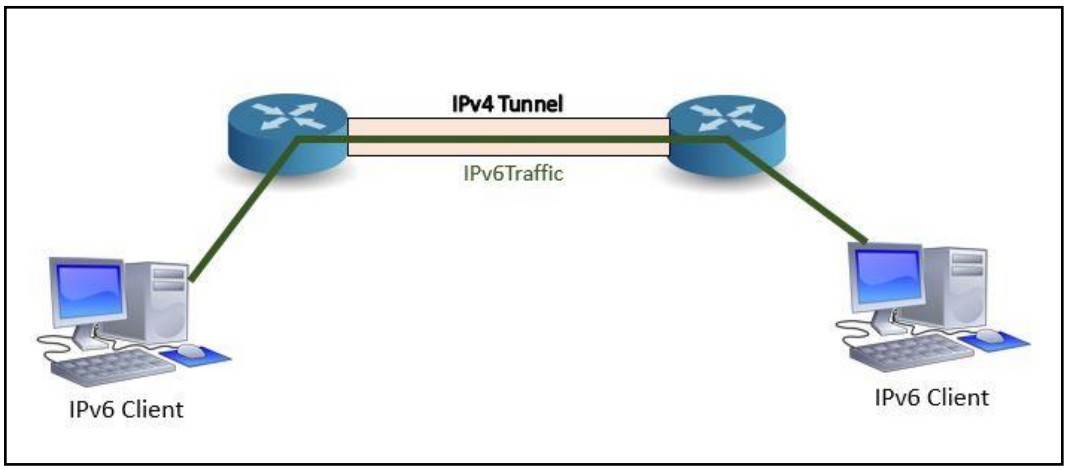


Abbildung 10: Tunneling von IPv6 Traffic über IPv4 Infrastruktur

- Dual Stacking

  Dual Stacking, auch „Dual IP Layer“ genannt, bedeutet, dass Geräte sowohl IPv4- als auch IPv6-Protokollfunktionen haben (RFC 4213 2005). Es ist eine Technik, die eine vollständige Unterstützung der Internetprotokolle IPv4 und IPv6 in Hosts und Routern ermöglicht und normalerweise als eine wesentliche Übergangsmethode für die schrittweise Bereitstellung von IPv6 angesehen wird. Abbildung 11 zeigt die schematische Darstellung des Dual Stack Protokolls (rechts) im Vergleich zu einem IPv4 Stack (links).

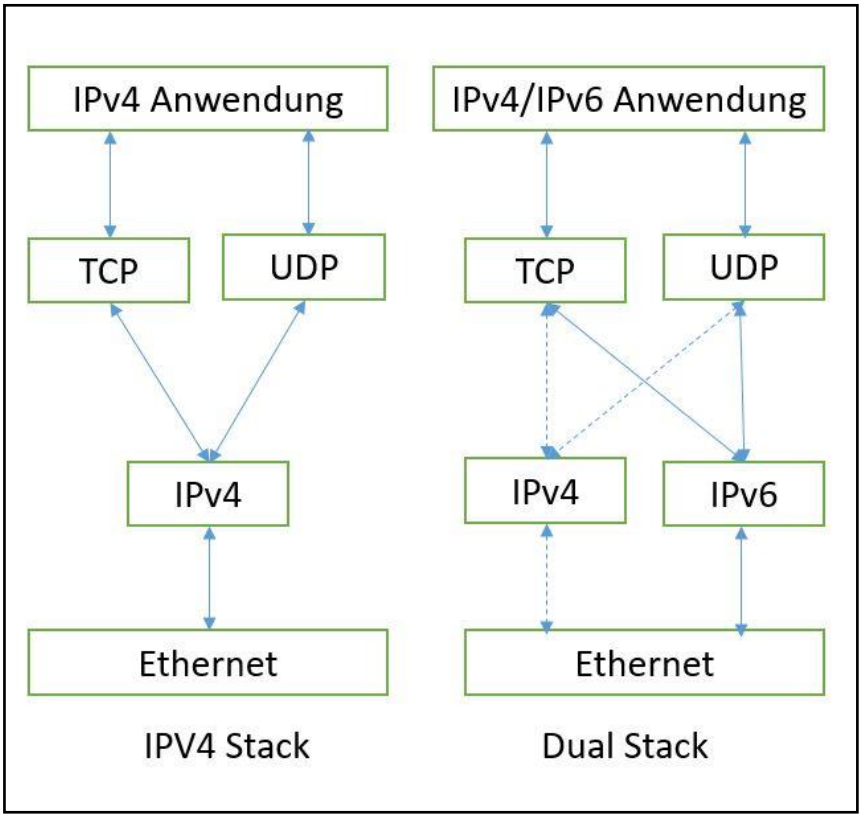


Abbildung 11: Schematische Darstellung eines Dual-Stacks im Vergleich zu einem IPv4-Stack.

Die Nutzung dieser Technik bedeutet aber auch, dass zwei Protokolle im Spiel sind. Die Sicherheit muss für beide Protokolle aufrechterhalten werden (Ladid et al. 2005). Dies ist in Bezug auf Zeit und Aufwand teuer, sodass einige große Organisationen IPv6 vollständig in ihren internen Netzwerken anwenden und Konvertierungstechniken (6to4) an den Netzwerkgrenzen zum Einsatz bringen, um diese Konnektivität zu gewährleisten.

Diese Beispiele bieten einen kurzen Einblick in die Vielzahl an Themengebieten, die von den administrierenden Personen bei der parallelen Nutzung von IPv4 und IPv6 zu berücksichtigen sind. Obwohl die Einführung von IPv6 einen vermeintlichen Sicherheitszuwachs mit sich bringt, ist es von größter Bedeutung, darauf zu achten, dass durch die parallele Nutzung potentiellen AngreiferInnen nicht Türen geöffnet werden, die die Sicherheit der Infrastruktur gefährden.

Der folgende Abschnitt beschäftigt sich gezielt mit Sicherheitsrisiken, die durch die ausschließliche Verwendung von IPv6 aufkommen. Die weitere Vorgehensweise in der vorliegenden Arbeit wird sich hauptsächlich auf die Sicherung von IoT-Geräten fokussieren,

welche in einem reinen IPv6-Netzwerk betrieben werden. Es wird daher im späteren Verlauf dieser Arbeit kein Augenmerk auf allfällige Absicherungen gegenüber gängigen IPv4-Attacken gelegt und auch kein Vergleich gegenüber der Absicherung von Geräten in IPv4-Netzwerken und Geräten in IPv6-Netzwerken gemacht werden.

- Autokonfiguration

  Autokonfiguration in IPv6 ist ein effizienter und wirtschaftlicher Prozess, weist jedoch potenzielle Schwachstellen auf. Stateless Address Autoconfiguration (SLAAC) ist der Prozess, bei dem ein Host seine eigene Adresse basierend auf seiner Hardware-Adresse (MAC) konfiguriert (siehe Kapitel 3.2.1). Die Offenlegung von MAC-Adressen kann jedoch eine Hostidentifizierung über Schnittstellen-ID, NIC-Anbieter oder Host-Anbieter ermöglichen, wodurch AngreiferInnen wertvolle Informationen über das Ziel in die Hände gespielt werden (Ladid et al. 2005). Durch zufällige, temporäre oder kryptographische Mittel erzeugte Adressen können dieses Problem lösen. Eine Möglichkeit diese potentielle Sicherheitslücke zu schließen bietet das Dynamic Host Configuration Protocol for IPv6 (DHCPv6) (RFC 3315 2003). Mit DHCPv6 kann ein Server Adressen an Hosts liefern, wie dies bereits durch das Dynamic Host Configuration Protocol (DHCP) in IPv4 der Fall war. Im Vergleich zu DHCP in IPv4, das externe Unterstützung zur korrekten Vergabe von IPv4-Adressen benötigt (Festlegen des Standardgateways und Definition der Subnetmaske), genügt in IPv6 das Vorhandensein nur eines funktionierenden Routers, damit der verbundene Host sofort erreichbar ist.

- Hosts mit mehreren Adressen

  In IPv4 sind mehrere Adressen an einem Interface zwar möglich, aber selten (RFC 6419 2011). In IPv6 sind sie jedoch sehr verbreitet und entstehen aus SLAAC, temporärem DHCPv6, verbindungslokalen Adressen, mehreren Präfixen, überlappender Lebensdauer sowie IPv4-Adressen. Abbildung 12 zeigt beispielhaft, welche IP Adressen einem Interface zugeordnet werden können und welche Bedeutung die einzelnen Adressen haben.

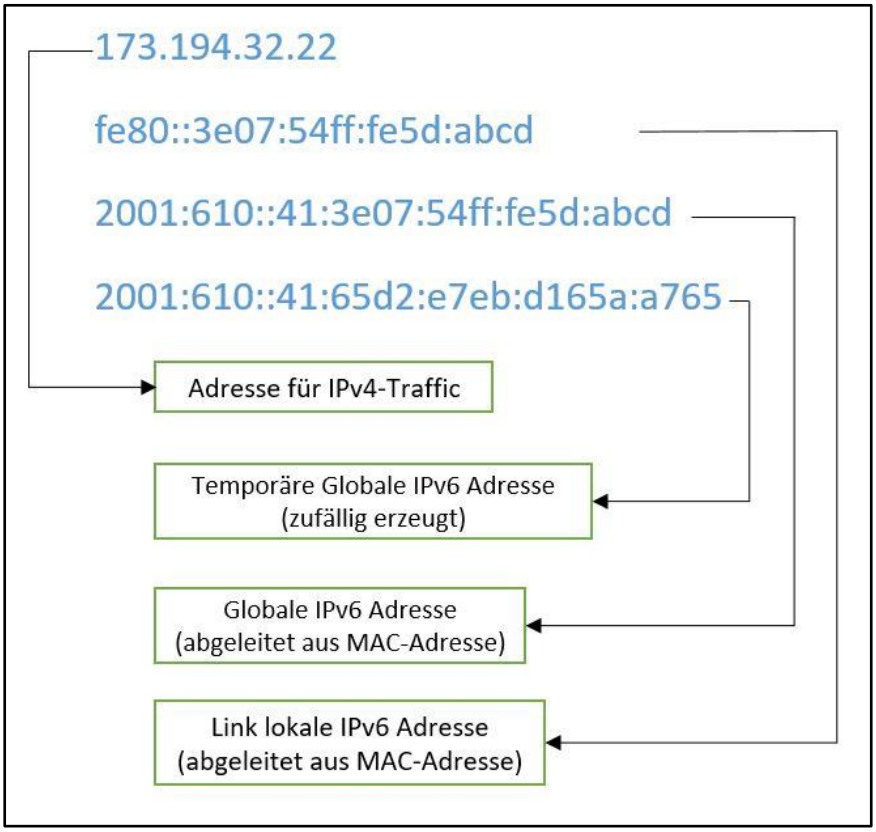


Abbildung 12: Beispiel von Multiplen IP-Adressen an einem Interface.

Administrierende Personen müssen sich aller möglichen Schnittstellenadressen und der Kapazität von Netzwerkgeräten bewusst sein, um ihre eigenen Adressen zu erstellen. Dies kann beispielsweise dazu führen, dass sich verschiedene Einträge einer Logdatei mit unterschiedlichen IP-Adressen auf einen einzelnen Host beziehen. Dies kann umfassende Auswirkungen auf Fehleranalysen und Problemfindungen haben. Auch AngreiferInnen wird dadurch die Möglichkeit gegeben, ihre Ziele über verschiedene Wege anzugreifen und ihr Handeln zu fragmentieren und hinter verschieden Adressen zu verschleiern und zu verstecken.

- Scans und IPv6

  Mit 18 Milliarden Adressen in einem /64-Subnetz ist das sequentielle Scannen sinnlos (RFC 7707 2016). Unter der Annahme von einer Million Durchläufen pro Sekunde, würde es immer noch circa 500.000 Jahre dauern, um ein einzelnes /64 Segment vollständig zu scannen. Es ist jedoch möglich das Scannen zu beschleunigen, indem man den zu durchsuchenden Bereich auf Basis von zusätzlichen Informationen eingrenzt. Diese Zusatzinformationen lassen sich beispielsweise durch den Einsatz gängiger Techniken zur Informationsgewinnung beschaffen.

Beispiele hierfür können sein:

- Neighbor Discovery Protocol
- Routing-Tabellen,
- „Whois“ oder „Reverse-DNS“

All diese Techniken können genutzt werden, um gezielt verwundbare Hosts in einem Netzwerk zu lokalisieren.

- NDP Spoofing

  Eine sehr effektive Methode um Informationen aus einem IPv4-Netzwerk zu generieren, bietet das sogenannte Address Resolution Protocol (ARP) Spoofing. Obwohl IPv6 ARP nicht mehr nutzt, ist dieser Angriffsvektor nur scheinbar eliminiert. Zwar findet ARP in IPv6-Netzwerken keinerlei Verwendung mehr, jedoch wird die Funktion durch das sogenannte Neighbor Discovery Protocol (NDP) ersetzt, welches ähnliche Angriffsszenarien zulässt wie ARP.

  IPv6-Knoten verwenden das Neighbor Discovery Protocol, um andere Knoten auf der Verbindung zu ermitteln und ihre Verbindungsschichtadressen zu eruieren. Darüber hinaus wird NDP genutzt, um Router zu finden und um Erreichbarkeitsinformationen über die Pfade zu aktiven Nachbarn zu erhalten (RFC 4861 2007). Es stellt sich heraus, dass dem Basis-Neighbor Discovery Protocol ein Mechanismus fehlt, um autorisierte Knoten zu identifizieren. Wenn NDP nicht abgesichert eingesetzt wird, ist es anfällig für verschiedene Angriffe. Dazu zählen beispielsweise die Umleitung von Traffic, das Stehlen von IP-Adressen, die Gefahr einer Denial-of-Service Attacke und das sogenannte Parameter-Spoofing (auch Parameter-Sniffing genannt). Ein Vorschlag (RFC 3682 2004) zur Verwendung einer "Sprunganzahl (Hop Count) von 255" hat nur einen ziemlich begrenzten Nutzen zur Erreichung einer größeren Sicherheit. Die Verwendung von IPSec-Authentifizierungsheader (AH) oder Encapsulating Security Payload (ESP) funktioniert nur mit manuellem Keying und vordefinierten Sicherheitszuordnungen.

