Hochschule für Angewandte Wissenschaften Hamburg
*Hamburg University of Applied Sciences*

**Anwendungen 1**
**WiSe 2010 / 2011**

Heiner Perrey
Betreuer: Prof. Dr. Dirk Westhoff

# Angriffe auf Funknetzwerke

*Fakultät Technik und Informatik*
*Department Informatik*

*Faculty of Engineering and Computer Science*
*Department of Computer Science*

# Inhaltsverzeichnis


**1 Einleitung** **3**

1.1 Motivation und Problemstellung . . . . . 3

1.2 Motivation: SKIMS . . . . . 4

**2 Konzept des Masterprojekts** **5**

2.1 Grundlagen . . . . . 5

2.1.1 Merkle's Puzzle . . . . . 5

2.1.2 Bluetooth Low Energy (BLE) . . . . . 6

2.2 Ziel des Masterprojekts . . . . . 7

2.2.1 Wissenschaftliche Relevanz . . . . . 8

2.2.2 Aktuelle Untersuchungen . . . . . 8

2.2.3 Probleme und offene Fragen . . . . . 11

**3 Zusammenfassung** **12**

**Literaturverzeichnis** **13**

# 1 Einleitung

***Hinweis:*** *Aufgrund neuer Erkenntnisse enthält diese Arbeit größere Änderungen zum Vortrag. Im Wesentlichen wurden die Abschnitte 2.1.2 und 2.2 überarbeitet.*

**Kurzzusammenfassung**

In dieser Arbeit wird ein Konzept für einen zusätzlichen Sicherheitsmechanismus in Bluetooth Low Energy (BLE) durch Merkle's Puzzle vorgestellt. Durch die Verteilung der (Merkle) Puzzle über einen unsicheren Kanal kann ein geheimer Schlüssel ausgetauscht und für eine gewisse Zeitspanne vor einem passiven Lauscher geheim gehalten werden. Merkle's Puzzle eignet sich insbesondere für Geräte mit stark asymmetrischen Kapazitäten. Hierbei kann die Rechenlast fast vollständig von dem stärkeren Gerät übernommen werden. BLE wurde vor allem für energiebeschränkte Geräte entwickelt. Durch den Verzicht auf ECDH bietet BLE keinen Schutz vor einem passiven Lauscher.

## 1.1 Motivation und Problemstellung

Bluetooth Low Energy (BLE) ist als neue energiesparende Technologie bereits auf dem Markt verfügbar (TIOnChip (2010)). Es wurde für Geräte entwickelt, die nicht genug Leistung für gängige Funkstandards haben (WLAN, Bluetooth Classic etc.). Deshalb sind viele dieser Geräte auch nicht in der Lage, komplexe Schlüsselaustauschverfahren wie Elliptic Curve Diffie-Hellman (ECDH) zu nutzen. So wird bei BLE vorerst auf den Schutz vor einem passiven Angreifer[1] während der Kopplungsphase verzichtet. Dieses kann Geräte, die BLE nutzen, vor ein ernstes Sicherheitsproblem stellen.

Merkle's Puzzle ist eines der ersten Verfahren, um einen Schlüssel über einen unsicheren Kanal auszutauschen. So stellte es die Weichen für heutige Schlüsselaustauschverfahren (MPInt (2002)). Es stellt aber auch andere Merkmale, die es aus heutiger Sicht zu einem sehr reizvollen Verfahren machen. Sind unterschiedlich rechenstarke Geräte beteiligt, kann die Rechenlast fast vollständig zu dem stärkeren Gerät verlagert werden. Somit können Schlüssel auch an sehr leistungsschwache Geräte verteilt werden, die sonst gänzlich auf einen Sicherheitsmechanismus verzichten müssten. Des Weiteren ist es möglich, an beliebig viele Geräte parallel einen Schlüssel auszuteilen.

Merkle's Puzzle bietet die Möglichkeit, BLE um den Schutz vor einem passiven Lauscher zu ergänzen. Um weiterhin auf die geringe Rechenleistung eingehen zu können, muss der Mehraufwand möglichst gering gehalten werden. Hierzu soll Merkle's Puzzle idealerweise direkt in die BLE-Spezifikation aufgenommen werden, um die bestehenden Strukturen von BLE ausnutzen zu können. In dieser Arbeit werden zum einen einige Ansätze präsentiert, zum anderen werden weitere Problem- und Fragestellungen für das fortlaufende Masterprojekt dargelegt.

