Karlsruher Institut für Technologie

KASTEL

Forschungsstand der Sicherheit von implantierbaren medizinischen Geräten

Bachelorarbeit von

Marc Palkowitsch-Amberger

an der Fakultät für Informatik
KASTEL – Institut für Informationssicherheit und Verlässlichkeit

Erstgutachter:	Prof. Dr. rer. nat. Jörn Müller-Quade
Zweitgutachter:	Prof. Dr. Thorsten Strufe
Betreuender Mitarbeiter:	M.Sc. Valerie Fetzer

13. November 2022 – 13. März 2023

Karlsruher Institut für Technologie
Fakultät für Informatik
Postfach 6980
76128 Karlsruhe

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Änderungen entnommen wurde.

Bretten, 13.03.2023

. . . . . .

(Marc Palkowitsch-Amberger)

Zusammenfassung

Millionen Menschen weltweit sind auf programmierbare medizinische Implantate (IMDs, engl. implantable medical devices) angewiesen, um am Leben zu bleiben oder ihre Lebensqualität zu verbessern. Viele dieser implantierten Geräte kommunizieren auch heute noch unverschlüsselt mit der Außenwelt und bieten so eine große Angriffsfläche. Auch wenn bis heute keine konkreten Angriffe auf IMDs bekannt geworden sind, häufen sich Angriffe auf die IT von Krankenhäusern.
Ab den 2000er Jahren wurde an der Absicherung von implantierbarer Medizintechnik gegen Angriffe von außen geforscht. Hierbei wurden Sicherheitsprotokolle entwickelt, die physische Körpersignale zur Schlüsselerzeugung verwenden, Schutzmechanismen gegen Angriffe auf die Batterieladung eines Implantats entwickelt, Anomalieerkennungsverfahren zur Erkennung von Angriffen auf implantierte Geräte entwickelt und externe Geräte als Proxy gegenüber Kommunikationspartnern des Implantats eingesetzt.
Da IMDs nur über begrenzte Energie, Prozessorleistung und Speicher verfügen stellt die Entwicklung von kryptographischen Protokollen für diese eine große Herausforderung dar. Ein Problem der Security bei IMDs stellt der medizinische Notfall dar. Sobald dieser eintritt muss das medizinische Personal sofort auf diesen zugreifen können, was beim Einsatz von Verschlüsselung die Utility einschränken könnte und auch in den entwickelten Protokollen oftmals durch Verzicht auf starke Authentifikation und/oder Verschlüsselung gelöst wird.
Wir geben in dieser Arbeit einen Überblick über die wissenschaftlichen Arbeiten bzgl. der Security von IMDs. Wir betrachten hierbei nur die Bedrohungslage direkt beim Patienten - falls ein IMD die Möglichkeit der Anbindung an das Internet hat wird dies hier nicht berücksichtigt, da dies den Rahmen dieser Arbeit sprengen würde.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einführung