  Knoten auf derselben Verbindung verwenden NDP, um die Anwesenheits- und Verbindungsschichtadressen der jeweils anderen Seite zu ermitteln, Router zu finden und Erreichbarkeitsinformationen über die Pfade zu aktiven Nachbarn zu erhalten. NDP wird sowohl von Hosts als auch von Routern verwendet. Zu seinen Funktionen gehören Neighbor Discovery (ND), Router Discovery (RD), automatische Adresskonfiguration, Adressauflösung, Neighbor Unreachability Detection (NUD) (ein Mechanismus zur Verfolgung der Erreichbarkeit von Nachbarn), Duplicate Address Detection (DAD) und die Umleitung von Datenpaketen.

Die ursprünglichen NDP-Spezifikationen forderten die Verwendung von IPsec zum Schutz von NDP-Nachrichten. Die in der Literaturrecherche berücksichtigten RFCs geben jedoch keine detaillierten Anweisungen für die Verwendung von IPsec. In dieser speziellen Anwendung kann IPSec nur mit einer manuellen Konfiguration von Sicherheitszuordnungen verwendet werden, da Probleme bei der Verwendung des Internet Key Exchange Protokolls auftreten. Darüber hinaus kann die Anzahl der manuell konfigurierten Sicherheitsverknüpfungen, die zum Schutz von NDP benötigt werden, sehr groß sein, was diesen Ansatz für die meisten Zwecke unpraktisch macht (Loshin 2004).

Das Secure Neighbor Discovery-Protokoll (SEND) kann einigen der Bedrohungen gegen das ND-Protokoll entgegenwirken, wenn IPSec nicht verwendet wird. SEND verwendet kryptografisch generierte Adressen, um zu verifizieren, dass ein absendendes Gerät tatsächlich im Besitz einer beanspruchten Adresse ist. Cryptographic Generated Addresses (CGA) sind IPv6-Adressen, bei denen ein Teil der Adresse durch Anwenden einer kryptografischen Einweg-Hash-Funktion erzeugt wird, die auf dem öffentlichen Schlüssel und den Hilfsparametern eines Knotens basiert. Der Hash-Wert kann dann verwendet werden, um die Bindung zwischen dem öffentlichen Schlüssel und der Adresse eines Knotens zu überprüfen. Abbildung 13 zeigt vereinfacht die Erzeugung einer CGA nach RFC 3972.

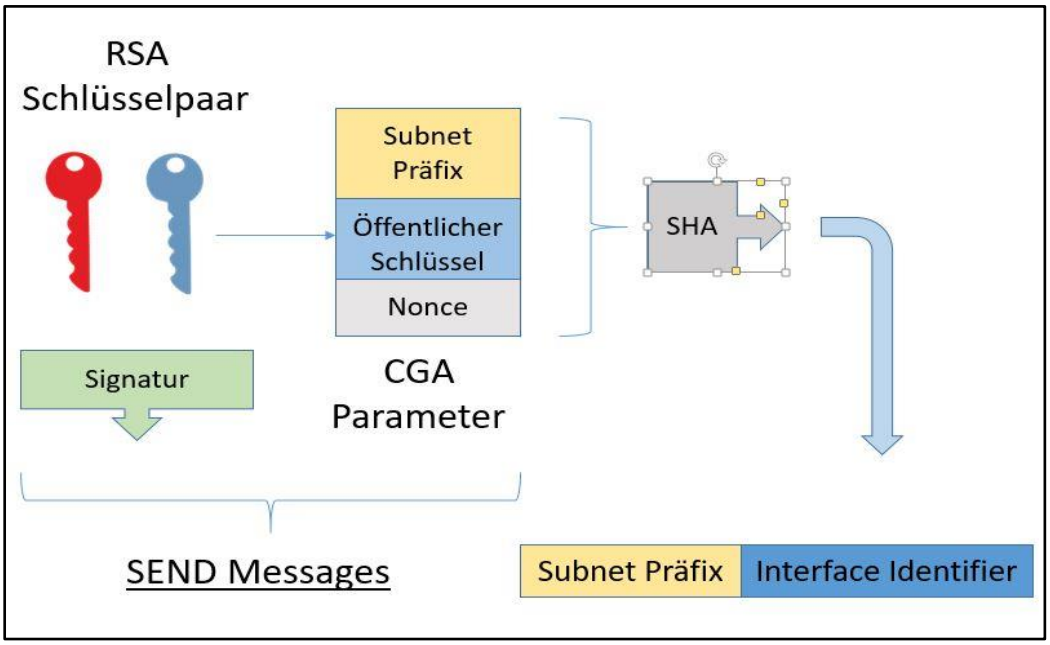


Abbildung 13: Kryptografisch generierte Adresse (CGA) (RFC 3972 2005).

Standardmäßig sollte ein SEND-aktivierter Knoten nur CGAs für seine eigenen Adressen verwenden. Der grundlegende Zweck von CGAs besteht darin, das Stehlen oder Spoofing von bestehenden IPv6-Adressen zu verhindern (Caicedo, Joshi Tuladhar, 2009).

- IPv6 Routing Header

  Gemäß der IPv6-Protokollspezifikation müssen alle IPv6-Knoten in der Lage sein, Routing-Header zu verarbeiten. Unglücklicherweise können Routing-Header verwendet werden, um Zugriffskontrollen basierend auf Zieladressen zu vermeiden. Ein solches Verhalten kann zu Sicherheitsproblemen führen (Loshin 2004). Es besteht die Möglichkeit, dass eindringende Geräte ein Paket an eine öffentlich zugängliche Adresse mit einem Routing-Header senden, der eine "verbotene" Adresse (Adresse im Opfer-Netzwerk) enthält. In diesem Fall leitet der öffentlich zugängliche Host das Paket an eine Zieladresse weiter, die im Routing-Header angegeben ist ("verbotene" Adresse), obwohl diese Zieladresse gefiltert ist. Durch Spoofing von Paketquelladressen können eindringende Geräte leicht einen Denial-of-Service-Angriff initiieren, indem sie einen öffentlich zugänglichen Host zum Umleiten von Angriffspaketen verwenden.

- Fragmentierung von Paketen

  Gemäß der IPv6-Protokollspezifikation ist eine Paketfragmentierung durch Zwischenknoten nicht zulässig. Da in IPv6-Netzwerken die Verwendung der Pfad-MTU-Ermittlungsmethode (basierend auf ICMPv6-Nachrichten) eine Verpflichtung ist, ist eine Paketfragmentierung nur am Quellknoten möglich. Die empfohlene minimale MTU-Größe für IPv6-Netzwerke beträgt 1280 Byte. Aus Sicherheitsgründen wird dringend empfohlen, alle Fragmente mit weniger MTU-Größe als Byte Oktetten zu verwerfen, es sei denn, das Paket ist das letzte im Fluss. Durch die Fragmentierung kann ein eindringendes Gerät erreichen, dass Portnummern nicht im ersten Fragment gefunden werden und auf diese Weise Sicherheitsüberwachungsgeräte, die Fragmente nicht wieder zusammensetzen, umgehen, da diese Geräte erwarten, Transportschichtprotokolldaten im ersten Fragment zu finden. Durch das Senden einer großen Anzahl kleiner Fragmente können AngreiferInnen eine Überlastung der Rekonstruktionspuffer auf dem Zielsystem verursachen, was möglicherweise ein System zum Absturz bringt (eine Art von Denial-of-Service-Angriff). Um solche Probleme zu vermeiden, ist es eine empfohlene Sicherheitspraxis, die Gesamtzahl der Fragmente und ihre zulässige Ankunftsrate zu begrenzen.

- Denial of Service Attacken (DoS)

  Sogenannte „Denial of Service“ Attacken sind eine sehr verbreitete Methode um Webserver zu blockieren. Dabei werden von AngreiferInnen eine große Zahl an Anfragen an den Zielserver gesendet, sodass dessen Ressourcen voll und ganz ausgelastet sind, und er nicht mehr in der Lage ist, Anfragen von anderen NutzerInnen abzuarbeiten (Soltanian und Amiri 2015).

  Eine Abwandlung dieser Angriffsmethode stellt die sogenannte „Distributed Denial of Service“ Attacke dar (DDoS). Man spricht von dieser Angriffsmethode, wenn DoS-Angriffe von großen Netzwerken oder Systemen aus unternommen werden. Hierbei wird der Zielserver durch eine groß angelegte, über viele Systeme verteilte DoS-Attacke an seine Kapazitätsgrenzen gebracht. Um einen DDoS-Angriff durchzuführen, verwenden AngreiferInnen verschiedene Ressourcen wie Netzwerkknoten und Internetdienste, die rund um den Globus verteilt sind und als sogenannte „Botnets“ gelten. Später werden diese Botnets verwendet, um den DDoS-Angriff gegen das Opfer zu starten, wodurch der Server für andere NutzerInnen nicht mehr erreichbar ist. Durch die zukünftig stark steigende Anzahl der im Internet verfügbaren Geräte und deren Anbindung an das Internet über eindeutige IPv6-Adressen wird diese Art der Bedrohung auch in IPv6-Netzwerken der Zukunft einen gravierenden Angriffsvektor darstellen.

  DoS- Angriffe im IPv6-Netzwerk können grob in zwei Hauptkategorien unterteilt werden, die jeweils auf der angegriffenen Ebene basieren. Einerseits der Anwendungsebene, andererseits der Netzwerkebene. Weitere DoS-Angriffe auf Netzwerkebene können in Gateway- (Router) bzw. lokale Verbindungsebenen unterteilt werden. Abbildung 14 stellt die Taxonomie von DoS-Angriffen in IPv6-Netzwerken dar (Rehman und Manickman 2016).

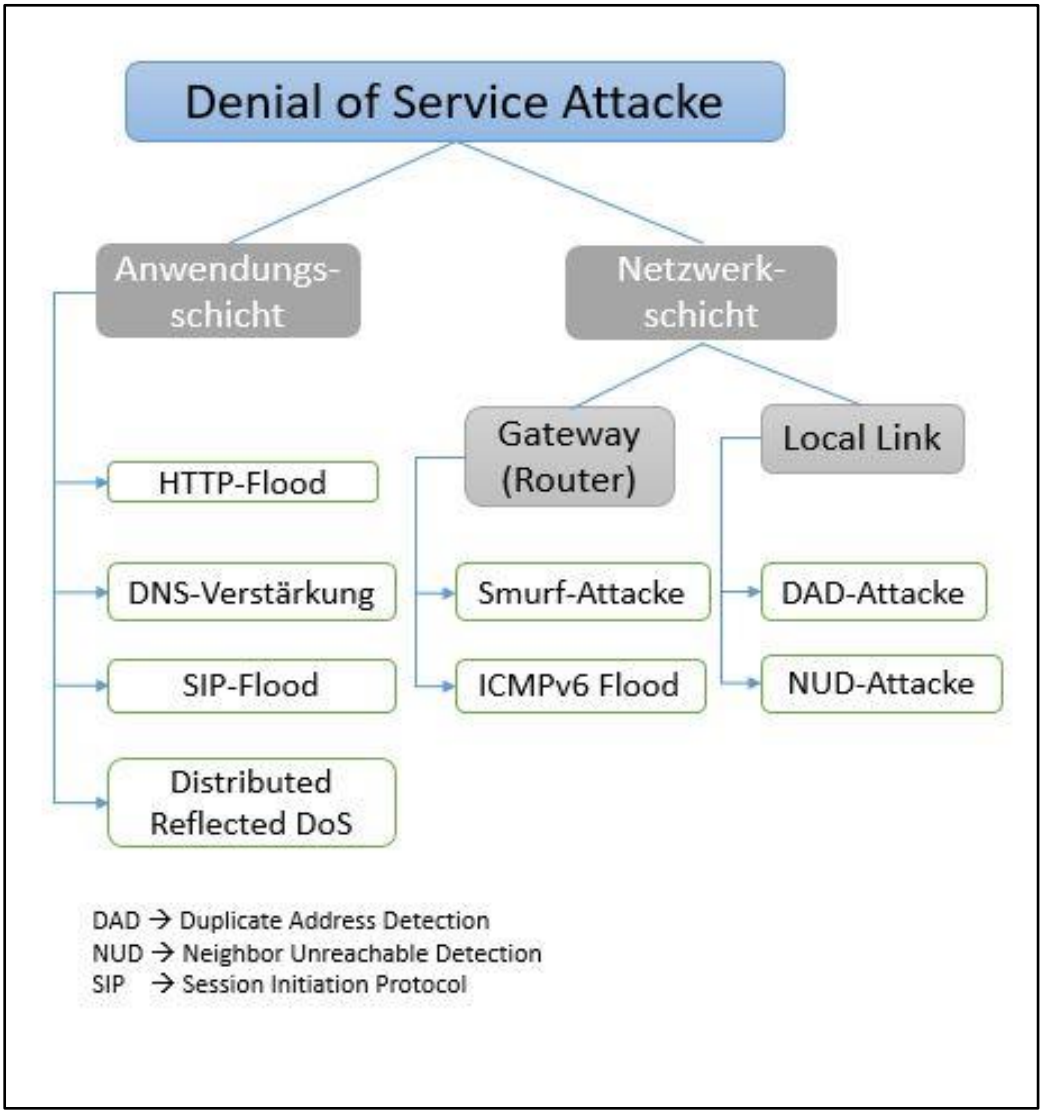


Abbildung 14: Taxonomie von DoS-Attacken in IPv6-Netzwerken (Rehman und Manickman 2016).

Während der automatischen Adresskonfiguration im lokalen IPv6-Verbindungsnetzwerk werden die Nachrichtentypen des Internet Control Message Protocol (ICMPv6) von den Hosts verwendet, um mit benachbarten Hosts innerhalb einer lokalen Verbindung zu kommunizieren. Studien haben jedoch gezeigt, dass ICMPv6-Nachrichten anfällig für Denial-of-Service-Attacken (DoS) sind, insbesondere während des DAD-Prozesses (Duplicate Address Detection), während Hosts versuchen ihre eigene generierte Schnittstellen-ID (IID) zu konfigurieren.

Aus diesem Grund können AngreiferInnen einen Vorteil daraus ziehen und diese ICMPv6-Nachrichten erstellen. Später können AngreiferInnen diese modifizierten Nachrichten ausnutzen, um DoS-Angriffe auf verschiedene Arten zu generieren. Dies ist möglich durch Spoofing der Nachrichten, Man-in-the-Middle-Form oder einfach durch das Senden einer

übermäßigen Anzahl gefälschter ICMPv6-Pakete an den Zielhost auf dem lokalen Link. Daher können AngreiferInnen die IPv6-Hosts stören, um ihre Schnittstellen-ID (IID) zu erhalten.