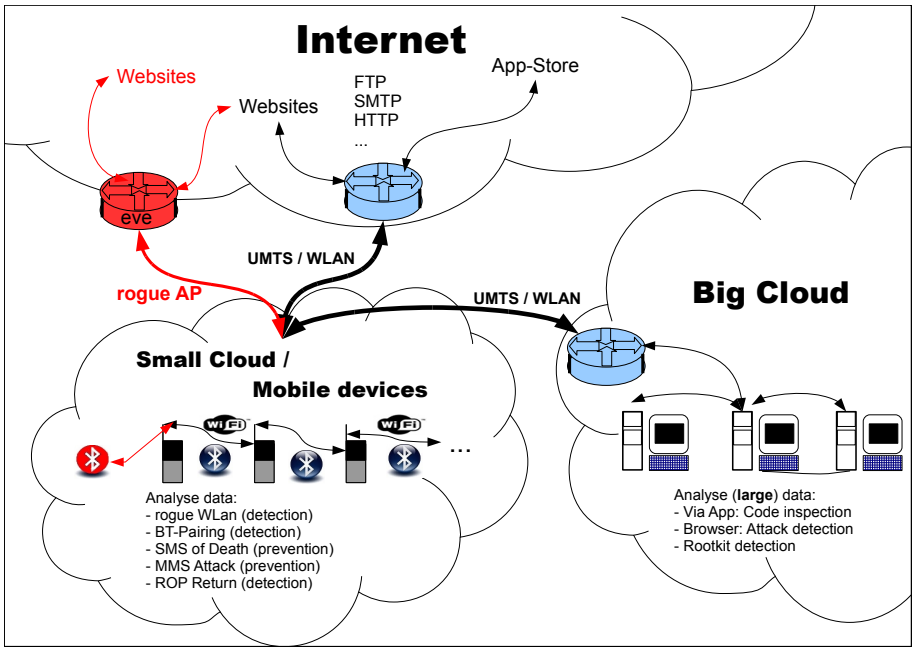


Abbildung 1.1: Abstrakte (vorläufige) Übersicht der Komponenten in SKIMS

## 1.2 Motivation: SKIMS

SKIMS steht für „**S**chichtenübergreifendes **K**ooperatives **I**mmunsystem für mobile **M**ehrseitige **S**icherheit“. Hierbei handelt es sich um ein vom Bundesministerium für Bildung und Forschung (BMBF) finanziertes Projekt, in dem ein kooperatives *Distributed Intrusion Detection System* (DIDS) für mobile Endgeräte entwickelt wird. Um dies zu bewerkstelligen, arbeitet die HAW Hamburg mit verschiedenen Partnern zusammen. Die Funktion des IDS wird in eine *große* und eine *kleine Wolke* unterteilt (siehe Abb. 1.1). In die große (Internet) Wolke können große Mengen an Daten hochgeladen und analysiert werden, um Angriffe und Schad-Code zu erkennen. Die kleine Wolke besteht aus mobilen Endgeräten (z.B. Smartphones), welche sich untereinander in Funk-Reichweite befinden. Auf jedem Smartphone in der kleinen Wolke kann die IDS-Applikation laufen, so dass sich das entsprechende Gerät an der Überwachung seiner Umgebung beteiligen kann.

Zu diesem Zeitpunkt stehen einige Konzepte zur Diskussion. Ein Ansatz beschäftigt sich damit, jedem Teilnehmer einen Anreiz zu bieten, um an der Kooperation mitzuwirken. Ähnlich wie von Buttyán und Hubaux (2001) vorgestellt könnte ein Micro-Payment für die Teilnahme geboten werden, damit genug Geräte mitwirken. Um die Micro-Payments ausschließlich an legitime Teilnehmer zu überweisen, müsste sich jedes Gerät gegenüber einer Zahlungsinstanz authentifizieren können, um einen Nachweis über den investierten Rechenaufwand zu erbringen.

Damit dies möglich ist, könnte ein Ticket-System genutzt werden. Jedes Gerät, das sich an dem IDS beteiligen will, bekommt ein Ticket zugewiesen, mit dem es nachweisen kann, dass es zu einem bestimmten Zeitpunkt an einem bestimmten Ort gewesen ist. Somit ist zwar noch nicht sichergestellt, dass dieses Gerät tatsächlich CPU-Aufwand investiert hat, aber es muss sich geographisch innerhalb des Immunsystems befunden haben. Hierzu ist es von Interesse, mehrere Empfänger parallel zu erreichen, ohne dabei iterativ über eine Liste zu traversieren. Wie im Kapitel 2.2.3 noch näher erläutert wird, kann Merkle’s Puzzle genutzt werden, um ein solches Ticket-System umzusetzen.

[1] Eine Definition des passiven Angreifers erfolgt in Abschnitt 2.2.2

# 2 Konzept des Masterprojekts

In diesem Kapitel wird das grundlegende Konzept des Masterprojekts vorgestellt. Hierzu wird zunächst auf einige Grundlagen zu Merkle's Puzzle und Bluetooth Low Energy (BLE) eingegangen. Danach wird beschrieben, wie das Konzept in die bestehenden Arbeiten passt und wie sich BLE und Merkle's Puzzle miteinander kombinieren lassen.

## 2.1 Grundlagen

### 2.1.1 Merkle's Puzzle

Merkle's Puzzle wurde 1974 von Ralph Merkle vorgestellt und schließlich 1978 veröffentlicht (HPMerkle (2011)). Es beschreibt das erste Verfahren, um einen Schlüssel über einen unsicheren Kanal sicher auszutauschen.[1] So nahmen W. Diffie und M. Hellman in ihrem Paper (Diffie und Hellman (1976)) Bezug auf Merkle's Idee. Das Verfahren ist im Gegensatz zum Diffie-Hellman-Verfahren, in Bezug auf den Rechenaufwand der beteiligten Kommunikationspartner asymmetrisch. Das bedeutet, dass es möglich ist, die Rechenlast fast vollständig auf das leistungsstärkere Gerät auszulagern, so dass es auch für sehr energiebeschränkte bzw. leistungsschwache Geräte, wie beispielsweise Wireless Sensor Networks (WSN), oder anderen in Bereichen Anwendung finden kann (Armknecht und Westhoff (2009)). Dies gelingt durch die Wahl der Sicherheitsparameter, die in Merkle's Puzzle genutzt werden. Somit ist außerdem die Gesamtlast für die Geräte anpassbar, so dass je nach Szenario und verwendeter Geräte der Aufwand reduziert werden kann (Merkle (1978)). Wie in diesem Abschnitt noch erläutert wird, verringert sich hierdurch ggf. das erreichte Sicherheitsniveau. Im Folgenden werden die Funktionsweise für den Sender und den Empfänger nach Merkle (1978) vorgestellt. Da Merkle's Puzzle nur im Zusammenhang mit einem *full functional device* (FFD) und einem oder mehreren *reduced functional devices* (RFD) sinnvoll ist, wird ähnlich argumentiert wie in Armknecht und Westhoff (2009).