Seit Menschengedenken hat der Mensch versucht durch die Entwicklung von Technologie seinen Aufenthalt auf diesem Planeten zu verbessern. So hat sich seit Anfang des 18. Jahrhunderts vor der Industriellen Revolution die Lebenserwartung des Menschen durch den Einsatz von Technologie wie bessere Behausungen und Heizung, aber auch durch die Entwicklung von Medizinprodukten wie Medikamenten und medizinischen Geräte kontinuierlich erhöht.
So wurde bereits 1958 der erste Herzschrittmacher implantiert, der nur eine Batterielaufzeit von einer Woche hatte [Lar+03]. Dies war das erste entwickelte implantierbare medizinische Gerät (IMD), dem bis heute viele weitere folgen sollten wie z.B. implantierbare Defibrillatoren, Insulinpumpen und Neurostimulatoren. In der Anfangszeit dieser großartigen technischen Entwicklungen, die insbesondere durch die Entwicklung von integrierten Schaltkreisen und Computern stark beschleunigt wurden lag der Fokus der Entwicklung natürlich auf der korrekten und sicheren Funktion der Geräte. So stand auch bei der Entwicklung des Internets, das anfangs nur von den Universitäten genutzt wurde, die Entwicklung und Umsetzung neuer Ideen und damit die korrekte Funktion im Vordergrund. Aber wie bei der Entwicklung des Internets, bei der 1988 der erste Computerwurm die IT-Systeme unsicher machte [ER89], erkannten ab den 2000er Jahren auch die Forscher der Medizintechnik die Notwendigkeit implantierbare Medizintechnik gegen Angriffe von außen absichern zu müssen [BJH16].
Bereits erfolgte Angriffe auf Medizinprodukte zeigen, dass dies eine ernstzunehmende Bedrohung ist. So starben am 29. September 1982 im Großraum Chicago sieben Menschen nach der Einnahme des mit Zyanid verseuchten Schmerzmittels Tylenol, das daraufhin vom Markt genommen wurde. Dies führte zur Entwicklung von manipulationssicheren Siegeln für Medikamente. 2008 postete jemand auf der Seite der Epilepsievereinigung der USA blinkende Bilder, vermutlich um Anfälle auszulösen [Mil08], 2019 wurden tausende blinkende Bilder und Videos an die Twitter-Follower der Vereinigung geschickt [Fer19].
Sicherheitslücken in medizinischen Geräten können auch ohne erfolgreichen Angriff negative Folgen für die Hersteller haben. So fanden im Jahr 2016 Security-Spezialisten der Firma MedSec Holdings gravierende Sicherheitslücken in Herzschrittmachern der Firma St. Jude Medical [16]. Anstatt St. Jude Medical über diese Erkenntnisse zu informieren, veröffentlichte MedSec Holdings zusammen mit der Investmentfirma Muddy Waters einen Bericht [16], woraufhin die Aktien von St. Jude Medical um 10 Prozent fielen. Im Nachhinein gaben Muddy Waters und MedSec Holdings zu, dass sie eine finanzielle Vereinbarung getroffen hatten, um vom Fall des Aktienkurses von St. Jude Medical zu profitieren [Mar18].
Die Anzahl der Sicherheitslücken bei Medizinprodukten hat in den letzten Jahren zugenommen [CIS17] [CIS18] [CIS21]. Dennoch besitzen die meisten IMDs keine oder

unzureichende Sicherheitsvorkehrungen und die Hersteller handeln nach dem Prinzip Security by Obscurity [Gup12].
In dieser Arbeit wird der Forschungsstand der Security von IMDs in Bezug auf die Bedrohungslage beim Patienten dargestellt - falls ein IMD die Möglichkeit der Anbindung an das Internet hat wird dies hier nicht berücksichtigt, da dies den Rahmen dieser Arbeit sprengen würde.

1.1 IMDs

Implantierbare medizinische Geräte (Implantable Medical Devices) werden definiert als elektronische Systeme, die in den menschlichen Körper eingebaut werden und kontinuierlich den Gesundheitszustand des Patienten überwachen, das Auftreten bestimmter medizinischer Bedingungen erkennen und vorhersagen und bei Bedarf eine Therapie einleiten [AY16]. Die IMDs werden chirurgisch in den Körper des Patienten einoperiert, um ein bestehendes medizinisches Problem des Patienten zu behandeln. Ein IMD besteht aus einer Batterie, seiner Elektronik (Hardware) und Elektroden, die entweder über Leitungen im Körper des Patienten mit dem IMD verbunden sind oder als eigenständige Geräte über Funk kommunizieren und mit dem IMD ein Body Area Network (BAN) bilden.
Die Hardware eines IMD Systems besteht nach [Rus+14] aus

1. Analoger Schnittstelle mit Signalverarbeitung für das Erfassen von Sensordaten und Auslösen von Signalen;

2. Arbeits- und Datenspeicher zum Erfassen von persönlichen Gesundheitsinformationen und gemessenen Daten;

3. Mikroprozessor, zur Ausführung von gerätespezifischer Software;

4. Funkschnittstelle zur Übertragung von Daten zwischen dem IMD und einem Programmer oder einem anderen Sensor/Aktor am Patienten;