Obwohl das neue Internet Protokoll bereits seit vielen Jahren existiert (die Veröffentlichung erster Standards erfolgte 1999), ist es operativ doch noch sehr jung und wenig verbreitet. Dadurch ist es denkbar, dass IPv6 weitere Probleme und Schwierigkeiten mit sich bringt, die es bisher noch nicht gibt, oder die man sich bis heute nicht vorstellen kann. Die angeführten Beispiele zeigen deutlich, dass durch die Einführung von IPv6 in Netzwerken, insbesondere bei der parallelen Nutzung beider IP-Protokolle, trotz der weit verbreiteten Meinung „…IPv6 bringt einen Zuwachs an Sicherheit…“ ein besonderes Augenmerk auf die Absicherung der Netzwerke gelegt werden muss.

4 IOT-GERÄTE UND IPV6

Das folgende Kapitel geht im Besonderen auf den Einsatz von IPv6 in IoT-Geräten ein. Es wird besonderes Augenmerk darauf gelegt, ob und inwieweit der Einsatz von IPv6 in IoT-Geräten die Funktionalität von IPv6 beeinträchtig und ob diese Beeinträchtigungen sicherheitsrelevante Auswirkungen nach sich ziehen können. Auf Basis dieser Ergebnisse werden darauffolgend mögliche Angriffsszenarien auf IoT-Geräte in IPv6-Netzwerken abgeleitet und ausgearbeitet. Unter Anwendung des bis zu diesem Zeitpunkt erarbeiteten Wissens über die Funktion und Spezifikationen von IPv6, sollen darüber hinaus bereits erste potentielle Schutzmechanismen erarbeitet werden, um IoT-Geräte gegen dargestellte Angriffe zu schützen.

4.1 Einsatz von IPv6 in IoT-Geräten

Einerseits führt die fortschreitende Verbreitung von IPv6 dazu, dass immer mehr IoT-Geräte an sich, als auch Betriebssysteme für IoT-Geräte das neue Internet Protokoll unterstützen. Andererseits ist der stetig wachsende Einsatz von IoT-Geräten und deren Verbreitung ein starker Treiber für einen flächendeckenden Einsatz von IPv6.

IPv6 gilt als die am besten geeignete Technologie für das Internet der Dinge, da es Skalierbarkeit, Flexibilität, getestete, erweiterte, allgegenwärtige, offene und End-to-End-Konnektivität bietet. (Jara et al. 2014).

Viele Geräte unserer Umgebung sind bereits über das Internetprotokoll miteinander verbunden. Darunter fallen beispielsweise Drucker, Sensoren, Beleuchtung, Gesundheitssysteme, intelligente Messgeräte, Videokameras, Fernseher und Heizungssteuerungssysteme und Mobiltelefone. Die steigende Verbreitung dieser Geräte hat dazu geführt, dass diese bei der Entwicklung neuer Standards und Technologien besonders berücksichtigt werden. Besonders das Aufkommen von IPv6-bezogenen Standards, die speziell für das Internet der Dinge entwickelt wurden, wie beispielsweise IPv6 over Low power Wireless Personal Area Network (6LoWPAN), auf das im Folgenden noch näher eingegangen wird, und Constrained Application Protocol (CoAP), haben es ermöglicht, dass auch stark eingeschränkte, ressourcenschwache Geräte IP-kompatibel werden (Ziegler et al. 2013).

Im Zuge der für diese Arbeit durchgeführten Recherche zu IPv6-fähigen IoT-Geräten im Internet wurde sehr schnell klar, dass beinahe alle am Markt befindlichen IoT-Geräte, die in irgendeiner Art und Weise über eine Schnittstelle zur Netzwerkkommunikation verfügen, IPv6 unterstützen. Auch existieren für beinahe alle Geräte ohne vorinstalliertes Betriebssystem (z.B. Einplatinencomputer) kompatible Betriebssysteme, die in der Lage sind, das neue Internetprotokoll zu verarbeiten.

4.2 Limitationen durch IoT-Geräte

Ein entscheidender Limitationsfaktor beim Einsatz von IoT-Geräten stellt neben den begrenzten Hardwareressourcen auch eine eingeschränkte Leistungsaufnahme dar. Dies liegt einerseits darin begründet, dass viele IoT-Geräte als mobile Endgeräte konzipiert sind, weshalb ihr Energiebedarf über Akkumulatoren gedeckt werden muss, andererseits setzen viele Hersteller zur Versorgung Ihrer IoT-Geräte auf gängige Standards wie beispielsweise USB. Dies definiert das Limit des Energieverbrauches der IoT-Geräte, da USB-Anschlüsse nur einen begrenzten Strom aufnehmen können. Zu einem Problem kann diese Tatsache führen, wenn beispielsweise ein Knoten in den Ruhezustand versetzt wird, um Energie zu sparen. Wird ein Knoten geweckt, um IPv6-Nachrichten zu empfangen, zu verarbeiten und weiterzuleiten, führt dies zu einer Verzögerung der Nachrichtenübertragung. Begründet durch die Zeit, die der Knoten braucht, um den Ruhezustand zu verlassen. Diese Verzögerungen und Leistungseinschränkungen werden in derzeitigen IPv6-Protokollen nicht berücksichtigt.

In der Vergangenheit wurden Technologien erarbeitet, um Herausforderungen in Bezug auf Konnektivität, Zuverlässigkeit, Sicherheit und Mobilität zu begegnen. Viele dieser komplexen Themenbereiche wurden durch IPv6-basierte Technologien gelöst.

Im Folgenden soll als Beispiel für eine solche IPv6-basierte Lösung für IoT-Geräte „IPv6 over Low power Wireless Personal Area Network“ (6LoWPAN) näher erläutert werden:

- 6LoWPAN

6LoWPAN ist ein Kommunikationsprotokoll, das speziell für Funkübertragung in Netzwerken mit Geräten mit beschränkten Ressourcen, wie CPU-Rechenleistung, Speicher sowie Energieversorgung, entwickelt wurde (RFC 4919 2007). Anstatt ein IoT-Anwendungsprotokoll wie Bluetooth oder ZigBee zu sein, ist 6LowPAN ein Netzwerkprotokoll, das Kapselungs- und Kopfkomprimierungsmechanismen definiert. Der Standard kann auch über mehrere Kommunikationsplattformen hinweg verwendet werden, beispielsweise Ethernet und Wi-Fi. Der 6LoWPAN Standard spezifiziert sowohl spezielle Möglichkeiten zur Header-Komprimierung, Fragmentierung und Defragmentierung einzelner Pakete, als auch Routing Algorithmen. Die Kernfunktionen von 6LoWPAN sind unter anderem 64-Bit und 16-Bit IEEE 802.15.4 Adressierung, effiziente Header-Kompression für IPv6, sowie UDP Header und Unterstützung des NDP (Brandt 2015). Darüber hinaus werden Unicast, Multicast und Broadcast unterstützt, wobei Multicast Pakete komprimiert als Broadcast verschickt werden.

Abbildung 15 zeigt die Einordung von 6LoWPAN in das OSI-Modell.

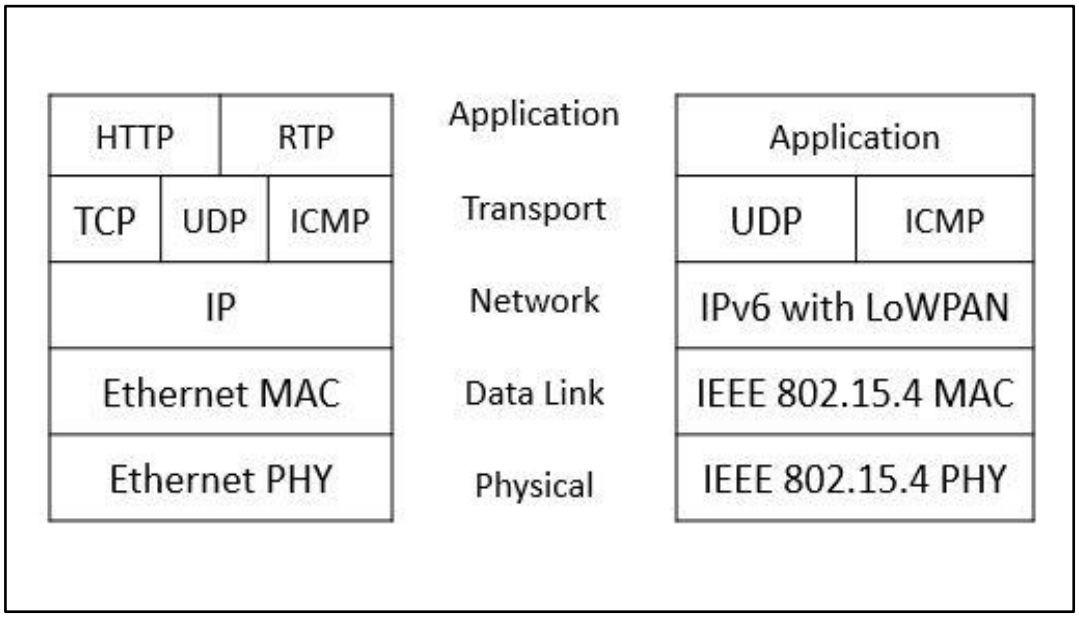


Abbildung 15: Einordnung von 6LoWPAN in das OSI-Modell (Brandt 2015).

Für das Internet der Dinge ist 6LoWPAN von entscheidender Bedeutung, da es viele der Probleme, welche in IPv6-Netzwerken mit mobilen, in ihren Funktionen reduzierten Geräten und in kabellosen Clients auftreten**,** löst. Geräte mit solchen Eigenschaften finden sich sehr häufig im Bereich von sogenannten Embedded Systems (Eingebettete Systeme). Dies sind Systeme, die einen sehr spezialisierten Funktionsumfang besitzen und dezidierte Aufgaben haben. Bekannte Beispiele für solche Systeme sind unter anderem Küchen- und Haushaltsgeräte, Geräte aus dem Bereich Heimentertainment aber auch medizinische Geräte.

Anhand des Beispiels von 6LoWPAN ist ersichtlich, dass der Einsatz von IPv6 in ressourcenschwachen Geräten die Entwicklung neuer Technologien vorantreibt und dadurch eine umfassende Verbreitung von IPv6 ermöglicht wird. Durch diese Entwicklung werden Limitierungen, welche aufgrund der eingeschränkten Leistungsdaten von IoT-Geräten bestehen, mehr und mehr minimiert. Es ist auf Grundlage der bisherigen technologischen Entwicklung davon auszugehen, dass in wenigen Jahren die geminderten Ressourcen von IoT-Geräten bezogen auf die Nutzung und Anwendung von IPv6 keine Rolle mehr spielen werden. Eine Limitation der Nutzung von IPv6 in IoT-Geräten kann daher nicht als relevante Einschränkung identifiziert werden.

4.3 IPv6-Konfiguration von IoT-Geräten

Ein entscheidender, limitierender Faktor für einen sicheren Einsatz von IoT-Geräten in IPv6-Netzwerken ist die eingeschränkte Möglichkeit, Manipulationen in den Einstellungen von IoT-Geräten tätigen zu können. Abbildung 16 zeigt das von Windows entwickelte „Windows 10 IoT Core Dashboard“ und die darin verfügbaren Netzwerk-Konfigurationsmöglichkeiten. Das Dashboard wurde speziell zur Verwaltung von IoT-Geräten entwickelt, die auf Windows 10 IoT Core als Betriebssystem aufbauen. Das Dashboard bietet eine effektive Möglichkeit, eine Vielzahl an IoT-Geräten in einem Netzwerk zentral zu verwalten.

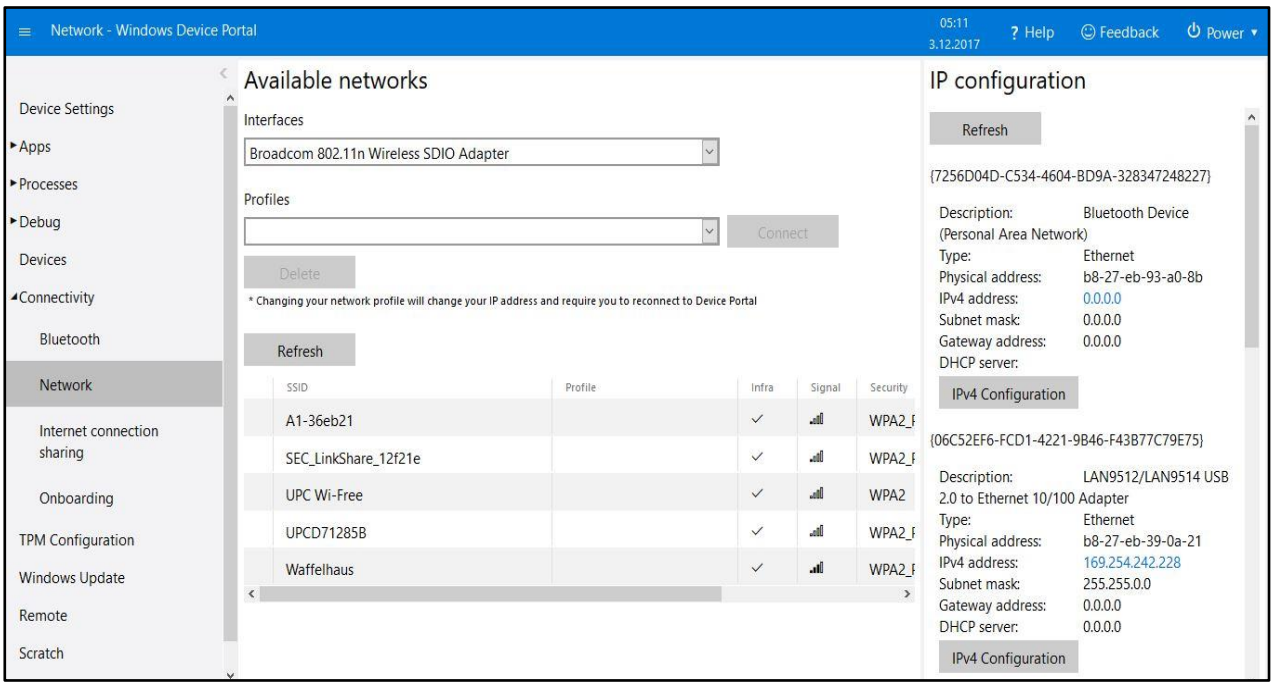


Abbildung 16: Windows IoT Core - Dashboard

Auch bei der Nutzung anderer, für IoT-Geräte optimierter Betriebssysteme beschränken sich die Konfigurationsmöglichkeiten direkt am Gerät meist auf die Zuweisung von IP-Adressen. Abbildung 17 zeigt das Command-line Interface von Ubuntu Core und die den Interfaces zugewiesenen IPv6-Adressen.

```
$ sudo ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:21:5a:50:d7:94
          inet addr:192.168.160.2  Bcast:192.168.160.255  Mask:255.255.255.0
          inet6 addr: fe80::221:5aff:fe50:d794/64 Scope:Link
          inet6 addr: 2001:db8:bad:a55::2/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

$ sudo ip -6 address show eth0
3: eth0: &lt;BROADCAST,MULTICAST,UP,LOWER_UP&gt; mtu 1500 qlen 1000
    inet6 2001:db8:bad:a55::2/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::221:5aff:fe50:d794/64 scope link
       valid_lft forever preferred_lft forever
```

Abbildung 17: zugewiesene IPv6-Adressen an IoT-Gerät mit Ubuntu Core

Diese Beispiele machen deutlich, dass die Absicherung von IoT-Geräten in IPv6-Netzwerken an den IoT-Geräten selbst nur bedingt möglich ist, besonders in Anbetracht der Tatsache, dass gezeigte IoT-Geräte vergleichsweise umfassende Möglichkeiten der Netzwerkkonfiguration bieten. Betrachtet man hingegen beispielsweise im Handel verfügbare IP-Kameras, bieten diese IoT-Geräte nur rudimentäre bis gar keine Möglichkeiten, Einstellungen zur Netzwerksicherheit zu manipulieren.