**Sender** Das FFD erzeugt eine Menge aus $n$ Puzzle. Ein Puzzle ist wie folgt aufgebaut:

$$P = E_{(k_{weak})}\{P_{ID}, k_{strong}\} \tag{2.1}$$

$E$ stellt hierbei eine kryptographische Funktion dar, die mit dem Schlüssel $k_{weak}$ verschlüsselt. Der $k_{weak}$ dient der Sicherung der Puzzleübertragung und kann beliebig schwach gewählt werden. Wichtig ist, dass das RFD einen $k_{weak}$ in angemessener Zeit „knacken" kann. Der $k_{strong}$ soll später für die sichere Kommunikation genutzt werden und ist somit der auszutauschende Schlüssel. Der Puzzle Identifier $P_{ID}$ macht ein Puzzle eindeutig identifizierbar, darf aber keinen Rückschluss auf die Reihenfolge, in der die Puzzle erstellt oder versendet wurden, bieten. Er sollte zufällig erstellt werden. Zusätzlich wird jedem Puzzle vor der Verschlüsselung mit $k_{weak}$ eine Markierung, z.B. erkennbare Padding-Bits, hinzugefügt.[2] Alle $n$ Puzzle werden nun über das unsichere Medium an das RFD übertragen.

[1] Die Sicherheit des Verfahrens wird in Abschnitt 2.2.2 diskutiert

[2] In Armknecht und Westhoff (2009) wird eine Technik vorgestellt, mit der die Anzahl der Padding-Bits stark reduziert werden kann.

**Empfänger** Das RFD empfängt alle Puzzle und speichert diese.[3] Nun sucht es sich zufällig eines heraus und führt eine Brute-Force-Attacke durch. Anhand der Markierung erkennt das RFD, sobald ein ausgesuchter $k_{weak}$ korrekt war. Es ist nun im Besitz des $k_{strong}$ und der entsprechenden $P_{ID}$. Die $P_{ID}$ sendet es in Klartext über den unsicheren Kanal zurück an das FFD. Das FFD kann die $P_{ID}$ dem richtigen $k_{strong}$ zuordnen, mit dem sie fortlaufend ihre Kommunikation sichern.

**Der passive Lauscher** Da ein passiver Angreifer $Eve$ unter optimalen Bedingungen lauscht, besitzt sie nach abgeschlossener Kommunikation sämtliche Puzzle und die ID des ausgewählten Puzzles. Da die $P_{ID}$ keinen Rückschluss auf die Reihenfolge bietet, besteht ihre einzige Möglichkeit darin, durch wahlloses Knacken der Puzzle nach der $P_{ID}$ zu suchen. Im Schnitt wird Eve $\frac{n}{2}$ Puzzle lösen müssen, um das richtige zu erraten. Ihr Mehraufwand wird also durch den Faktor $n$, die Anzahl der gesendeten Puzzle, bestimmt. Der Aufwand des FFD und RFD ist linear. Eve hingegen hat einen quadratischen Aufwand. Außerdem ist sie im Gegensatz zum RFD gezwungen, alle Pakete zu empfangen und zu speichern, was unter Umständen einen Mehraufwand bedeutet.

Somit wird deutlich, dass Merkle's Puzzle nur eine zeitlich begrenzte Sicherheit bietet, so dass der Schlüssel ggf. häufig erneuert werden muss. Hierbei steht Eves Rechenkapazität in Relation zur Sicherheit. Lädt Eve die Puzzle in eine Cloud zur Analyse hoch, kann ggf. von keiner Sicherheit mehr ausgegangen werden. Der Anwendungsfall und das Angreifermodell bestimmen somit die Relevanz des Verfahrens. In dieser Arbeit wird davon ausgegangen, dass Eve höchstens bereit ist, minimalen bis moderaten Aufwand in den Angriff zu investieren. Näheres hierzu befindet sich in Abschnitt 2.2.2.

**Unterschiedliche Rechenkapazität** Um mit nahezu beliebig geringer Rechenleistung des Empfängergerätes umgehen zu können, müssen die Sicherheitsparameter angepasst werden. Je nachdem wie stark das RFD ist, kann der $k_{weak}$ so parametrisiert werden, dass er von dem RFD in angemessener Zeit knackbar ist. Der Schlüssel kann theoretisch beliebig schwach gewählt werden. Ist das RFD nicht in der Lage, eine Brute-Force-Attacke durchzuführen, kann der $k_{weak}$ von dem FFD auf null gesetzt werden, was einem Senden der Puzzle in Klartext gleicht. Um einen Lauscher dennoch für eine gewisse Zeit abzuwenden, muss das FFD dementsprechend mehr Puzzle senden. Eine Bewertung hierzu befindet sich in Abschnitt 2.2.3.

### 2.1.2 Bluetooth Low Energy (BLE)

Bluetooth Low Energy wurde ursprünglich von Nokia unter dem Namen „Wibree" entwickelt. Heute ergänzt es die Bluetooth (BT) Spezifikation in der Version 4.0 (Wibree (2007), BTSpec (2010)). Eines der Hauptmerkmale von BLE ist der im Vergleich zu BT Classic relativ geringe Energieverbrauch. Somit wurde BLE insbesondere für dementsprechend leistungsschwache Geräte entwickelt. Auf den Einsatz von Secure Simple Pairing (SSP) wurde verzichtet. SSP nutzt zum Austausch der Schlüssel ECDH. Durch SSP kann ein Schlüssel trotz Anwesenheit eines passiven Lauschers sicher ausgetauscht werden. Zukünftige Versionen werden allerdings den Einsatz von ECDH unterstützen (BTSpec (2010)). BLE ist schon in einige Systeme integriert. Texas Instruments vertreibt eine On-Chip-Lösung (TIOnChip (2010)). Die Firma Casio entwickelte den Prototyp einer Uhr, die BLE nutzt, um sie z.B. mit einem Smartphone kommunizieren zu lassen (BLEWatch (2011)).