Nach der Implantation kann ein IMD nur über seine analoge Schnittstelle und seine Funkschnittstelle mit der Außenwelt interagieren. Über die Funkschnittstelle wird das IMD durch einen externen **Programmer** mit Hilfe eines Funkprotokolls von außen eingestellt. Dieses verwendet normalerweise das Medical Implant Communication Service (MICS) Frequenzband, welches von der FCC (Frequenzverwaltungsbehörde der USA, vergleichbar mit der Bundesnetzagentur) im Bereich von 401 MHz bis 406 MHz (Med-Radio) für medizinische Geräte reserviert wurde [IY16]. Nach [Rus+14] ermöglicht das MICS-Band eine gute Signalausbreitung durch den menschlichen Körper ohne andere Geräte zu stören. Außerdem erlaubt es im Gegensatz zu früheren IMDs einen größeren Abstand von bis zu 5 Metern [IY16] zwischen Patient und externem Programmer, bei denen z.B. ein Herzschrittmacher auf 175 kHz sendete, was einen Höchstabstand von 5 cm vorgab [KI09]. Die Funkprotokolle der Kommunikation zwischen IMD und Programmer sind alle proprietär und werden von den Herstellern nicht veröffentlicht, so dass Informationen über diese nur durch Hacker an die Öffentlichkeit gelangen.

Falls mit dem IMD noch andere Geräte wie z.B. Sensoren implantiert werden, die nicht physikalisch mit dem IMD verbunden sind, findet die Kommunikation mit diesen Geräten durch ein entsprechendes Funkprotokoll statt. Einige Programmer können über das Internet geupdatet werden, manche IMDs können Live-Telemetriedaten an eine lokale Basisstation senden, die mit dem Internet verbunden ist. Diese beiden Schnittstellen werden wir aufgrund der Menge an Literatur darüber nicht betrachten, da dies den Rahmen dieser Arbeit überschreiten würde.
Um die Häufigkeit von invasiven Operationen zum Austausch erschöpfter Implantate zu minimieren fordert man von IMDs Ressourcenbeschränktheit, d.h. eine geringe Größe, ein geringes Gewicht, einen niedrigen Peak-Energieverbrauch und eine geringe Prozessorauslastung über die Lebensdauer des Geräts. Dies schließt eine energiesparende Funkkommunikation mit ein.
Nach [SE09] ist ein IMD für eine durchschnittliche Batterielebensdauer von 90 Monaten bei einer Batteriekapazität von 0,5 A h bis 2 A h ausgelegt.

1.2 BANs (body area networks)

Wir definieren in Anlehnung an [Rus+14] ein Body Area Networks (BAN) als Funknetzwerk, in dem ein IMD mit im menschlichen Körper implantierten medizinischen Sensoren und Aktoren kommuniziert. Dabei dienen Sensoren zur Messung von physiologischen Signalen und Aktoren zur Beeinflussung von physiologischen Signalen durch Aussenden elektrischer Impulse. Wir schließen auch hier die Betrachtung von Funkprotokollen für die Kommunikation mit einem externen (oft mit dem Internet verbundenen) Netzwerk aufgrund der diese Arbeit übersteigenden Menge an Literatur aus.
In der klassischen Darstellung von BANs besteht dieses aus Sensoren, Aktoren und einem als „Senke“ bezeichneten externen Gerät, das über mehr Rechenleistung als alle anderen Teilnehmer des BANs verfügt. Die Verbindung der Geräte eines BANs bilden eine Netzwerktopologie, die oft als Baum mit der Senke als Wurzel oder als Graph mit der Senke als Hauptknoten dargestellt wird. Alle Geräte in einem BAN außer diesem Hauptknoten kommunizieren nur mit anderen Geräten des eigenen BANs.
In unserem Fall übernimmt immer das IMD die Rolle der Senke des BANs.