Es müssen daher bereits an den Netzwerksgrenzen entsprechende Maßnahmen getätigt werden, um schädlichen IPv6-Traffic gar nicht erst in das eigene Netzwerk gelangen zu lassen. Diese Maßnahmen betreffen hauptsächlich Einstellungen an Routern, Switches und Firewalls.

4.4 Resultierende Angriffsvektoren

Basierend auf den gewonnenen Informationen über IPv6 und IoT-Geräte der vorangegangenen Kapitel, lassen sich folgende Angriffsvektoren ableiten. Hierbei wird ausschließlich auf Szenarien eingegangen, welche durch die Verwendung von IPv6 begründet sind. Angriffe mit starken Ähnlichkeiten zu IPv4 werden in dieser Arbeit bewusst außer Acht gelassen.

- Identifikation und Aufklärung

  Die enorme Größe von Subnetzen in IPv6- Netzwerken erschwert das klassische Durchlaufen der Subnetzadressen, wie sie bei „Brute Force“-Attacken üblich sind.

  Bekannte Multicast-Adressen erleichtern jedoch das Auffinden von Schlüsselsystemen innerhalb eines Netzwerks (z.B. FF05::2 ist eine site-local Adresse vieler Router).

  Da davon auszugehen ist, dass viele NetzwerkadministratorInnen aufgrund mangelnder Erfahrung mit IPv6-Netzwerken anfangs auf gängige Adressen zurückgreifen, sollte es in den kommenden Jahren noch verhältnismäßig einfach sein, Schlüsselkomponenten wie beispielsweise Router und Nameserver in Netzwerken zu identifizieren.

- Unautorisierter Zugriff

  Viele neue Überlegungen hinsichtlich der Filtermöglichkeiten von ICMP, Multicast, IPsec und Erweiterungsheadern sind für IPv6 von Nöten. Besonders die Verwendung von Erweiterungsheadern bietet eine Möglichkeit, um Sicherheitsrichtlinien zu umgehen. Da die in IPv4 gängige Praxis, ICMP-Pakete „ad-hoc“ an der lokalen Netzwerkgrenze zu blockieren, um einen Sicherheitsgewinn zu erzielen, in IPv6-Netzwerken nicht möglich ist, stellen solche Attacken einen völlig neuen Angriffsvektor dar, auf den sich administrierende Personen bisher nicht vorbereitet haben.

- Header Manipulation und Fragmentierung

  Fragmentierung wird nicht mehr von Vermittlungsgeräten durchgeführt und MTU-Erkennung ist erforderlich. Das bedeutet, dass verschiedene Erweiterungsheader-Optionen die traditionelle Fragmentierungswiederherstellung, wie sie heute von Netzwerkgeräten gemacht wird, erschweren können. Gemäß (RFC 2460 1998) ist die IPv6-Mindest-MTU 1280 Oktette. Aus diesem Grund können administrierende Personen dem Sicherheitsgerät erlauben, Fragmente mit weniger als 1280 Oktetten fallen zu lassen, es sei denn, das Paket ist das letzte Paket im Fluss. Administrierende Personen können diese Aktion ausführen, wenn das sendende Betriebssystem das ursprüngliche Paket an der von den „Path MTU Descovery“-Nachrichten bereitgestellten MTU fragmentiert und weiterhin diese Größe von IPv6-Fragmenten erstellt, bis das letzte Segment des ursprünglichen Pakets übermittelt wird. Wenn sich das Host-Betriebssystem nicht auf diese Weise verhält, muss das Sicherheitsgerät weiterhin IPv6-Fragmente mit weniger als

1280 Oktetten akzeptieren und verarbeiten. Dieses Verhalten würde weiterhin die Verschleierung von Angriffen ermöglichen, indem große Mengen kleiner fragmentierter Pakete gesendet werden. Um das Potenzial dieser Filterung zu validieren, ist es notwendig, das Fragmentierungs- und Reassembly-Verhalten populärer Betriebssysteme zu integrieren.

- Layer 3 und Layer 4 Spoofing

  Ein Schlüsselelement, das zahlreiche verschiedene Arten von IP-Angriffen ermöglicht, ist die Fähigkeit eines angreifenden Geräts, seine Quell-IP-Adresse und die Ports, über die es kommuniziert, so zu verändern, als würde der IP-Angriff von einem anderen Standort oder einer anderen Anwendung initiiert. Diese sogenannten „Spoofing-Attacken“ stellen auch in IPv6-Netzwerken eine präsente Bedrohung dar.

  Vom Standpunkt verschiedener Übergangstechnologien aus, bieten die verschiedenen Tunnelmechanismen einem gegnerischen Gerät mit entweder IPv4- oder IPv6-Konnektivität die Möglichkeit, Verkehr an die andere IP-Version zu senden, während die wahre Quelle maskiert wird. Beispielsweise können AngreiferInnen 6to4-Relay-Router verwenden, um Datenverkehr in ein IPv6-Netzwerk einzubringen, ohne befürchten zu müssen, dass die Datenpakete bis zur wahren Quelle zurückverfolgt werden können.

- ARP and DHCP Attacken

  IPv4-ARP-Angriffe werden durch IPv6-ND-Angriffe mit ungefähr denselben Problemen ersetzt. IPv4-DHCP-Angriffe werden zusätzlich zu herkömmlichen DHCP-Problemen für IPv6 durch zustandslose Autokonfigurationsangriffe unterstützt. Da die zustandslose Autokonfiguration (eine einfache DHCP-ähnliche Funktionalität in ICMPv6) in vielen Fällen eine praktikable Alternative zu DHCP darstellt, sind dedizierte DHCP-Server in IPv6 nicht üblich und in IoT-Netzwerken nicht weit verbreitet. Dedizierte DHCPv6-Server werden möglicherweise angezeigt, um zusätzliche Konfigurationsparameter wie DNS-Server, Zeitserver usw. anzubieten, sodass weiterhin ein DHCP-Schutzniveau erforderlich ist. Leider können „Stateless Autoconfiguration“-Nachrichten gefälscht werden und Spoofing kann verwendet werden, um den Zugriff auf Geräte zu verweigern. Um dies zu mindern, sollte das „Trusted-Ports“-Konzept in Verbindung mit „Router-Advertised-Messages“ verwendet werden.

- Smurf-Attacken (Verstärkung von Broadcasts)

  Es gibt kein IPv6-Äquivalent eines IPv4-Broadcast-Pakets, das herkömmliche Smurf-Attacken unmöglich macht. Dies bedeutet, dass auch so genannte Fraggle-Attacken, das Auslösen eines UDP-Broadcasts und das Generieren von Antworten an ein vermeintliches Opfer, unter IPv6 immer noch möglich sind.

In IPv6 wird das Konzept einer IP-gerichteten Rundsendung aus dem Protokoll entfernt. Dem Protokoll wird eine spezifische Sprache hinzugefügt, welche entwickelt wurde, um diese Arten von Angriffen zu limitieren. Speziell in Bezug auf einen Smurf-Angriff gibt (RFC 4443 2006) an, dass eine ICMPv6-Nachricht nicht als Antwort auf ein Paket mit einer IPv6-Multicast-Zieladresse, einer Link-Layer-Multicast-Adresse oder einer Link-Layer-Broadcast-Adresse generiert werden sollte.

- Routing-Attacken

  Routing-Attacken konzentrieren sich darauf, den Verkehrsfluss in einem Netzwerk zu unterbrechen oder umzuleiten. Dies wird auf verschiedene Arten erreicht, angefangen von Überschwemmungsangriffen, schneller Ankündigung und Entfernung von Routen, bis hin zur falschen Ankündigung von Routen. Einzelheiten der Angriffe variieren je nach verwendetem Protokoll.

  IPv6-Routing-Protokolle verwenden zur Sicherung der Integrität und Vertraulichkeit IPsec, um den Transport im Gegensatz zu anwendungsspezifischen Schutzmechanismen (d. H. MD5) zu sichern.

  Die Sicherheitsmechanismen zum Sichern von Protokollen, die sich mit IPv6, Open Shortest Path First version 3 (OSPFv3) und Routing Information Protocol next generation (RIPng) geändert haben, werden inkonsistent über Internetworking-Anbieter implementiert.

- Translation, Transition, und Tunneling Mechanismen

  Verschiedene Techniken in diesem Bereich erzeugen neue Angriffsvektoren rund um Spoofing, Redirecting, Flooding und Encapsulating Traffic. Es wird viel Wert darauf gelegt, NAT nicht zu benötigen, aber es werden viele Organisationen weiterhin darauf beharren, NAT in ihren Sicherheitskonzepten zu verwenden.

Diese Angriffsvektoren stellen sowohl NetzwerkadministratorInnen als auch Provider bei der Umstellung auf IPv6 vor neue Herausforderungen. Zwar sind mache Angriffsszenarien sehr ähnlich zu IPv4, doch kommen in IPv6-Netzwerken häufig völlig andere Methoden zum Einsatz, wodurch völlig neue Sicherheitskonzepte notwendig werden.

Die vorliegenden Beispiele zeigen jedoch auch, dass IPv6-Security für IoT-Geräte schwerpunktmäßig in Netzwerkgeräten wie Routern, Switches und Firewalls realisiert werden muss. Betrachtet man die oben angeführten Beispiele, ist eine Absicherung direkt am IoT-Gerät nur in beschränktem Ausmaß bis gar nicht realisierbar. Es ist zwar möglich an IoT-Geräten grundlegende Sicherheitseinstellungen vorzunehmen, wie das Verwenden von Privacy Extension oder dem klassischen Ändern von Standardpasswörtern, umfassender Schutz ist jedoch nur möglich, wenn bereits an den Netzwerkgrenzen Maßnahmen gesetzt werden, um verdächtige IPv6 Pakete zu verwerfen, bevor sie in das interne Netzwerk vorgedrungen sind.

Basierend auf den bis hier gewonnenen Ergebnissen lässt sich sagen, dass eine Absicherung von IoT-Geräten in IPv6-Netztwerken durch alleinige Manipulation der Endgeräte nicht möglich ist. Es ist notwendig, umfassende Maßnahmen an Routern und Firewalls vorzunehmen, um den Schutz von IoT-Geräten in solchen Netzwerken zu gewährleisten.

Aus diesem Grund wird im weiteren Verlauf der Arbeit auch besonderes Augenmerk auf die Möglichkeit der Konfiguration und Absicherung von Netzwerkgeräten gelegt, um das gesetzte Ziel der vollständigen Absicherung von IoT-Geräten zu erreichen.

4.5 Potentielle Schutzmechanismen

Der folgende Abschnitt geht erneut auf die im vorangegangen Teil angeführten Angriffsvektoren ein und versucht, auf Basis der erarbeiteten und bisher erlangten Kenntnisse über IPv6 IoT-Geräte, potentielle Schutzmechanismen zu erarbeiten, die im anschließenden praktischen Teil anhand einer definierten Laborumgebung bestätigt werden sollen.

Basierend auf den bisherigen Ergebnissen lassen sich folgende Methoden zum Schutz von IoT-Geräten in IPv6-Netzwerken ableiten:

- Identifikation und Aufklärung
    - Implementierung von Privacy Extensions

    Obwohl Privacy Extensions im Hinblick auf Scanning-Angriffe einen Vorteil bieten, können sie es auch erschweren, Probleme zu verfolgen und Probleme in einem Netzwerk zu beheben. Wenn sich in einem Netzwerk ein fehlerhafter Host befindet und sich die Adresse des Hosts regelmäßig ändert, kann es sehr schwierig sein, den genauen Host zu ermitteln oder festzustellen, ob die Probleme von einem oder mehreren Hosts stammen. Bessere Optionen sind die Verwendung MAC-Adressen-basierter, statischer Adressen für die interne Kommunikation und Pseudozufalls-Adressen für den für das Internet bestimmten Datenverkehr.

- Verwendung nicht offensichtlicher, statischer Adressen für kritische Systeme

Anstatt Hostadressen wie ::10 oder ::20 zu standardisieren, ist es ratsam, sensible Infrastrukturelemente mit Adressen zu versehen, welche systematisch verteilt werden, jedoch ohne gängige, bekannte Muster. z. B. ::DEF1 für Standardgateways. Dadurch wird es dem gegnerischen Gerät erschwert durch simples „Durchprobieren“ von Standardadressen auf Router und andere Knoten zu gelangen.

- Selektives Filtern von ICMP

Da Neighbor Discovery ICMP verwendet und die Fragmentierung nur auf Endstationen geschieht, ist es unerlässlich, dass einige ICMP-Nachrichten in IPv6 zugelassen werden. Das heißt, nicht-essentielle ICMP-Nachrichten können und müssen an einer Firewall gefiltert werden, ebenso wie ICMP-Echo und Echo-Reply Nachrichten, wenn dieser Aspekt der Verwaltbarkeit geopfert werden kann. Darüber hinaus benötigt IPv6 ICMPv6 Neighbor-Discovery-Neighbor Solicitation (ND-NS) und Neighbor-Discovery-Neighbor-Advertisement (ND-NA) Nachrichten, um zu funktionieren (RFC 4861 2007) (beschrieben in Abschnitt 3.1.3), sowie Router-Solicitation (RS) und Router- Advertisement-Nachrichten (RA-Nachrichten), wenn Autokonfiguration verwendet wird.