BLE hat eine maximale Übertragungsrate von 1 Mbit/s. BT Classic bietet hier je nach Geräteklasse 1-3 Mbit/s (BTRate (2011)). Die meisten mobilen Geräte nutzen derzeit BT Classic in der Version 2.0+EDR (Enhanced Data Rate) oder höher und haben somit eine Übertragungsrate von ca. 2 Mbit/s. Dies

[3] Um den Aufwand des Empfanges zu reduzieren, kann das RFD auch zu einem definierten Zeitpunkt ein Puzzle mitschneiden (Synchronisation mit dem FFD nötig).

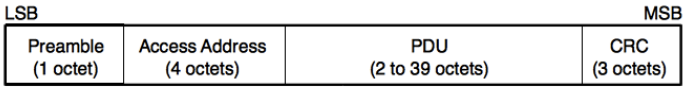


Abbildung 2.1: Link Layer packet format (BTSpec (2010)).

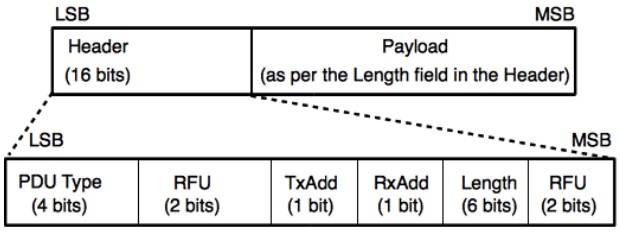


Abbildung 2.2: Advertising channel PDU und Header (BTSpec (2010)).

entspricht also ca. der doppelten Übertragungsrate von BLE. Die Reichweite von BLE ist allerdings nicht geringer als die von BT Classic. BT Classic hat eine Reichweite zwischen 1 und 100 Meter (BTRange (2011)), BLE sogar mehr (BLERange (2011)). Des Weiteren spart BLE Energie, indem weitere Anpassungen, wie ein effizienterer *idle*-Modus, gemacht wurden. Kompatibel sind BLE und BT Classic nur durch die Nutzung eines Dual-Mode-Radio. Es ist also neue Hardware nötig, um BLE zu nutzen (BLEfaq (2009)).

**Advertisement Events** Ein neues Feature von BLE sind die *Advertisements* (BTSpec (2010)). Diese bieten eine Möglichkeit, um Nachrichten bzw. *Protocol Data Units* (PDU) über einen Broadcastkanal zu versenden. Sie sind vorgesehen, um nach neuen Geräten zu scannen, eine Verbindung zu einem Gerät aufzubauen oder Nutzerdaten zu versenden. Abbildung 2.2 zeigt den wesentlichen Aufbau einer solchen PDU. Der Header besteht aus einem *PDU Type*, *TxAdd*, *RxAdd* Feld und Bereichen für die zukünftige Nutzung (*reserved for future use* RFU). Während TxAdd und RxAdd konkrete Informationen über den PDU Typ liefern (BTSpec (2010)), spezifiziert das Feld PDU Type die Art der Nachricht. Hierbei gibt es vier verschiedene Typen des Advertisings:

- ADV_IND: connectable undirected advertising event
- ADV_DIRECT_IND: connectable directed advertising event
- ADV_NONCONN_IND: non-connectable undirected advertising event
- ADV_SCAN_IND: scannable undirected advertising event

Für diese Arbeit ist vorerst der *ADV_NONCONN_IND* interessant. Hierbei können Advertisements ohne den Aufbau einer Verbindung über einen Broadcastkanal gesendet werden. Die Payload für ein Advertisement liegt zwischen 0 und 31 Bytes für die Daten und 6 Byte für die Adresse des Advertisers. Somit können Inhalte bis zu 248 Bit versendet werden.[4] Eine PDU ist Teil eines Link Layer Pakets, welches in Abbildung 2.1 dargestellt ist. Je nachdem wie groß die Payload einer PDU ist, kann ein Link Layer Paket bis zu 376 Bit groß sein.

## 2.2 Ziel des Masterprojekts

Ziel des Masterprojekts ist ein konzeptioneller Zusammenschluss von BLE und Merkle's Puzzle. So sollen die in Abschnitt 2.2.1 angesprochenen Aspekte miteinander verbunden werden. Hierbei soll

[4] Zur Analyse des Transports der Puzzle über Advertisements siehe Abschnitt 2.2.2.

Merkle's Puzzle nach Möglichkeit in die bestehenden Konzepte der BLE-Spezifikation eingeflochten werden. Es wird geprüft, welche bestehenden Konzepte der BLE-Spezifikation sich eignen, um Merkle's Puzzle umzusetzen. Des Weiteren werden Anwendungsfälle erstellt, in denen dieses Konzept praktisch sinnvoll ist. Es wird ein Angreifermodell definiert, anhand dessen die Anwendungsfälle bewertet werden. Einige zu diesem Zeitpunkt bekannte Probleme, die im Laufe des Projekts geklärt werden sollen, aber auch Lösungsansätze sind im Folgenden aufgeführt. Zunächst wird auf die wissenschaftliche Relevanz eingegangen.

### 2.2.1 Wissenschaftliche Relevanz

BLE bietet bislang keinen Schutz vor einem passiven Angreifer. Auf SSP wurde verzichtet und auch andere Mechanismen, um SSP zu ersetzen, wurden noch nicht eingeführt. Zukünftige Versionen sollen allerdings ECDH unterstützen und werden somit Schutz vor einem Lauscher bieten. Da BLE gerade in sehr schwachen Geräten, wie z.B. Wireless Senser Networks (WSN), Biosensoren und Health-Monitoring einsetzbar sein soll, sind die meisten Verfahren, unter anderem ECDH, zu rechenintensiv (Armknecht und Westhoff (2009)). Durch die steigende Nutzung von Sensoren in verschiedenen Bereichen wie Gesundheit, Umwelt oder im Haushalt kann angenommen werden, dass BLE schnell an Bedeutung gewinnen wird und dass ein Schutz vor einem passiven Lauscher somit von immer größerem Interesse sein wird. Hierbei kann Merkle's Puzzle auf die Bedingungen der Geräte mit asymmetrischen Kapazitäten angepasst werden (siehe Abschnitt 2.1.1).