1.3 Typen von IMDs

Bei IMDs unterscheidet man zwischen einzelnen Implantaten, deren Sensoren und Aktoren mit dem Gerät verbunden sind und Implantaten, die über Funk mit Sensoren/Aktoren kommunizieren und zusammen ein BAN bilden. Ein typisches Beispiel für ein Einzelimplantat ist der Herzschrittmacher, während für den zweiten Typ implantierte Insulinpumpen als typischer Vertreter gelten. Im folgenden werden einige Implantattypen vorgestellt:
Implantat am Herzen: Unter die sog. CIEDs (engl. cardiac implantable electronic

devices) fallen Herzschrittmacher und implantierbare Defibrillatoren (sog. ICDs (engl. implantable cardioverter defibrillator)). Diese werden in der Nähe des Brustkorbs unter die Haut implantiert und überwachen die elektrische Aktivität des Herzens. Falls notwendig geben sie elektrische Impulse ab, um das Herz zum regelmäßigen Pumpen zu bringen [Nuñ18]. Ein ICD verfügt zusätzlich zum Herzschrittmacher Elektroden für die Defibrillation, die beim Auftreten eines Kammerflimmerns einen Schock abgeben [DD03].

implantierte Insulinpumpe: Ein Insulinpumpe mit externem Port zum Auffüllen des Insulinspeichers wird zur Behandlung von Diabetes implantiert. Zu einem Insulinpumpensystem gehören neben der Pumpe eine externe Fernbedienung mit Anzeige und ein implantierter Glukosesensor, der permanent Glukosemesswerte an die Pumpe und die Fernbedienung sendet. Die Pumpe injiziert automatisch Basaldosen, zur Gabe von hohen Insulindosen (z.B. nach einer reichhaltigen Mahlzeit) muss der Patient diese per Fernbedienung auslösen.

Tiefe Hirnstimulation (THS): Die tiefe Hirnstimulation ist eine in der funktionellen Neurochirurgie eingesetztes IMD, das eine kontinuierliche elektrische Stimulation durch implantierte Elektroden ermöglicht [Ben03]. Die Elektroden werden in einem bestimmten Hirnbereich implantiert und erzeugen elektrische Impulse, um Zittern, Muskelsteifheit und andere Hirnstörungen zu reduzieren. Die THS wird zur Therapie von Patienten mit Bewegungsstörungen, Parkinson, essentiellem Tremor, Dystonie, Tourette-Syndrom, Depression und Zwangsstörungen eingesetzt [Hag18].

Rückenmarkstimulation (SCS - spinal chord stimulation): Ein Rückenmarksstimulator [Hag18] sendet elektrische Impulse an die Schmerzwahrnehmungsbahn im Rückenmark und unterbricht die Nervenimpulse die an das Gehirn gesendet werden und ersetzt sie durch ein Kribbeln. Mit einer Fernbedienung kann der Stimulator gesteuert werden, um die Stimulation an die Schmerzstärke und die täglichen Aktivitäten anzupassen.

Cochlea-Implantat: Ein Cochlea-Implantat [Bow22] ist ein kleines elektronisches Gerät, das den Cochlea-Nerv elektrisch stimuliert. Das Cochlea-Implantat besteht aus zwei Teilen (der äußere Teil befindet sich außerhalb des Ohrs, der innere Teil wird unter die Haut hinter dem Ohr implantiert). Der äußere Teil verarbeitet den Schall und leitet ihn an den inneren Teil des Implantats weiter. Der innere Teil sendet Signale an den Cochlea-Nerv, der Schallsignale an das Gehirn weiterleitet, um ein Hörgefühl zu erzeugen. Dieses IMD wird eingesetzt, um das Gehör von Patienten wiederherzustellen, die an Hörverlust oder Taubheit leiden.

Implantat zur Blasenstimulation: Ein Blasenstimulator sendet elektrische Signale an die Blase, um z.B. einer überaktiven Blase, Stressharninkontinenz oder Harnverhalt entgegenzuwirken. Dieses IMD besteht aus einem äußeren Teil (Controller, Transmitter) und einen implantierbaren Mikrostimulator (Implantat und Elektrode).

Peronaeus-Stimulator: Ein Peronaeus-Stimulator [Hau+00] ist über ein Kabel mit einer mehrpoligen Nervenmanschettenelektrode verbunden, die am Peroneusnerv des Knies implantiert wird. Dieses Implantat wird von einer externen Steuereinheit mit Strom versorgt und gesteuert und ermöglicht die selektive Aktivierung der Dorsalflex-Muskulatur zur Wiederherstellung einer verlorenen Knöchelfunktion.