- Wahrung der Host- und Anwendungssicherheit

Obwohl rechtzeitiges Patching und Host Lockdown kritische Elemente in der Anfangsphase von IPv6 sind, ist dieser Aspekt besonders bei IoT-Geräten problematisch, da IoT-Geräte häufig nur sehr eingeschränkt supported werden. Es ist notwendig, sich auf die Aufrechterhaltung der Hostsicherheit zu konzentrieren, um sicherzustellen, dass Hosts, die kompromittiert sind, keine Trittsteine werden, um andere Endhosts zu gefährden.

- Unautorisierter Zugriff

Um unautorisiertem Zugriff bestmöglich vorzubeugen, ist es notwendig zu definieren, welche ICMPv6-Nachrichten erforderlich sind. Besonders zu berücksichtigen sind folgende ICMPv6 Nachrichten:

ICMPv6 Typ 2 –	Paket zu groß
ICMPv6 Typ 4 –	Parameterproblem
ICMPv6 Typ 130-132 –	Multicast-Listener
ICMPv6 Typ 133/134 –	Router Solicitation und Router Advertisement
ICMPv6 Typ 135/136 –	Neighbor Solicitation und Neighbor Advertisement

- Header Manipulation und Fragmentierung

  Um Fragmentierungsangriffen bestmöglich entgegenzuwirken, können folgende Maßnahmen ergriffen werden:

  - Verweigerung von IPv6-Fragmenten für Internetworking-Geräte. Dadurch werden bestimmte Angriffe auf das Gerät eingeschränkt
  - Sicherstellung angemessener IPv6-Fragmentierungsfilterfunktionen

  Die Kombination mehrerer Erweiterungsheader und eine Fragmentierung in IPv6 erzeugen das Potenzial, dass das Layer 4-Protokoll nicht in das erste Paket eines Fragmentsets aufgenommen wird.

  - Löschung aller Fragmente mit weniger als 1280 Oktetten (außer dem letzten)

- Layer 3 und Layer 4 Spoofing
  - Verwendung kryptografischer Schutzmechanismen

  Wenn eine Anwendung einen starken kryptografischen Schutz verwendet, ist ein erfolgreicher Spoofing-Angriff bedeutungslos, da durch den Einsatz kryptographischer Technologien ein Bruch der Integrität und Authentizität der Daten festzustellen ist.

- ARP und DHCP-Attacken
  - Nutzung von „Static Neighbor Entries“

  In sehr sensiblen Umgebungen können statische Einträge für Standardrouter festgelegt werden, wodurch viele der typischen Neighbor-Discovery-Angriffe vermieden werden können.

- Smurf-Attacken (Verstärkung von Broadcasts)
  - Filterung von Paketen mit IPv6-Multicast-Quelladressen

  Es gibt keinen triftigen Grund, der einen Eintritt einer Multicast-Quelladresse in das eigene Netzwerk rechtfertigt. Daher sollten alle Pakete mit einer Multicast-Quelladresse an der Grenze des Netzwerks verworfen werden.

- Routing-Attacken
  - Verwendung von IPv6-Hop-Limits zum Schutz von Netzwerkgeräten
  - Verwendung von IPSec zum Sichern von Protokollen wie OSPFv3

- Translation, Transition, und Tunneling Mechanismen

  Da in dieser Arbeit von reinen IPv6-Netzwerken ausgegangen wird, wird dieser Punkt für das weitere Vorgehen in dieser Arbeit außer Acht gelassen.

4.6 Zusammenfassung

Auf Grundlage der gesammelten Ergebnisse aus der durchgeführten Recherche an Fachliteratur und etablierten Standards, lassen sich folgende, für die vorliegende Arbeit relevanten Schlüsse ableiten:

- IoT-Geräte dringen immer weiter in unseren Alltag vor und bestimmen mehr und mehr unser tägliches Handeln. Darüber hinaus spielen IoT-Geräte eine immer größer werdende Rolle in sensiblen Bereichen wie dem Gesundheitswesen und werden mehr und mehr Teil von kritischer Unternehmensinfrastruktur.

- IoT-Geräte bieten aufgrund ihrer Bauart und spezialisierten Hard- und Software nur sehr eingeschränkte Möglichkeiten der umfassenden Absicherung gegenüber potentiellen Gefahren, wie sie durch die Einführung von IPv6 auf NutzerInnen von IoT-Geräten zukommen werden.

- Darüber hinaus unterliegen IoT-Geräte, aufgrund von schnellen Produktlebenszyklen und oftmals spezialisierter Firmware, nur sehr eingeschränkt bis kaum den vorhandenen Support-Programmen. Dies führt dazu, dass in Verwendung befindliche IoT-Geräte keine Sicherheitsupdates durchlaufen, wodurch es zu umfassenden und schwerwiegenden Sicherheitslücken in Unternehmensnetzwerken kommen kann. Durch diese Sicherheitslücken können AngreiferInnen lange bekannte Schwachstellen ausnutzen und sich Zugriff auf unternehmensrelevante Daten verschaffen, oder einen Ausfall der Infrastruktur eines Unternehmen verursachen und bedeutenden Schaden anrichten.

- IPv6 öffnet potentiellen AngreiferInnen ein breites Feld an neuen Möglichkeiten, gezielt einzelne Knoten zu attackieren oder ganze Unternehmensnetzwerke anzugreifen und zu manipulieren.

- Die aufgelisteten Risiken und Gefahren, die durch die Nutzung von IPv6 auf NetzwerkadministratorInnen zukommen, betreffen großteils Netzwerkkomponenten wie Router und Layer-3-Switches. Auf Grundlage der erlangten Erkenntnis der eingeschränkten Konfigurierbarkeit von IoT-Geräten gilt es, diesen Komponenten bei der Umsetzung eines IPv6-Netzwerkes, in denen IoT-Geräte betrieben werden, ein besonderes Augenmerk zu schenken.

Die Resultate der Literaturrecherche und die daraus abgeleiteten Ergebnisse lassen bereits eine erste Beantwortung der Forschungsfrage zu, die wie folgt lautet:

„**Welche Risiken entstehen durch die Verwendung von IPv6 auf IoT-Geräten bezogen auf IT-Security, und wie lassen sich diese neu entstandenen Sicherheitsrisiken durch entsprechende Konfiguration der IoT-Geräte bestmöglich minimieren**“

Durch den Einsatz von IPv6 kommt es in Netzwerken zu neuen Bedrohungsszenarien, die im Besonderen für IoT-Geräte relevant sind, da auf Grund mangelnder Sicherheitsmechanismen eine Absicherung dieser nur im eingeschränktem Ausmaß möglich ist. Zu den neu aufkommenden Risiken zählen:

- Die Möglichkeit der einfachen Identifikation durch „hardcodiert“ IP-Adressen von IoT-Geräten auf Basis von Hersteller- und Produktidentifikation.
- Unautorisierter Zugriff durch ICMP-Erweiterungsheader
- Verschleierung von Angriffen durch Headermanipulation und Fragmentierung von Datenpaketen
- Verschleierung des Ursprunges eines Angriffes durch Layer 3 und Layer 4 –Spoofing
- Fälschung von „Stateless Autoconfiguration“-Nachrichten und darauf basierende ARP und DHCP – Attacken
- Gefahr von SMURF-Attacken aufgrund der großen Anzahl von IoT-Geräten welche korrumpiert werden können.
- Routing und Translationsattacken aufgrund der Weiterverwendung von NAT oder unzureichender Implementierung von Translationsmechanismen.

Diese genannten Risiken stellen eine besondere Gefahr für IoT-Geräte dar. Aufgrund der in der bisherigen Arbeit gewonnenen Erkenntnis, dass IoT-Geräte eine sehr eingeschränkte Möglichkeit der Manipulation der IPv6-Einstellungen bieten, kann ein umfassender Schutz von IoT-Geräten vor den genannten Bedrohungen nicht realisiert werden.

Diese vorläufige Beantwortung der Forschungsfrage macht deutlich, dass eine Anpassung der Methodik dieser Arbeit notwendig ist. Aufgrund der gewonnenen Erkenntnisse wird von einem Laborversuch abgesehen und der Eingriff in die IPv6-Konfiguration der IoT-Geräte nicht weiter berücksichtigt. Auf Basis des generierten Wissens wird deutlich, dass eine Absicherung von IoT-Geräten bereits an der Netzwerksgrenze stattfinden muss, um eine böswillige Manipulation von IoT-Geräten zu unterbinden. Aus diesem Grund soll in der vorliegenden Arbeit wie folgt vorgegangen werden:

- Erstellung eines Frameworks zur effizienten Umsetzung eines IPv6-Netzwerkes zum sicheren Betrieb von IoT-Geräten durch:
    - Identifikation der Systemgrenzen
    - Definition der Anforderungen
    - Ausarbeitung konkreter Empfehlungen zur Umsetzung der Anforderungen
    - Bündelung der Empfehlungen zu einem zu bestehenden Guidelines und Richtlinien kohärenten Frameworks

Die weitere Arbeit soll in die Erstellung eines Security Frameworks für IPv6-Netzwerke für IoT-Geräte münden. Auf die Evaluierung des Frameworks soll in dieser Arbeit nicht eingegangen werden, da diese als Gegenstand weiterer zukünftiger Forschung dienen soll.

5 IPV6/IOT-SECURITY FRAMEWORK

Dieses Kapitel beschreibt die Erstellung des IPv6/IoT-Security-Frameworks, das NetzwerkadministratorInnen zukünftig bei der Erstellung einer IPv6-Netzwerkinfrastruktur, die einen sicheren Einsatz von IoT-Geräten gewährleistet, unterstützen soll. Die Basis für dieses Framework bilden die vorangegangenen Kapitel, in denen ein Grundwissen für IPv6 und IoT-Geräte vermittelt und damit verbundene Bedrohungsszenarien erarbeitet wurden.

Daraus ergab sich im Besonderen, dass die höchste Priorität der Absicherung der Netzwerkgrenzen gilt. Dies betrifft speziell Router und Layer 3 Switches.

Die Zielgruppe des resultierenden Frameworks soll hauptsächlich aus klein- und mittelständischen Unternehmen bestehen, die über eine relativ kleine Netzwerkinfrastruktur und begrenztes IT-Budget verfügen. Die Umsetzung des erarbeiteten Frameworks soll einen möglichst umfassenden Schutz gegenüber IPv6-Attacken gegen IoT-Geräte bieten, aber auch bestehende IT-Infrastruktur bestmöglich gegen Angriffe von außerhalb schützen.

Zu diesem Zweck soll in Abschnitt 5.1 eine Definition des Begriffes „Framework“ erarbeitet und genauer konkretisiert werden. Abschnitt 5.2 gibt einen genauen Einblick in die Systemgrenzen, um zu definieren welchen Einfluss das Framework auf die IT-Infrastruktur hat, und welche Komponenten davon weitgehend unberührt bleiben werden. Im Abschnitt 5.3 werden die Anforderungen an das Framework konkret definiert, um im Abschnitt 5.4 Möglichkeiten darzulegen, wie diese Anforderungen umgesetzt werden können. Der letzte Abschnitt dieses Kapitels fasst schließlich die erarbeiteten Empfehlungen zu einem umfassenden Framework zusammen.

5.1 Definition

Der Begriff Framework wird in der IT-Branche sehr häufig verwendet. Laut (Heller 2009) ist ein Framework ein Konstrukt, welches sich aus einem „Skelett“ und einer Bibliothek zusammensetzt. Die Bibliothek beinhaltet verschiedenste Lösungen eines bestimmten Lösungsbereiches. Das „Skelett“ bildet laut Heller die strukturelle Schnittmenge der verschiedenen Lösungen (Abbildung 18).

Da es relativ großen Aufwand erfordert, die Schnittmenge verschiedener Lösungen zu modellieren, wird angestrebt, dass das Ergebnis im Anschluss mit möglichst geringem Aufwand rekonstruiert werden kann, sprich, dass ein hoher Grad an Wiederverwendbarkeit gegeben ist.

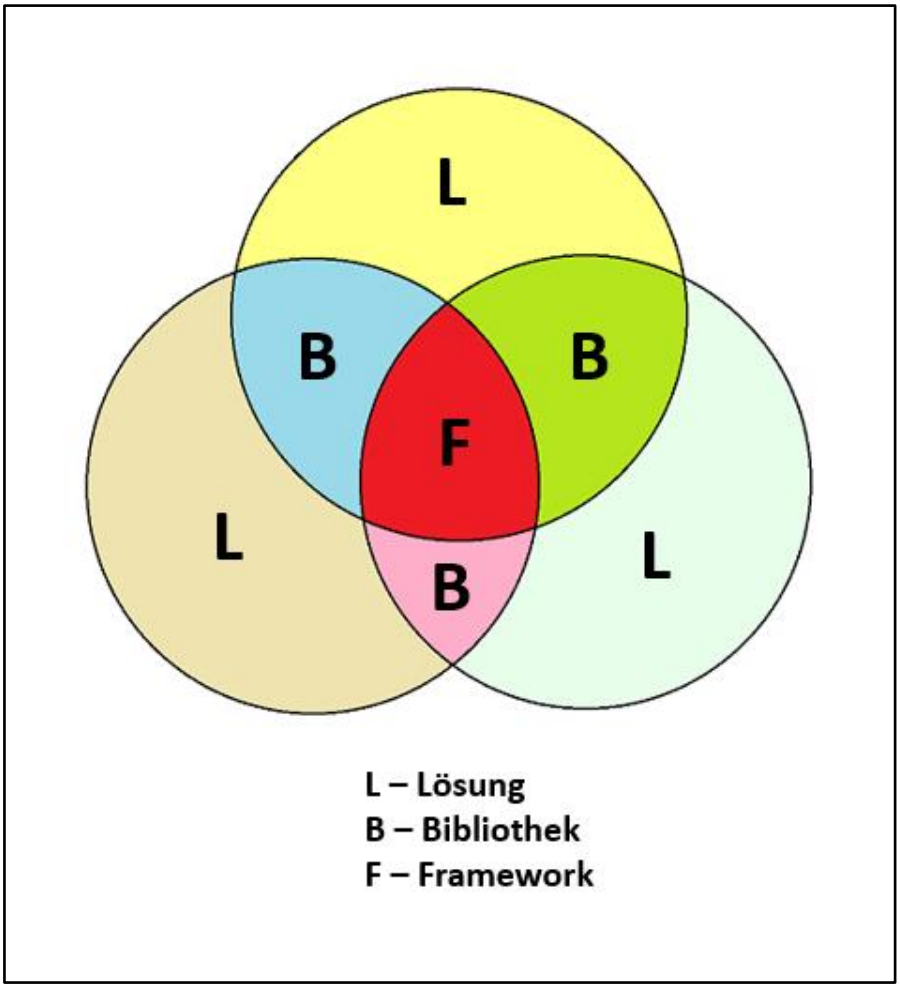


Abbildung 18: Definition eines Frameworks nach (Heller 2009).