In Armknecht und Westhoff (2009) wird der Schlüsselaustausch durch Merkle's Puzzle im Sensorenbereich vorgestellt. Die Autoren beziehen sich hierbei auf den Standard IEEE 802.15.4. Bei dem Einsatz von BLE könnte die höhere Übertragungsrate nicht nur das Verfahren optimieren, sondern es könnten auch neue Anwendungsfälle spezifiziert werden. Hierzu kommen insbesondere Anwendungen mit Smartphones in Betracht. Durch das stetig ansteigende Interesse an Smartphones werden auch hier Sicherheitsmechanismen immer bedeutender (Ogus u. a. (2011)). Auch die zukünftige Ausstattung von Smartphones mit BLE ist anzunehmen.

Die Sicherheit von Bluetooth Classic ist seit dem Einsatz von SSP relativ hoch, bietet aber immer noch Möglichkeiten für einen Angriff (Hypponen und Haataja (2007)). Die Autoren in Ogus u. a. (2011) fordern zusätzlich ein schärferes Bewusstsein des Nutzers, um die Sicherheit von BT Classic zu erhöhen. Durch den fehlenden Schutz vor einem passiven Lauscher in BLE wäre somit noch mehr Vorsicht von Seiten des Nutzers erforderlich. Um die Verantwortung mehr von dem Benutzer zum Gerät zu verlagern, ist ein aufbauendes Konzept nötig. Hierfür werden einige Grundlagen im nächsten Abschnitt (2.2.3) vorgestellt.

### 2.2.2 Aktuelle Untersuchungen

**Anwendungsfall 1** Ein Anwendungsfall bezieht sich auf das in Abschnitt 1.2 beschriebene Szenario eines Ticket-Systems für eine DIDS. Hierfür ist wichtig, dass mehrere Teilnehmer möglichst effizient mit einem Ticket ausgestattet werden können. Ein $k_{strong}$ aus einem Puzzle könnte als Ticket dienen. Zum Austausch würde ein Ticket-Service die Puzzle versenden.

Außerdem muss dieser Anwendungsfall konkret auf die Anforderungen von SKIMS abgestimmt werden. Da es, wie später in diesem Abschnitt noch deutlich wird, mehrere Stunden dauern kann, bis ein geheimer Schlüssel ausgetauscht ist, muss hierbei die Flexibilität des SKIMS-Netzwerks berücksichtigt werden. Ordnen sich die Endgeräte in der kleinen Wolke (siehe Abb. 1.1) zu schnell neu, so ist dieses Verfahren in der zur Zeit gedachten Form ggf. ungeeignet.

**Anwendungsfall 2** In einem zweiten Anwendungsfall wird davon ausgegangen, dass zwei oder mehrere Geräte sich zwar in geographischer Nähe zueinander befinden, aber kein pre-shared secret möglich ist. Ein Schlüssel soll also ad-hoc ausgetauscht werden. Diese Anforderung ist zum Beispiel für einen automatisierten Prozess denkbar. Die Geräte sind relativ leistungsschwach (z.B. Smartphones oder Sensoren). Die Geheimhaltungsdauer der ausgetauschten Daten ist zeitlich begrenzt. Vorstellbar wäre ein mit vielen kleinen Sensoren ausgestatteter Raum. Alle Sensoren sollen regelmäßig und automatisch mit einem Schlüssel versehen werden. Die Sensoren sind wegen ihrer beschränkten Kapazitäten nicht in der Lage, komplexe Schlüsselaustauschverfahren zu nutzen. Es könnte ein Smartphone oder Notebook genutzt werden, um die Schlüssel zu erzeugen und sie an die Sensoren gemäß Merkle's Puzzle zu verteilen.

**Angreifermodell** Das Angreifermodell geht von einem passiven Lauscher (Eve) aus, der unter optimalen Bedingungen lauscht, also jedes Paket mitlesen kann. Passiv heißt hierbei, dass nicht in die Kommunikation von dem FFD und RFD eingegriffen wird. Der Angreifer kann insofern aktiv werden, als dass er versucht, aktiv den $k_{weak}$ oder $k_{strong}$ zu erraten. Eve ist bereit, minimalen bis moderaten Aufwand zu betreiben. Das bedeutet, dass sie zum Beispiel ein Notebook (2-4 GHz) besitzt, mit dem sie einen Angriff ausführt. Sie ist also um Größenordnungen von 500 bis 2000 mal rechenstärker als das RFD. Sie ist nicht in der Lage, ein verteiltes System oder eine Cloud zu nutzen, um das System zu untergraben. Der zeitliche Aufwand, den Eve bereit ist zu investieren bzw. die Zeitspanne, in der die Daten geheim bleiben müssen, variiert von Anwendungsfall zu Anwendungsfall.