Magenschrittmacher: Ein Magenschrittmachersystem besteht aus dem Magenstimulator, einer Elektrode und einem externen Programmer. Er wird im Allgemeinen zur

Kontrolle der motorischen Fehlfunktion des Magens eingesetzt, um schwere Gastroparese [Abe+03] und schwere Adipositas [Shi+09] zu behandeln.
implantierbarer Funkdrucksensor: Ein implantierbarer Funkdrucksensor [Tan+17] wird in der Nähe des Organs, des Nervs oder des Gewebes implantiert, das überwacht werden soll. Durch seinen Einsatz können Infektionen durch Katheter vermieden werden. Beispiele für Funkdrucksensoren sind Überwachung des intraokularen Drucks zur Behandlung der Glaukomkrankheit [Chi+13], Blasendruck-Überwachung [Tan+17] oder Herzdrucküberwachung zur Früherkennung einer möglichen Herzinsuffizienz [Rio17].

1.4 Security und Datenschutz für IMDs

Entwurfsziele beim Design von IMDs sind zuallererst Betriebssicherheit (engl. safety) und Nützlichkeit/Funktionsumfang. Wir werden im Folgenden für Nützlichkeit den englischen Begriff Utility verwenden. Safety impliziert, dass das IMD mehr nützt als es Schaden beim Patienten anrichtet und Utility bedeutet, dass die verfügbare Funktionalität einen Mehrwert für Patienten und medizinisches Personal darstellt [Hal+08b]. Die Safety eines IMD garantiert Datenzugriff für autorisierte Personen, korrekte Datenerfassung und -speicherung im IMD und Konfigurierbarkeit für autorisierte Personen. Zur Utility tragen die Möglichkeit zum Softwareupdate bei Fehlern, die Erfassung eines Audit-Logs und eine ressourcenschonende Programmierung des IMDs bei.
Neben diesen Eigenschaften ist es wichtig, einen IMD vor Angriffen zu schützen, was unter dem Begriff der Security fällt. Hierbei muss darauf geachtet werden, dass die Betriebssicherheit eines IMD niemals auf Kosten der Security eingeschränkt wird. Im Rahmen der Security muss ein IMD auch die klassischen Schutzziele der Informationssicherheit Vertraulichkeit (engl. confidentiality), Verfügbarkeit (engl. availability) und Integrität (engl. integrity) erfüllen. Hierbei steht **Vertraulichkeit** dafür, dass nur berechtigte (autorisierte) Personen Zugriff (durch Authentifikation) auf die Informationen des IMD erhalten, **Integrität** bedeutet, dass die Informationen bei der Kommunikation mit dem IMD sowie auf dem IMD gespeicherte Informationen nicht verändert wurden und **Verfügbarkeit** charakterisiert, dass berechtigte Personen innerhalb eines vereinbarten Zeitraums Zugriff auf die Informationen auf dem IMD erhalten [WMD16].
Ein weiteres Entwurfsziel für IMDs ist der Datenschutz. Hierbei bestehen folgende Datenschutzziele für IMDs/BANs (Quellen: [Hal+08b] [Rus+14]):

- **Datenschutz bzgl. IMD-Existenz:** Unbefugte sollten nicht in der Lage sein festzustellen, dass ein Patient ein IMD/BAN trägt.
- **Datenschutz bzgl. IMD-Typ:** Unbefugte sollten den Typ des vom Patienten getragenen IMD/BANs nicht ermitteln können.
- **Geheimhaltung der Geräte-ID:** Unbefugte sollten nicht in der Lage sein, die eindeutige ID eines IMDs bzw. BAN-Sensors zu bestimmen.

- **Datenschutz von Aufzeichnungsdaten:** Unbefugte sollten nicht auf gespeicherte Daten über den Patienten zugreifen können.

- **IMD-Träger Datenschutz:** Unbefugte sollten nicht in der Lage sein, Eigenschaften des IMDs bzw. BANs zur Identifizierung des Patienten zu verwenden.

- **Schutz vor Tracking:** Unbefugte sollten nicht in der Lage sein mit vom IMD/BAN abgegebenen Signalen einen Patienten zu verfolgen oder zu lokalisieren ([Rus+14] erwähnt hier die Verfolgunge mittels eines Funk-Fingerabdrucks).