Dies wird in der Regel durch den Versuch erreicht, mit dem resultierenden Framework einen möglichst großen Bereich an Lösungen abzudecken. Dadurch wird der Bereich der Überschneidungen der Lösungen verkleinert und somit der Umfang des Frameworks reduziert.

5.2 Identifikation der Systemgrenzen

Das aus dieser Arbeit mündende Framework soll zum Ziel haben, administrierende Personen von Unternehmensnetzwerken bei der Überführung ihrer Infrastruktur insofern zu unterstützen, als dass es ihnen als Leitfaden bei der Überführung einer bestehenden IPv4-Infrastruktur hin zu einer zukunftsorientierten IPv6-Infrastruktur, mit besonderem Augenmerk auf die Sicherheitserfordernisse von IoT-Geräten, dienen soll. Dabei soll das Framework möglichst breit konzipiert werden, so dass es für einen Großteil der Unternehmensnetzwerke angepasst werden kann. Auch soll ein besonderer Fokus auf die Möglichkeit der zukünftigen Erweiterung des Frameworks auf Grundlage weiterer Forschung gelegt werden. Der Einflussbereich des zu erstellenden Frameworks limitiert sich primär auf die Netzwerkkomponenten der Infrastruktur.

Abbildung 19 zeigt in welchem Bereich das Framework Anwendung findet und welche Komponenten des Unternehmensnetzwerkes es umfasst.

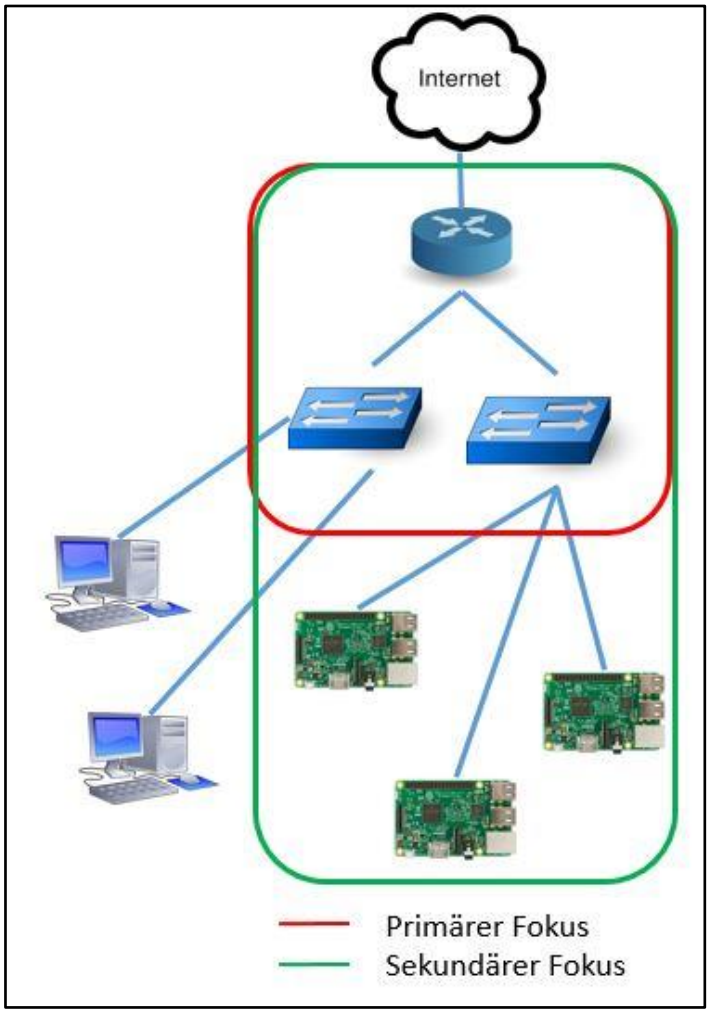


Abbildung 19: Grenzen der Einflussbereiche des Frameworks

Aus der Abbildung geht hervor, dass der primäre Fokus des Frameworks auf Layer 2 und Layer 3 Komponenten der Netzwerkinfrastruktur gerichtet ist. Das Framework soll die administrierenden Personen bei der Absicherung des Unternehmensnetzwerkes nach außen hin unterstützen und interne Netzwerkkomponenten dadurch absichern, dass das Risiko eines möglichen Eindringens eines angreifenden Geräts durch Ausnutzung von Schwachstellen des IPv6-Protokolles weitgehend minimiert wird. Außerdem ist ersichtlich, dass der sekundäre Fokus des Frameworks auf der Absicherung von IoT-Geräten gerichtet ist. Das Framework hegt keinen Anspruch auf Absicherung anderer Infrastrukturkomponenten als Netzwerkgeräte und IoT-Geräte.

5.3 Definition der Anforderungen

Nachfolgend sollen die Anforderungen an das Framework umfassend definiert werden. In erster Linie soll bereits eine erste Segmentierung des zu erstellenden Frameworks auf Basis der ermittelten Anforderungen erfolgen. Diese Anforderungen basieren auf der im ersten Teil dieser Arbeit durchgeführten Literaturrecherche und den in dieser Phase ermittelten Risiken, die durch den Einsatz von IPv6 in Netzwerken mit IoT-Geräten entstehen.

Um ein sicheres IPv6-Netzwerk für IoT-Geräte zu schaffen, ist es daher notwendig, die folgenden Anforderungen hinreichend zu definieren und in weiterer Folge zu erfüllen (Abbildung 20).

Abbildung 20: Definition der Anforderungen an das Framework

- Absicherung gegen Aufklärung und Identifikation

  Der erste Aspekt der auf Basis der bisherigen Erkenntnisse zu erfüllen ist, ist die Absicherung des Netzwerkes gegen Aufklärungsangriffe und der damit verbundenen eindeutigen Identifizierung von Komponenten innerhalb des Netzwerkes. Die vorangegangene Arbeit zeigt, dass der in IPv6 zur Verfügung stehende Adressraum trotz seiner enormen Größe per se keinen ausreichenden Schutz gegen solche Angriffe darstellt. Deshalb sind in diesem Bereich Aktionen der NetzwerkadministratorInnen erforderlich.

- Absicherung gegen unautorisierten Zugriff

  Dieser Aspekt ist dann erfüllt, wenn alle nötigen ICMPv6-Nachrichten, die in der Netzwerkkommunikation eine Rolle spielen, identifiziert und definiert wurden. Alle nicht in der Netzwerkkommunikation definierten Nachrichten sind im Netzwerkverkehr zu unterbinden, um einem unautorisierten Zugriff vorzubeugen.

- Schutz gegen Fragmentierung und Manipulation des Headers

  Ein besonders kritischer Aspekt von IPv6, der eine breite Palette von Angriffen ermöglicht, ist die Möglichkeit der Manipulation des IPv6-Headers. Auch die Tatsache, dass eine Fragmentierung des IPv6-Headers möglich ist, stellt ein Risiko für Netzwerke dar. Daher ist es erforderlich, dass jegliche Manipulation des Headers von Netzwerkkomponenten erkannt und betreffende Pakete verworfen werden.

- Schutz vor Spoofing (Layer 3 und 4)

  Der Schutz vor Spoofing-Attacken, sowohl auf Layer 3 als auch auf Layer 4 Ebene, ist ein entscheidender Faktor für die Sicherheit eines Netzwerkes. Um dies zu gewährleisten, besteht die Möglichkeit, etablierte kryptographische Technologien einzusetzen. Als zu erfüllende Anforderung zur Prävention solcher Attacken wird daher festgelegt, dass verpflichtend ein kryptographisches Verfahren zur Anwendung kommen muss.

- Schutz gegen ARP und DHCP-Attacken

  Obwohl das Address Resolution Protocol (ARP) in IPv6 durch das sogenannte Neighbor Discovery Protocol (NDP) ersetzt wurde, sind viele der bekannten Angriffsvektoren auch im neuen Internetprotokoll immer noch von bedeutender Relevanz. Eine Absicherung gegen diese Art von Attacken ist daher von immenser Bedeutung.

- Verhinderung von Smurf- und anderen DoS-Attacken

  Besonders im Bereich von IoT-Geräten spielen Smurf-Angriffe eine immer bedeutendere Rolle. Bei dieser Form einer DoS-Attacke werden gezielt ICMP-Pakete des Typs „Echo Request“ an alle Teilnehmenden des Netzwerkes gesendet und als Absender wird stets die Adresse des zu attackierenden Knotens eingetragen (Satapathy et al. 2016). Infolge dessen wird dieser danach mit einer Flut von Ping-Antworten überlastet. Da ICMP-Pakete in IPv6-Netzwerken eine wichtige Rolle spielen, dürfen sie nicht per se ausgefiltert werden. Es ist jedoch erforderlich zu definieren, welche Arten von ICMP-Paketen an das Netzwerk weitergeleitet und welche verworfen werden sollen.

  Durch den ständig steigenden Datenverkehr im Internet und die damit verbundene zunehmende Belastung von Netzwerkkomponenten, können diese durch solche Angriffe zum Erliegen gebracht werden. Die Unterbindung solcher Angriffe stellt eine entscheidende Notwendigkeit für zukunftsorientierte Netzwerke dar.

- Abwehr von Routing-Attacken

  Die Abwehr von Routing-Attacken ist primär eine Aufgabe von Internetanbietern und daher keine Priorität des Frameworks. Aufgrund der Vollständigkeit und der Anforderung an das Framework auch für größere Unternehmensnetzwerke anwendbar zu sein, sei dennoch die Anforderung einer größtmöglichen Unterbindung von Routing-Attacken festgelegt.

5.4 Möglichkeiten der Umsetzung

Dieses Kapitel beinhaltet die möglichen Lösungsansätze zur Umsetzung der zuvor definierten Anforderungen. Durch diese Bündelung von Lösungen soll es den zukünftigen AnwenderInnen des Frameworks möglich sein, aus mehreren verfügbaren Möglichkeiten die für ihr Netzwerk am besten geeignete Methode für die Erreichung der gesetzten Sicherheitsziele auszuwählen.

- Identifikation und Aufklärung

  Um die Identifikation von Netzwerkadressen und den dahinter liegenden Komponenten für AngreiferInnen zu erschweren, wurden im Zuge der Literaturrecherche folgende Lösungsmöglichkeiten aufgezeigt, die nun zusammengefasst dargestellt werden sollen:

  - Verwendung von Privacy Extension

  Durch die Nutzung von Privacy Extensions ist es möglich die Koppelung von MAC-Adressen und des Interface-Identifiers aufzuheben. Somit können quasi zufällige Interface-Identifier erzeugt werden, die einen hohen Grad an Anonymität gewährleisten.

  Dadurch wird es erschwert anhand der IPv6-Adresse Rückschlüsse auf dahinterliegende Komponenten zu ziehen. Somit ist es möglich eine Anonymität, ähnlich der von NAT in IPv6-Netzwerken, herzustellen. Der Vollständigkeit halber muss aber erwähnt werden, dass Privacy Extensions besonders in kleinen Netzwerken keine absolute Identität gewähren, da mit dieser Methode zwar der Hostteil von IPv6-Adressen regelmäßig gewechselt wird, der Präfix jedoch derselbe bleibt. In Netzwerken mit wenigen Knoten wären einige somit durch den Präfix der Adresse immer noch erkennbar. Wenn sich Personen, die mit der Administration von Netzwerken betraut sind, für Privacy Extensions entscheiden, so ist es empfehlenswert, dass diese nur an Clients aktiviert werden. Die Nutzung von Privacy Extensions an Netzwerkkomponenten kann sowohl das Managen des Netzwerkes als auch eine mögliche Fehlersuche unnötig erschweren.

- Verwendung nicht offensichtlicher, statischer Adressen für kritische Systeme

Ein weiterer Ansatz zur Wahrung der Anonymität von Netzwerkkomponenten ist die Verwendung von nicht offensichtlichen Adressen für kritische Komponenten. Im Laufe der Zeit haben sich in der Branche Verhaltensweisen und ungeschriebene Standards in Bezug auf die Vergabe von statischen Netzwerkadressen manifestiert. Dies ist nicht per se als schlecht zu bezeichnen, jedoch kann es bereits einen entscheidenden Sicherheitsgewinn darstellen, wenn die systematische Verteilung von Adressen nach keinen gängigen und bekannten Mustern erfolgt. Hier sind NetzwerkadministratorInnen gefordert, frühzeitig ein Konzept zu erstellen, das auch ein mögliches späteres Wachstum der Infrastruktur berücksichtigt.

- Wahrung der Host- und Anwendungssicherheit

Wie bereits im Kapitel 4 dieser Arbeit erarbeitet, spielen besonders in der Anfangsphase von IPv6 und im Speziellen bei der Nutzung von IoT-Geräten, das Patchen und Updaten von Netzwerkkomponenten und Clients eine kritische Rolle. Da IoT-Geräte oftmals nur eingeschränkt und nicht automatisiert supported werden, ist es von entscheidender Bedeutung sich auf die Aufrechterhaltung der Hostsicherheit zu konzentrieren. Aus diesem Grund soll in das zu erstellende Framework auch ein Plan Do Check Act-Zyklus (PDCA) aufgenommen werden, um in regelmäßigen Abständen zu überprüfen, ob alle Komponenten mit der jeweils aktuellen Software betrieben werden.

- Unautorisierter Zugriff

IMCP-Nachrichten spielen in IPv6 eine entscheidende Rolle bei der Kommunikation und dem Informationsaustausch zwischen Routern. Diese ICMPv6-Pakete beinhalten hauptsächlich Statusinformationen und Fehlermeldungen und sind von großer Bedeutung für die Funktionsweisen von IPv6-Verbindungen. Diese ICMPv6-Nachrichten können jedoch auch von AngreiferInnen genutzt werden, um in ein IPv6-Netzwerk einzudringen. Um diesen Missbrauch zu unterbinden, ist es notwendig zu definieren, welche ICMP-Nachrichten für den Betrieb des Netzwerkes von Bedeutung sind und welche bereits an der Netzwerksgrenze verworfen werden sollen. Daher muss von den NetzwerkadministratorInnen ein Konzept erarbeitet werden, in dem die notwendigen ICMP-Typen definiert sind.