**Anpassung der BLE-Spezifikation** Im Moment ist das *Pairing* in BLE für den Schlüsselaustausch zuständig. Die verschiedenen Pairing-Varianten sind im Security Manager (SM) definiert. Vor dem Pairing wird eine Link Layer (LL) Verbindung aufgebaut, da die Informationen zum Schlüsselaustausch auf einem anderen Kanal stattfinden, als zum Beispiel Informationen zum Scannen nach Geräten (BTSpec (2010)). Merkle's Puzzle kann hierbei als ein neues *Association Model*, wie zum Beispiel *Passkey Entry*, gesehen werden. Bei Passkey Entry wird auf einem Gerät eine PIN erzeugt, die vom Nutzer in ein zweites Gerät eingegeben wird. Wenn Merkle's Puzzle genutzt wird, erzeugt das FFD die Schlüssel bzw. Puzzle und sendet diese an alle Geräte, die sich in Reichweite befinden und empfangsbereit sind. Die Logik von Merkle's Puzzle sollte ebenfalls im SM definiert werden. Aber auch die restlichen Schichten des BLE-Controllers müssen entsprechend ergänzt werden, um Merkle's Puzzle von Haus aus zu unterstützen.

Als Verschlüsselungsfunktion bietet sich zur Zeit RC5 an (Rivest (1995)). RC5 hat eine variable Schlüsselgröße und unterstützt somit auch beispielsweise den hier ausgewählten Schlüssel von 17 Bit. Das von BLE AES-128 hat den Nachteil, dass die Ausgabe mit zuvielen Bits aufgefüllt wird. Um die Puzzle möglichst effizient zu verteilen, sollen 32 Bit als Blockgröße von RC5 gewählt werden. Nur so können mehrere Puzzle in eine PDU verstaut werden, da sonst die Ausgabe der Verschlüsselungsfunktion erweitert wird und somit nicht mehr in die Payload der PDU passt. Da BLE zur Zeit nur AES-128 unterstützt, muss hier eine weitere Anpassung am SM vorgenommen werden, um RC5 anzubieten.

**Performanz-Analyse** Um die Nutzbarkeit von Merkle's Puzzle und BLE einzuordnen, wird an dieser Stelle eine Analyse durchgeführt. Es wird ein ähnliches Szenario wie in Armknecht und Westhoff (2009) angenommen. Mobile Biosensoren im Gesundheitsbereich sollen über Nacht mit einem Schlüssel versehen werden. Die Übertragungsrate ist BLE spezifisch und liegt somit bei 1 Mbit/s. Ein Puzzle soll über den Advertisement Mechanismus von BLE versant werden. Jedes Puzzle wird mit einem $k_{weak}$ von 17 Bit verschlüsselt. Es wird von einem FFD als Sender und einem RFD als Empfänger mit beschränkten Kapazitäten ausgegangen. Das RFD ist ein Microcontroller mit 4 MHz. Das FFD ist nur durch die Übertragungsrate beschränkt. Um das Verfahren zu brechen, soll Eve

| Anzahl Pakete | Aufwand FFD | Aufwand Eve | Aufwand RFD |
|---|---|---|---|
| 17747146 | 1.58 h | 7 Tage | 17.04 s |
| 35494291 | 3.16 h | 14 Tage | 17.04 s |
| 70988582 | 6.31 h | 28 Tage | 17.04 s |

Tabelle 2.1: Aufwand für das FFD, RFD und Eve bei Verdoppelung der gesendeten Pakete mit zwei Puzzle pro PDU.

einen Aufwand von sieben Tagen haben. Sie ist ungefähr $500$ mal leistungsstärker als das RFD. Das RFD braucht $0.26$ ms, um einen möglichen Schlüssel zu probieren und $17.04$ s, um ein Puzzle zu knacken (Armknecht und Westhoff (2009)). Somit benötigt Eve $\frac{0.26}{500} = 0.00052$ ms und schließlich $0.00052 \cdot \frac{2^{17}}{2} = 34.07872$ ms, um ein Puzzle zu knacken. Damit Eve im Schnitt sieben Tage benötigt, um das richtige Puzzle zu finden, muss die folgende Anzahl an Puzzle versendet werden:

$$M = \frac{604800000 \text{ ms}}{34.07872 \text{ ms}} \cdot 2 \approx 35494291 \tag{2.2}$$

Durch die Multiplikation mit $2$ wird berücksichtigt, dass Eve im Schnitt nur die Hälfte der gesendeten Puzzle knacken muss.

Um zu untersuchen, wie hoch der Zeitaufwand $t_{FFD}$ des FFD ist, muss zunächst die Puzzlegröße bestimmt werden. In jede PDU passen maximal $31 \text{ Byte} \cdot 8 = 248$ Bit (Abb. 2.1). Um alle Puzzle indizieren zu können, sind $log_2(35494291) = 25.08$ Bit nötig. Es müssen also mindestens $26$ Bit für die $P_{ID}$ reserviert werden. Zusätzlich sind Padding-Bits nötig (Merkle (1978)). Ein Ansatz, diese auf einem Minimum zu halten, befindet sich in Armknecht und Westhoff (2009). Hier werden ebenfalls $12$ Bit zum Padding als ausreichend angenommen. Somit wäre es möglich, $k_{strong}$ der Größe $248 - 26 - 12 = 210$ Bit zu senden.

Die Größe des $k_{strong}$ muss allerdings abhängig von der Zeit, die Eve benötigt, um an den Schlüssel zu gelangen, gewählt werden. Braucht Eve länger den $k_{strong}$ als $\frac{n}{2}$ Puzzle zu knacken, bringt dies keine zusätzliche Sicherheit, da sie im Schnitt nach $\frac{n}{2}$ Puzzle im Besitz des $k_{strong}$ ist. In dem hier vorgestellten Fall muss der $k_{strong}$ lediglich $42$ Bit umfassen, um Schutz für etwas länger als sieben Tage zu bieten. Mit $k_{strong} = 41$ Bit benötigt Eve knapp unter einer Woche. Somit würde es sich erst ab $42$ Bit für Eve lohnen, die Puzzle anstelle des $k_{strong}$ anzugreifen.