- **Telemetrie Datenschutz:** Unbefugte sollten nicht in der Lage sein, vom IMD/BAN ausgesendete (private) Telemetriedaten zu ermitteln.

1.5 Angreifermodell und -ziele

In Anlehnung an [Hal+08b] und [Rus+14] können Angreifer nach ihren Zielen, Fähigkeiten und Rollen im Bezug auf den IMD-tragenden Patienten klassifiziert werden.
Klassischerweise unterscheidet man zwischen aktiven und passiven Angreifern. Der passive Angreifer kann jegliche mit Messinstrumenten erfassbare Signale des IMD/BANs empfangen, was Seitenkanäle mit einschließt. Der aktive Angreifer kann darüber hinaus alle gesendeten Nachrichten verändern, jammen und eigene Nachrichten senden.
Angreifer können Einzelentitäten sein oder eine Gruppe bilden. Außerdem unterscheidet man noch, ob ein Angreifer mit auf dem Markt frei verfügbaren Geräten oder mit Spezialhardware angreift.
Prinzipiell betrachtet man auch noch die Rolle des Angreifers als Insider oder Outsider des medizinischen Systems. Wir gehen aber davon aus, dass ein Außenstehender beim Angriff eine Rolle im System (wie z.B. Mitarbeiter des Krankenhauses) vortäuschen kann (z.B. durch Diebstahl eines Authentifizierungstoken).
Mögliche Ziele für Angreifer im Kontext von IMDs/BANs sind (nach [Rus+14] und [Mar18] sowie in Anlehnung an [AY16]):
Angriffsziel Patient: Der Angreifer könnte dem Patienten körperlichen Schaden zufügen, entweder durch Veränderung der Therapie des IMD oder z.B. durch Angriffe auf die Batterieladung des IMD (siehe Kapitel 4). Kriminelle Organisationen könnten den Patienten bedrohen und erpressen.
Angriffsziel Patienteninformationen: Der Angreifer könnte durch IMD/BAN Informationen über den Patienten (siehe auch die Datenschutzziele in Abschnitt 1.4) erlangen. So könnte ein Angreifer den Patienten tracken oder die Informationen über den getragenen IMD/über die eingesetzte Therapie sammeln und das vorliegende Krankheitsbild extrahieren und diese an Werbefirmen verkaufen oder den Patienten damit erpressen. Auch ist es möglich, aus den EKG-Daten eines Herzschrittmachers die Zeitpunkte zu ermitteln, wann eine Person Geschlechtsverkehr hat [Mar18] (was wiederum möglicherweise zur Erpressung verwendet werden kann).
Angriffsziel Hersteller des IMDs: Der Angreifer könnte durch Androhung der Aufdeckung von entdeckten Schwachstellen versuchen den Hersteller zu erpressen. Auch

könnte ein Angreifer versuchen, durch Offenlegung von Schwachstellen finanzielle Vorteile zu erzielen. So veröffentlichte die Investmentfirma Muddy Waters nach Benachrichtigung durch eine Gruppe von Security-Forschern namens MedSec 2016 Sicherheitslücken in ICDs der Firma St. Jude, worauf deren Aktienkurs um 10 % fiel. Beide Parteien profitierten vom Rückgang des Aktienkurses von St. Jude [Mar18].
Angriffsziel Systemressourcen des IMD: Hier geht man davon aus, dass solch ein Angriff unabsichtlich durch eine Malware geschieht, die nicht weiß, dass sie auf einem IMD läuft, da dieser nur geringe Energie- und Rechenleistungsressourcen hat.