- Header Manipulation und Fragmentierung

  Die Manipulation des IPv6-Headers und dessen Fragmentierung sind sicherheitskritische Aspekte, die bei der Erstellung einer IPv6-Infrastruktur zu berücksichtigen sind. Um solchen Angriffen entgegenzuwirken, können folgende, im Zuge der Literaturrecherche erarbeiteten, Maßnahmen in Betracht gezogen werden:

  - Verweigerung von IPv6-Fragmenten für Internetworking-Geräte.
  - Sicherstellung angemessener IPv6-Fragmentierungsfilterfunktionen
  - Löschung aller Fragmente mit weniger als 1280 Oktetten (außer dem letzten)

- Layer 3 und Layer 4 Spoofing

  Um einen sicheren Schutz vor Spoofing-Angriffen zu gewährleisten, können kryptografische Schutzmechanismen eingesetzt werden. Hierbei ist besonders darauf zu achten, dass es immer wieder zu der Situation kommt, dass vermeintlich sichere Algorithmen geknackt werden und somit ein augenscheinlich geschütztes System angreifbar machen. Es ist daher auch hier ratsam, dass ein ständiger PDCA-Prozess in das Framework eingebunden wird, um sicherzustellen, dass die verwendeten Verschlüsselungsmethoden noch sicher sind.

  Zusätzlich wird der Einsatz von effizientem Netzwerk-Monitoring empfohlen, um Spoofing innerhalb des Netzwerkes zu erkennen und entsprechend dagegen vorzugehen.

- ARP und DHCP-Attacken

  In sensiblen Umgebungen ist es möglich ARP und DHCP-Attacken durch die Nutzung von „Static Neighbor Entries“ zu unterbinden. Eine weitere Möglichkeit ist der Einsatz von sogenanntem DHCP-Snooping. Mittels DHCP-Snooping lässt sich fehlerhafter oder möglicherweise böswilliger DHCP-Verkehr unterbinden. Darüber hinaus werden Informationen zu Hosts, die eine DHCP-Transaktion erfolgreich abgeschlossen haben, in einer Datenbank mit "Bindungen" gesammelt, die dann von anderen Sicherheits- oder Kontoführungsfunktionen verwendet werden können (Wilkins und Smith 2011).

- Smurf-Attacken (Verstärkung von Broadcasts)

  Zur Verhinderung von Smurf-Attacken und anderen Broadcast-Angriffen, die IoT-Geräte innerhalb eines Netzwerkes dazu bringen eine Flut an Ping-Requests zu senden, ist es möglich, Pakete mit Multicast-Quelladressen an den Grenzen des Netzwerkes herauszufiltern.

- Routing-Attacken

  Obwohl der umfassende Schutz gegen Routing-Attacken primär in den Aufgabenbereich von Internet-Service-Providern fällt, spielt er auch bei Firmen-Netzwerken eine Rolle, sofern diese eine entsprechende Größe aufweisen und lokal voneinander getrennt sind. Aus diesem Grund sind folgende 2 Ansätze zur Vermeidung von Routingangriffen angeführt:

  - Verwendung von IPv6-Hop-Limits zum Schutz von Netzwerkgeräten
  - Verwendung von IPSec zum Sichern von Protokollen wie OSPFv3

5.5 Erstellung des Frameworks

Dieser Abschnitt befasst sich mit der Erstellung des IPv6/IoT-Security-Frameworks. Hierzu werden die im vorherigen Abschnitt angeführten Lösungsansätze gebündelt und priorisiert. Des Weiteren werden zusätzliche Aspekte betrachtet und dem Framework zugefügt, die dem Ziel einer langfristigen und umfassenden Absicherung des Netzwerkes dienlich sind.

- PDCA-Zyklus

Aufgrund der Tatsache, dass einige IoT-Geräte keine automatisierten Softwareupdates unterstützen, diese jedoch für ein sicheres Netzwerk von entscheidender Bedeutung sind, soll ein PDCA-Zyklus zu einem zentralen Bestandteil des Frameworks werden. Bei einem PDCA-Zyklus nach Deming werden Entscheidungen auf Basis des aktuellen Wissensstandes getroffen (Abbildung 21). Diese Änderungen und vor allem deren Auswirkungen bilden die Basis für folgende Entscheidungen (Kreitner 2009).

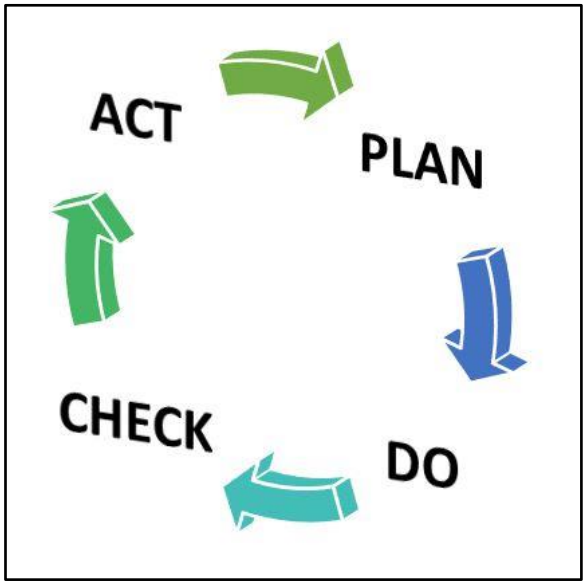


Abbildung 21: PDCA-Zyklus nach Deming. (Kreitner 2009)

Durch das ständige, zyklische Durchlaufen der vier Phasen können sowohl Veränderungen der Umweltfaktoren als auch Änderungen, die sich durch interne Ereignisse, wie zum Beispiel Updates und Modifikationen des Netzwerkes, ergeben, frühzeitig erkannt werden, und sofern diese eine negative Auswirkung auf die Sicherheit mit sich bringen, Maßnahmen zur erneuten Erreichung des Sicherheitszieles gesetzt werden.

Die stufenweise und aufeinander aufbauende Umsetzung der in der bisherigen Arbeit identifizierten Sicherheitsbereiche wird wie folgt determiniert:

1. Schutz vor Identifikation und Aufklärung
2. Schutz vor unautorisierten Zugriffen
3. Absicherung gegen Header-Manipulation und dessen Fragmentierung
4. Schutz vor Spoofing-Attacken
5. Absicherung gegen ARP und DHCP-Attacken
6. Absicherung gegen Broadcast-Angriffe
7. Schutz vor Routing-Attacken

Auf Basis des dargestellten PDCA-Zyklus kann die Integration des Frameworks in folgende 3 Phasen unterteilt werden:

1. Analyse der bestehenden Infrastruktur und deren Komponenten
2. Implementierung und Verbesserung der Netzwerksicherheit
3. Zyklische Überwachung der Netzwerk-Sicherheit und IoT-Sicherheit

Diese drei Prozessschritte bilden den Kern des Frameworks. Dabei beschreibt der erste Prozessschritt die Analyse der bestehenden Infrastruktur und definiert die zur Erreichung des gesetzten Sicherheitsstandards erforderlichen Schritte.

Wurden diese Schritte ausreichend analysiert und beschrieben, werden sie in der zweiten Phase umgesetzt und Schritt für Schritt in die bestehende Infrastruktur integriert. Diese Integration wird bereits von einem PDCA-Zyklus begleitet, da durch jede Änderung der bestehenden Netzwerklandschaft und Konfiguration an den Komponenten neue Risiken und Schwachstellen entstehen können.

In der dritten Phase beginnt die permanente und andauernde Überwachung der Infrastruktur in Bezug auf den Grad ihrer Aktualität und neu bekannt werdenden Gefahren. Werden solche neuen Gefahren bekannt, ist eine Risikoanalyse in Anlehnung an (Norm DIN ISO/IEC 27001:2017-6) durchzuführen, und auf deren Basis die weitere Vorgehensweise zu konkretisieren. Diese permanente Überwachung wird begleitet von einem ständigen Monitoring der Sicherheit der Netzwerkkomponenten und der Infrastruktur. Dadurch kann eine dauerhafte Aufrechterhaltung der Netzwerksicherheit gewährleistet werden.

6 ZUSAMMENFASSUNG

Durch die zu Beginn dieser Arbeit durchgeführte umfassende Recherche konnte aufgezeigt werden, dass mit der Einführung des Internet Protokolls IPv6 neue Risiken auf NutzerInnen und AdministratorInnen zukommen, die es zu berücksichtigen gilt und denen entgegengetreten werden muss. Insbesondere im Bereich von IoT-Geräten, die sich in den kommenden Jahren mehr und mehr in unseren Alltag etablieren werden, zeigt sich, wie bereits in Kapitel 4.6 beschrieben, welche neue Möglichkeiten für AngreiferInnen entstehen, um diese Geräte für sich zu nutzen und großflächige Angriffe durchzuführen.

Durch die Recherche hinsichtlich der Technologien und Standards in den Bereichen IPv6 und IoT-Geräte, war es mit dem erworbenen Wissen bereits in Kapitel 4.6 möglich, die dieser Arbeit zugrunde liegenden Forschungsfrage zu beantworten. Die Erstellung des Frameworks liefert in diesem Zusammenhang keinen Wissenszuwachs, der für die Beantwortung dienlich wäre, wodurch die Beantwortung der Frage, wie in diesem Kapitel dargelegt, als vollständig erachtet werden kann.

Die Arbeitshypothesen, die im Zuge dieser Arbeit zu überprüfen waren, lauteten wie folgt:

H1: Durch den Einsatz von IPv6 auf IoT-Geräten entstehen potentielle Sicherheitslücken, die sich durch Änderung der Konfiguration minimieren lassen.

H0: Die Verwendung von IPv6 auf IoT-Geräten hat keinen sicherheitskritischen Einfluss für EndnutzerInnen.

Wie bereits in Kapitel 4.6 erwähnt, zeigt die Beantwortung der Forschungsfrage, dass durch den Einsatz von IPv6 in Netzwerken, in denen IoT-Geräte betrieben werden, viele Sicherheitsrisiken bestehen bleiben, wie sie bereits in IPv4-Netzwerken bekannt sind. Es entsteht jedoch auch eine beträchtliche Anzahl neuer Angriffsvektoren für außenstehende Personen, die eine andere Herangehensweise an Netzwerk- und IoT-Sicherheit erforderlich macht. Herkömmliche Sicherheitskonzepte reichen daher nicht aus, um einen umfassenden Schutz von IoT-Geräten in IPv6-Netzwerken zu gewährleisten. Zusammenfassend lässt sich durch diese Arbeit die H1-Hypothese bestätigen, dass IPv6 potentiellen AngreiferInnen neue Wege eröffnet, um im Speziellen IoT-Geräte für ihre Zwecke zu missbrauchen. Aus diesem Grund ist es nötig, nicht nur an den IoT-Geräten selbst, sondern explizit auch an den Netzwerkkomponenten, Änderungen der Konfiguration vorzunehmen und diese dauerhaft zu überwachen, um einen effektiven und permanenten Schutz der Infrastruktur zu gewährleisten.

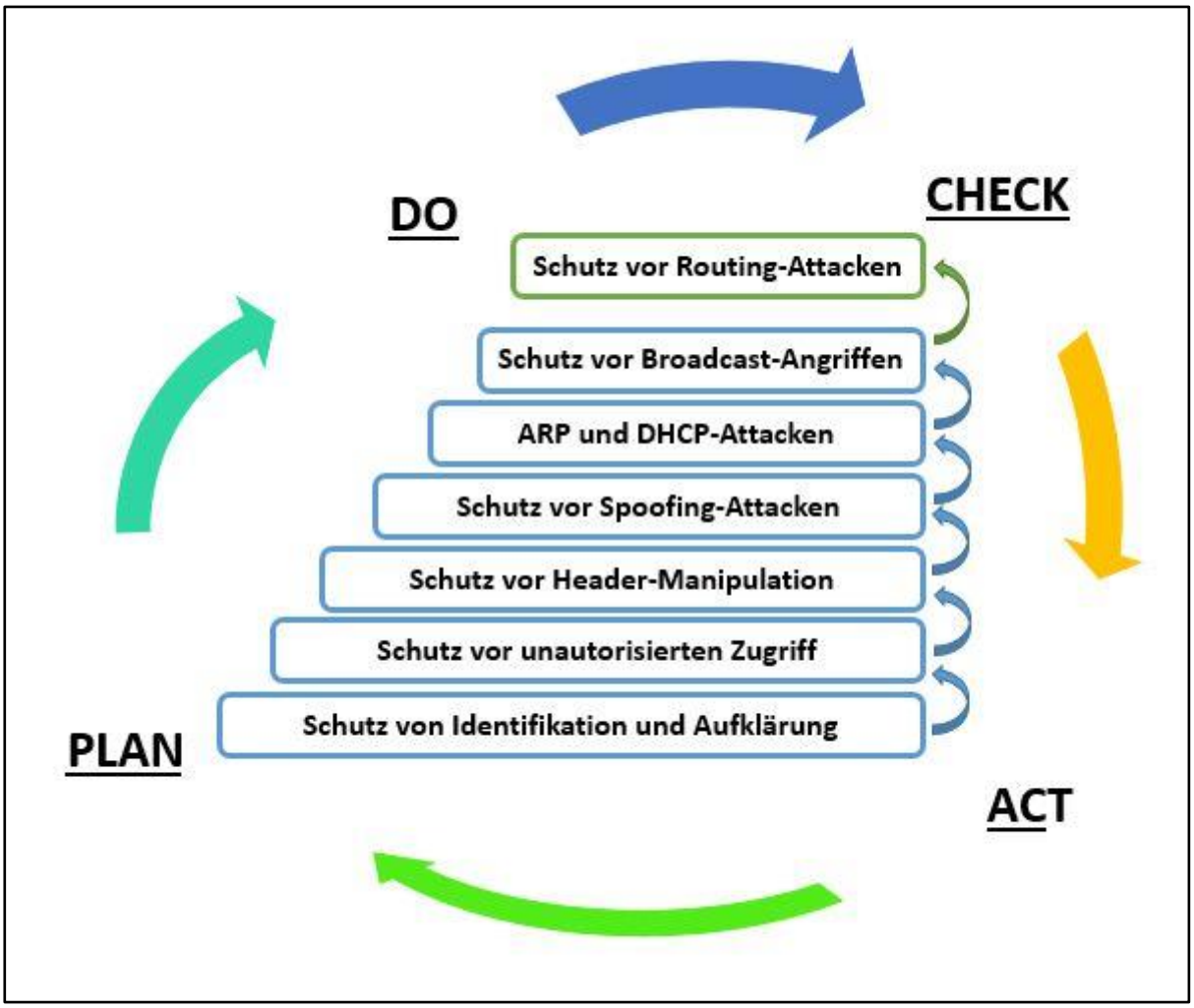
6.1 Resümee

IPv6 bietet Sicherheitsvorteile für diejenigen, die es am besten nutzen können. Diese Stärken können entweder vom verteidigenden oder vom eindringenden Gerät ausgeübt werden. Daher ist die Zeit IPv6 zu ignorieren endgültig vorbei. Die Zeit es zu verstehen, zu erkennen und seine offensichtlichen Vorteile einzusetzen, ist jetzt. Besonders in gemeinsamem Einsatz mit IoT-Geräten offenbart die neue Version des Internetprotokolls seine umfassenden Möglichkeiten.