Hieraus ergibt sich schließlich die Größe für jedes Puzzle: $26 + 12 + 42 = 80$ Bit. Die kleinste Blockgröße des RC5 ist 32. Somit muss jedes Puzzle auf $96$ Bit gepadded werden. Es passen also zwei Puzzle in eine PDU. Da die Puzzle nicht miteinander in Verbindung stehen und das RFD wartet bis alle Puzzle gesendet wurden, bevor es die $P_{ID}$ zurücksendet, ist das Senden mehrerer Puzzle pro PDU kein Sicherheitsrisiko (Perrey u. a. (2011)).

Das FFD muss $\frac{35494291}{2}$ Pakete, je $320$ Bit ($128$ Bit + $192$ Bit Payload) versenden. Daraus ergibt sich folgender Zeitaufwand, $t_{FFD}$, für das FFD:

$$t_{FFD} = \frac{17747146 \cdot 320 \text{ Bit}}{1000000 \frac{\text{Bit}}{\text{s}}} \approx 5679.1 \text{ s} \approx 1.58 \text{ h} \tag{2.3}$$

Das FFD hat also einen Aufwand von $1.58$ Stunden. Der Aufwand des RFD ist durch die Anzahl der Puzzle nicht betroffen, solange er nicht alle Puzzle empfängt und speichert. Tabelle 2.1 illustriert den ungleichen Mehraufwand bei Verdoppelung der Puzzleanzahl von Eve im Gegensatz zu dem FFD und RFD.

### 2.2.3 Probleme und offene Fragen

Sollen mehrere Empfänger gleichzeitig erreicht werden, zum Beispiel für das Ticketsystem einer DIDS, könnten Skalierungsprobleme auftreten. Schickt das FFD die $n$ Puzzle an $x$ Empfänger, so bekommt sie auch $x$ Puzzle-Identifier zurück. Die Wahrscheinlichkeit, dass Eve ein Puzzle findet, in dem einer dieser $P_{ID}$ enthalten ist, steigt an. Um dies auszugleichen, muss das FFD mehr Puzzle verschicken. Der Aufwand steigt daher auch für das FFD.

Die Funktionsweise des Merkle's Puzzle hängt maßgeblich davon ab, ob eine Mindestmenge an Puzzle gesendet werden kann. Die konkrete Anzahl mag variieren, wird aber immer relativ groß sein. Da das Verteilen der Puzzle eine gewisse Zeit in Anspruch nehmen wird, limitiert die Übertragungsrate von BLE die Nutzbarkeit in einigen Szenarios. Je nach Anwendungsfall kann es Vorgaben über die Größe dieses Zeitfensters geben. Beschränkende Faktoren könnten zum Beispiel die Akku-Laufzeit des Sender-Geräts sein.

Wie in Abschnitt 2.1.1 aufgeführt, kann Merkle's Puzzle auf nahezu beliebig leistungsschwache Geräte angepasst werden. Dies ist allerdings kritisch zu betrachten, da die Last für das FFD hierdurch beliebig stark ansteigen kann. Wird auf den $k_{weak}$ verzichtet, kann nicht mehr von einem Verschlüsselungsverfahren gesprochen werden. Es müsste der Begriff *Steganographie* eingeführt werden. Zusätzlich ist fraglich, ob ein Gerät, das keinen Brute-Force-Angriff durchführen kann, einen $k_{strong}$ in einer kryptographischen Funktion verwenden kann, um die Kommunikation zu verschlüsseln. Es bleibt kritisch zu betrachten, für welchen Anwendungszweck der Mehraufwand für das FFD noch in Relation zum Nutzen steht (Armknecht und Westhoff (2009)). Hierzu soll eine untere Grenze für die Größe von $k_{weak}$ definiert werden.

Eine ganz entscheidene Rolle in diesem Zusammenhang kann auch die asymmetrische Entwickling der Rechenleistung der Endgeräte in Relation zu der Übertragungsrate der Funkstandards spielen. Während die Rechenleistung in der Regel stetig ansteigt, stagniert die maximale Übertragungsrate der Funkstandards häufig über längere Zeit. Da eine Reaktion auf mehr Rechenleistung (des Angreifers) durch das Senden von mehr Puzzle kompensiert werden kann, besteht die Gefahr eines zu großen Ungleichgewichts. Es ist irgendwann nicht mehr sinnvoll, mehr und mehr Puzzle zu senden, da schließlich noch Daten ausgetauscht werden müssen. Hierzu wird ein Angreifermodell erstellt, welches eine untere und obere Schranke für Eve darstellt.[5]

Jeder Anwendungsfall hat unterschiedliche Anforderungen, so dass die hier beschriebenen Probleme sich unterschiedlich stark niederschlagen. Dazu soll in diesem Masterprojekt zwischen den unterschiedlichen Anwendungsfällen, dem Angreifermodell und dem so erreichten Sicherheitsniveau differenziert werden. Es werden verschiedene Anwendungsfälle definiert, in denen „zusätzliche Sicherheit in BLE durch Merkle's Puzzle" einen Vorteil erzielt. So werden die Grenzen zwischen dem theoretischen Konzept und der praktischen Umsetzung gezogen. Genauer betrachtet wird hierbei auch das Verhältnis aus Performanz und Sicherheit.

[5] Die Grundlagen hierzu werden in Armknecht und Westhoff (2009) vorgestellt.

# 3 Zusammenfassung

BLE stellt eine neue Funktechnologie, die auf extrem energiebeschränkten Geräten nutzbar ist. Hierbei wurde auf das aus BT Classic bekannte SSP-Verfahren bzw. ECDH verzichtet. BLE ist somit nicht vor einem passiven Lauscher während der Kopplungsphase sicher. Obwohl in zukünftige Versionen der BLE Spezifikation ECDH nachgerüstet wird, sind viele Geräte aufgrund ihrer beschränkten Kapazitäten dennoch nicht in der Lage, komplexe Schlüsselaustauschverfahren zu nutzen.

Merkle's Puzzle bietet eine zeitlich begrenzte Sicherheit für den Schlüsselaustausch über einen unsicheren Kanal. Durch geschicktes Setzen der Parameter ist es hierbei möglich, das Sicherheitslevel an die genutzten Geräte anzupassen. Insbesondere asymmetrische Kapazitäten der Geräte können durch Verschieben der Rechenlast nahezu komplett kompensiert werden. Hierbei übernimmt das FFD die Rechenlast zu Gunsten des RFD, welches mit sehr wenig zusätzlichem Aufwand belastet wird.

Ziel dieses Masterprojekts ist der konzeptionelle Zusammenschluss dieser beiden Komponenten. Merkle's Puzzle soll den fehlenden Schutz vor einem passiven Lauscher in BLE stellen und besonders in Szenarios mit asymmetrischen Geräteklassen den Schlüsselaustauch überhaupt möglich machen. Hierdurch kann die Sicherheit der Kommunikation von der Nutzungsweise der Geräte bis zu einem gewissen Grad entkoppelt werden. Die wesentliche Problemstellung dieser Arbeit umfasst zum einen den konzeptionelle Zusammenschluss und zum anderen eine kritische Analyse über Anwendungsfälle und Angreifermodell.

# Literaturverzeichnis


[Armknecht und Westhoff 2009] ARMKNECHT, F. ; WESTHOFF, D.: Using Merkle's Puzzle for Key agreement with Low-end Devices. In: *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on* (2009), Dezember, S. 858–864

[BLEfaq 2009] *BLUETOOTH LOW ENERGY TECHNOLOGY FAQ*. bluetooth.com. December 2009. – [online] http://www.bluetooth.com/SiteCollectionDocuments/Low_Energy_FAQ_External_General_Public.pdf Abruf: 12.01.2011

[BLERange 2011] *Bluetooth Low Energy Technology - Technical Info*. bluetooth.com. 2011. – [online] http://www.bluetooth.com/English/Products/pages/bluetooth_low_energy_technology__technical_info.aspx Abruf: 03.02.2011

[BLEWatch 2011] *Casio Bluetooth Watch Puts Your Phone's Info on Your Wrist*. gizmodo.com. January 2011. – [online] http://gizmodo.com/5725608/casio-bluetooth-low-energy-watch-prototype-has-2-year-battery-life Abruf: 02.02.2011

[BTLinkCtr 2011] *Bluetooth Baseband*. bluetooth.com. 2011. – [online] http://www.bluetooth.com/English/Technology/Works/Pages/Architecture__Baseband.aspx Abruf: 12.01.2011

[BTRange 2011] *Get Technical: Basics*. bluetooth.com. 2011. – [online] http://www.bluetooth.com/English/Technology/Pages/Basics.aspx#5 Abruf: 12.01.2011

[BTRate 2011] *Bluetooth Radio*. bluetooth.com. 2011. – [online] http://www.bluetooth.com/English/Technology/Works/Pages/Architecture__Radio.aspx Abruf: 12.01.2011

[BTSpec 2010] : *BLUETOOTH SPECIFICATION Version 4.0*. Document - Bluetooth SIG. Juni 2010. – URL https://www.bluetooth.org

[Buttyán und Hubaux 2001] BUTTYÁN, L. ; HUBAUX, J.-P.: Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks. In: *ACM/Kluwer Mobile Networks and Applications (MONET)* 8 (2001), S. 579–592

[Diffie und Hellman 1976] DIFFIE, W. ; HELLMAN, M.: New Directions in Cryptography. In: *Information Theory, IEEE Transactions on* (1976), November, S. 644–654

[HPMerkle 2011] *Homepage of Ralph C. Merkle*. Webseite. 2011. – [online] http://www.merkle.com/ Abruf: 12.01.2011

[Hypponen und Haataja 2007] HYPPONEN, K. ; HAATAJA, K.M.J.: "Niño" Man-In-The-Middle Attack on Bluetooth Secure Simple Pairing. In: *The 3rd IEEE/IFIP ICI 2007*, September 2007, S. 1–5

[KnktNokia 2011] : *Kontakt zu Mitarbeiter von Nokia*. Januar 2011. – URL www.nokia.com

[Merkle 1978] MERKLE, R.: Secure Communications Over Insecure Channels. In: *Communications of the ACM* (1978), April, S. 294–299

[MPInt 2002] *Secure Communications over Insecure Channels (1974) - With an Interview from the year 1995*. itas.fzk.de - Edited by Arnd Weber. January 2002. – [online] http://www.itas.fzk.de/mahp/weber/merkle.htm Abruf: 09.01.2011

[Ogus u. a. 2011] OGUS, O. ; PERREY, H. ; RAJASEKARAN, H. ; WESTHOFF, D.: Why we can not (yet) trust our smartphones - how co-operation could help. In: *under submission to SECRYPT* (2011), Februar

[Perrey u. a. 2011] PERREY, H. ; OGUS, O. ; WESTHOFF, D.: Poster: Security Enhancement for Bluetooth Low Energy with Merkle's Puzzle. In: *under submission to WiSec'11* (2011), Februar

[Rivest 1995] RIVEST, R.: The RC5 Encryption Algorithm. In: *Proceedings of the 1994 Leuven Workshop on Fast Software Encryption (Springer)* (1995), S. 86–96

[TIOnChip 2010] : *2.4-GHz Bluetooth® low energy System-on-Chip*. Document. October 2010. – URL http://focus.ti.com/lit/ds/symlink/cc2540.pdf

[Wibree 2007] *Wibree becomes ULP Bluetooth*. electronicsweekly.com. June 2007. – [online] http://www.electronicsweekly.com/Articles/2007/06/12/41582/Wibree-becomes-ULP-Bluetooth.htm Abruf: 09.01.2011