6.2 Limitierung und weitere Forschung

Das vorliegende Framework stellt in seiner derzeitigen Form die Basis für ein umfassendes und adaptives Führungsmittel zur Erstellung einer IPv6-Netzwerklandschaft dar, die optimal an das Potential aber auch an die Schwachstellen von IoT-Geräten angepasst ist. Da diese Arbeit jedoch rein auf die Erstellung des Frameworks gerichtet ist, ist darauffolgend ein erster Test der Anwendung des Frameworks durchzuführen. Dieser Test soll auch aufzeigen, an welchen Punkten das Framework erweitert und adaptiert werden kann, um die Überführung und langfristige Absicherung von IPv6-Netzwerken noch besser zu bewerkstelligen und AndministratorInnen ein Höchstgrad an Unterstützung bieten zu können. Es ist auch notwendig, den Fokus des Frameworks für eine effektive Nutzung derart zu erweitern, dass nicht nur die sichere Betreibung von IoT-Geräten im Speziellen, sondern jegliche Knoten innerhalb eines IPv6-Netzwerkes umfassend geschützt sind.

ANHANG A - Integrierung des Frameworks



ABKÜRZUNGSVERZEICHNIS

6LoWPAN	IPv6 over Low power Wireless Personal Area Network
AAL	Ambiet Assistet Living
AH	Authentication Header
API	Application Programming Interface
CGA	Cryptographic Generated Addresses
DAD	Duplicate Address Detection
DDoS	Distributed Denial of Service
DHCPv6	Dynamic Host Configuration Protocol for IPv6
DoS	Denial of Service
ESP	Encapsulated Security Payload
EUI	Extended Unique Identifier
ICMP	Internet Communication Messaging Protocol
ICT	Informations- und Kommunikationstechnologien
IDS	Intrusion Dedection System
IERC	European Research Cluster on the Internet of Things
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IoT	Internet of Things
IPSec	Internet Protocol Security
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISAKMP	Internet Security Association und Key Management Protocol
ITU	International Telecommunication Union
LGPLv2	GNU Lesser General Public License
MIT	Massachusetts Institut of Technology
MTU	Maximum Transfer Unit
NA	Neighbor Advertisement
NAT	Network Address Translation
ND	Neighbor Discovery
ND	Neighbor Discovery
NDP	Neighbor Discovery Protocol
NIC	Network Interface Controller
NS	Neighbor Solicitationund
NUD	Neighbor Unreachability Detection
ORCHID	Overlay Routable Cryptographic Hash Identifiers
OSPFv3	Open Shortest Path First version 3
OUI	Organizationally Unique Identifier
PDCA	Do Check Act

PFS	Perfect Forward Secrecy
PKI	Public Key Infrastructure
POSIX	Portable Operating System Interface
RA	Router Advertisement
RAM	Random Access Memory
RD	Router Discovery
RIPng	Routing Information Protocol next generation
RS	Router Solicitation
SA	Security Association
SEND	Secure Neighbor Discovery Protokoll
SLAAC	Stateless Address Autoconfiguration
SPI	Security Parameter Index
VPN	Virtual Private Network

ABBILDUNGSVERZEICHNIS

TABELLENVERZEICHNIS

7 LITERATURVERZEICHNIS

Ashton, Kevin (2009): That "Internet of Things" Thing. In the real world, things matter more than ideas. Hg. v. RFID Journal. Online verfügbar unter http://www.rfidjournal.com/articles/view?4986.

Badach, Anatol; Hoffmann, Erwin (2015): Technik der IP-Netze. Internet-Kommunikation in Theorie und Einsatz. 1. Aufl. s.l.: Carl Hanser Verlag München. Online verfügbar unter http://ebooks.ciando.com/book/index.cfm/bok_id/1864353;B:CIANDO.

Badamchizadeh, Mohammad Ali; Chianeh, Ali Akbari (2006): Security in IPv6. University College of Nabi Akram (UCNA), Iran. Online verfügbar unter http://www.wseas.us/e-library/conferences/2006istanbul/papers/534-390.pdf, zuletzt geprüft am 10.10.2017.

Brandt, Stephan B. (2015): IPv6 for the Internet of Things. Institute of Computer Science.

Doraswamy, Naganand; Harkins, Dan (2003): IPSec. The new security standard for the Internet, intranets, and virtual private networks. 2nd ed. Upper Saddle River, NJ: Prentice Hall PTR (Prentice-Hall PTR Web infrastructure series). Online verfügbar unter http://proquest.tech.safaribooksonline.de/013046189X.

Eclipse Foundation (Hg.) (2017): IoT Developer Survey 2017.

Gartner (07.02.2017): Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016. Rob van der Meulen, rob.vandermeulen@gartner.com. Online verfügbar unter http://www.gartner.com/newsroom/id/3598917, zuletzt geprüft am 02.10.2017.

Hagen, Silvia (2016): IPv6. Grundlagen - Funktionalität - Integration. 3., Auflage. Maur: Sunny Connection.

Halang, Wolfgang A.; Unger, Herwig (Hg.) (2016): Internet der Dinge. Echtzeit 2016 : Fachtagung des gemeinsamen Fachausschusses Echtzeitsysteme von Gesellschaft für Informatik e.V. (GI), VDI /VDE-Gesellschaft für Mess- und Automatisierungstechnik (GMA) und Informationstechnischer Gesellschaft im VDE (ITG) Boppard, 17. und 18. November 2016. Fachtagung Echtzeit; Gesellschaft für Informatik; Gesellschaft Meß- und Automatisierungstechnik; Echtzeit 2016. Berlin, Heidelberg: Springer Vieweg (Informatik aktuell). Online verfügbar unter http://dx.doi.org/10.1007/978-3-662-53443-4.

Heller, Hannes (2009): Frameworks. Aufbau, Design und Einsatz. Fachhochschule Kiel, Kiel. Fachbereich Informatik und Elektrotechnik.

Hogg, Scott; Vyncke, Eric (2009): IPv6 security. Indianapolis, Ind: Cisco Press (Cisco Press networking technology series). Online verfügbar unter http://proquest.tech.safaribooksonline.de/9781587058387.

HP-Inc. (2017): HP Instant Ink - Tinten Lieferservice. Online verfügbar unter https://instantink.hpconnected.com/de/de, zuletzt geprüft am 21.09.2017.

IERC (2016): European Research Cluster on the Internet of Things. Online verfügbar unter http://www.internet-of-things-research.eu/about_ierc.htm, zuletzt geprüft am 20.09.2017.

International Telecommunication Union (2016): SERIES Y: GLOBAL INFORMATION. Next Generation Networks – Frameworks and functional, 06.2016.

IPv6Now (Hg.) (2017): IPv6 Security. IPv6 Security Impact. Online verfügbar unter http://ipv6now.com.au/primers/IPv6SecurityIssues.php, zuletzt geprüft am 26.10.2017.

Norm DIN ISO/IEC 27001:2017-6, 01.06.2017: IT-Sicherheitsverfahren – Informationssicherheits-Managementsysteme – Anforderungen, zuletzt geprüft am 04.03.2018.

Jara, Antonio J.; Ladid, Latif; Skarmeta, Antonio (2014): The Internet of Everything through IPv6: An Analysis of Challenges, Solutions and Opportunities. Online verfügbar unter http://ai2-s2-pdfs.s3.amazonaws.com/962f/e4c4069ac2540368fad56d7a863abdd15a77.pdf.

Kreitner, Robert (2009): Management. 11th ed., U.S. student ed. Mason, OH: South-Western Cengage.

Ladid, Latif; McGibney, Jimmy; Ronan, John (2005): Security with IPv6.

Loshin, Peter (2004): IPv6. Theory, protocol, and practice. 2nd ed. Amsterdam, Boston: Morgan Kaufmann. Online verfügbar unter http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=195293.

Mattern, Friedemann, Flörkemeier, Christian (2010): Vom Internet der Computer zum Internet der Dinge. In: *Informatik Spektrum* 2010 (33 Issue 2), S. 107–121.

Memon, Mukhtiar; Wagner, Stefan Rahr; Pedersen, Christian; Bevi, Femina Hassan Aysha; Hansen, Finn Overgaard (2014): Ambient Assisted Living Healthcare Frameworks, Platforms, Standards, and Quality Attributes. Department of Engineering, Aarhus University, Aarhus 8200, Denmark. Online verfügbar unter http://www.mdpi.com/1424-8220/14/3/4312/htm, zuletzt aktualisiert am 04.03.2014, zuletzt geprüft am 21.09.2017.

Müller, Stefan (2016): Internet of Things (IoT). Ein Wegweiser durch das Internet der Dinge: Books on Demand.

Proenen, Hanns (2016): Securing the IoT and Beyond. IPv6 - a Security Perspective. KuppingerCole. Chiemsee, 12.05.2016.

Rehman, Shafiq Ul; Manickman, Selvakumar (2016): Denial of Service Attack in IPv6 Duplicate Address Detection Process. An Impact Analysis on IPv6 Address Auto-configuration Mechanism. Hg. v. (IJACSA) International Journal of Advanced Computer Science and Applications (7, 6). Online verfügbar unter https://www.researchgate.net/profile/Shafiq_Rehman18/publication/304704640_Denial_of_Service_Attack_in_IPv6_Duplicate_Address_Detection_Process/links/5777d58208ae1b18a7e43dac/Denial-of-Service-Attack-in-IPv6-Duplicate-Address-Detection-Process.pdf.

RFC 2373 (1998): IP Version 6 Addressing Architecture. Hg. v. IETF. Online verfügbar unter https://www.ietf.org/rfc/rfc2373.txt.

RFC 2460 (1998): Internet Protocol, Version 6 (IPv6) Specification. Hg. v. IETF. Online verfügbar unter https://www.ietf.org/rfc/rfc2460.txt.

RFC 2526 (1999): Reserved IPv6 Subnet Anycast Addresses. Hg. v. IETF. Online verfügbar unter https://www.ietf.org/rfc/rfc2526.txt.

RFC 3315 (2003): Dynamic Host Configuration Protocol for IPv6 (DHCPv6). Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc3315, zuletzt geprüft am 28.10.2017.

RFC 3682 (2004): The Generalized TTL Security Mechanism (GTSM). Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc3682.

RFC 3849 (2004): IPv6 Address Prefix Reserved for Documentation. Hg. v. IETF. Online verfügbar unter https://www.ietf.org/rfc/rfc3849.txt.

RFC 3972 (2005): Cryptographically Generated Addresses (CGA). Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc3972.

RFC 4193 (2005): Unique Local IPv6 Unicast Addresses. Hg. v. IETF. Online verfügbar unter https://www.ietf.org/rfc/rfc4193.txt.

RFC 4213 (2005): Basic Transition Mechanisms for IPv6 Hosts and Routers. Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc4213.

RFC 4291 (2006): IP Version 6 Addressing Architecture. Hg. v. IETF. Online verfügbar unter https://www.ietf.org/rfc/rfc4291.txt.

RFC 4302 (2005): IP Authentication Header. Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc4302.

RFC 4303 (2005): IP Encapsulating Security Payload (ESP). Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc4303.

RFC 4443 (2006): Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification. Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc4443.

RFC 4861 (2007): Neighbor Discovery for IP version 6 (IPv6). Unter Mitarbeit von IETF. Online verfügbar unter https://tools.ietf.org/html/rfc4861.

RFC 4884 (2007): Extended ICMP to Support Multi-Part Messages. Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc4884, zuletzt geprüft am 28.10.2017.

RFC 4919 (2007): IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs). Overview, Assumptions, Problem Statement, and Goals. Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc4919.

RFC 6419 (2011): Current Practices for Multiple-Interface Hosts. Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc6419, zuletzt geprüft am 28.10.2017.

RFC 7296 (2014): Internet Key Exchange Protocol Version 2 (IKEv2). Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc7296.

RFC 7707 (2016): Network Reconnaissance in IPv6 Networks. Hg. v. IETF. Online verfügbar unter https://tools.ietf.org/html/rfc7707, zuletzt geprüft am 28.10.2017.

Romdhani, Imed; Abdmeziem, Raid; Tandjaoui, Djamel (2015): Architecting the Internet of Things: State of the Art. ResearchGate. Online verfügbar unter https://www.researchgate.net/publication/274718805_Architecting_the_Internet_of_Things_State_of_the_Art, zuletzt geprüft am 17.09.2017.

Satapathy, Suresh Chandra; Joshi, Amit; Modi, Nilesh; Pathak, Nisarg (Hg.) (2016): Proceedings of International Conference on ICT for Sustainable Development. ICT4SD 2015 Volume 2. 1st ed. 2016. Singapore, s.l.: Springer Singapore (Advances in Intelligent Systems and Computing, 409). Online verfügbar unter http://dx.doi.org/10.1007/978-981-10-0135-2.

Soltanian, Mohammad Reza Khalifeh; Amiri, Iraj Sadegh (2015): Theoretical and experimental methods for defending against DDoS attacks. Waltham, MA: Syngress is an imprint of Elsevier.

Wilkins, Sean; Smith, Franklin H. (2011): CCNP security secure 642-637 official cert guide. Indianapolis, Ind: Cisco Press. Online verfügbar unter http://proquest.tech.safaribooksonline.de/9780132378581.

Ziegler, Sébastien; Crettaz, Cedric; Ladid, Latif; Krco, Srdjan; Pokric, Boris; Skarmeta, Antonio F. et al. (2013): IoT6 – Moving to an IPv6-Based Future IoT. In: David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell et al. (Hg.): The

Future Internet, Bd. 7858. Berlin, Heidelberg: Springer Berlin Heidelberg (Lecture Notes in Computer Science), S. 161–172.