1.6 Angriffsszenarien für IMDs/BANs

Konkrete Angriffsszenarien gegen IMDs (und die bei erfolgreichem Angriff möglicherweise verletzten Schutzziele) sind (nach [KC22] und [Rus+14])

- **Abhörangriff (engl. eavesdropping)**: Ein Angreifer hört die Funkkommunikation zwischen IMD und Programmer mit. Dadurch könnte er z.B. das genaue Modell und die Softwareversion des IMD oder auch Patientendaten und Gesundheitsdaten erhalten. (verletztes Schutzziel : Vertraulichkeit)

- **Seitenkanalangriff:** Ein Angreifer empfängt vom IMD/BAN Signale, die von diesem unbeabsichtigt ausgesendet werden, wie z.B. elektromagnetische Strahlung und akustische Signale. Das Messen der Ausführungszeit zwischen Signalen von IMD und Programmer kann auch ein Seitenkanalangriff darstellen. So werden dann aus den elektrischen, akustischen oder chronologischen Daten dann Daten oder Schlüssel extrahiert. (verletztes Schutzziel : Vertraulichkeit)

- **Denial of Service (DoS) Angriff**: Die korrekte Funktion des IMDs wird verhindert. Durch Blockieren von Funkfrequenzen könnte ein Angreifer die Kommunikation zwischen Mediziner und IMD verhindern. Der Angreifer könnte sich auch gegenüber dem IMD als berechtigte Gegenstelle ausgeben und diesem extrem viele Nachrichten senden, auf die dieser dann antworten muss. Dies führt dann nach einer gewissen Zeit zur Erschöpfung der Batterie des IMDs. Diese Art von DoS-Angriff heißt **Battery-Draining-Angriff** (siehe auch Kapitel 4) (verletzte Schutzziele : Verfügbarkeit)

- **Replay Angriff**: Ein Angreifer hört die Kommunikation zwischen Programmer und IMD ab und sendet diese Nachrichten erneut an das IMD. (verletzte Schutzziele : Vertraulichkeit, Integrität)

- **Reordering Angriff:** Ein Angreifer hört Nachrichten zwischen Programmer und IMD ab und jammt sie gleichzeitig, so dass sie nicht bei der Gegenstelle ankommt. Danach sendet er die Nachrichten in veränderter Reihenfolge.

- **Man in the Middle (MitM) Angriff**: Der Angreifer platziert sich zwischen IMD und Programmer und gibt sich für die jeweils anderen Seite als die legitime Gegenstelle aus. Ein realistischeres Szenario wäre ein MitM-Angriff, falls das IMD

mit einem Server kommunizieren würde. (verletzte Schutzziele : Vertraulichkeit, Integrität)

- **Code Injection Angriff**: Dem Angreifer gelingt es Teile der Software des IMD zu verändern, z.B. durch Kapern des Updateprozesses der Firmware des IMD. (verletzte Schutzziele : Vertraulichkeit, Integrität, Verfügbarkeit)

Zum Schutz der Integrität der Daten auf dem IMD kann man MACs (message authentification codes) einsetzen, zum Schutz der Vertraulichkeit bietet sich der Einsatz von Verschlüsselung an. Hierbei ist die Sicherheit der kryptographischen Schlüssel entscheidend. Dabei muss man nicht nur den Schlüssel betrachten, sondern alle Operationen der Schlüsselverwaltung wie Schlüsselerzeugung, Schlüsselverteilung, Schlüsselspeicherung, Schlüsselbenutzung und auch Schlüsselwiderruf (engl. revocation) [KC22].
Die Nutzung von Verschlüsselung bedeutet für IMDs einen erhöhten Stromverbrauch und somit für den Patienten ein kürzeres Zeitintervall nach dem das IMD per Operation ersetzt werden muss. Außerdem muss die Verschlüsselung stark genug sein, um alle Angriffe abzuwehren. Hierbei ergibt sich ein Spannungsfeld zwischen Security und Safety, da in Notsituationen das medizinische Personal schnell auf die IMDs zugreifen können muss, um diesen bei Bedarf auszulesen bzw. umprogrammieren zu können.

1.7 Ausblick auf die Arbeit

In den letzten Jahren griffen viele Forscher das **touch-to-access** Prinzip auf [Ros+13b], das den unverschlüsselten direkten Zugriff auf das IMD des Patienten erlaubt, wenn der Programmer die Haut berührt. Die Logik ist hier, dass, wenn ein Angreifer physischen Zugriff auf den Patienten hat, er nicht das IMD benötigt, um dem Patienten (physisch) zu schaden. Aufbauend auf dieser Prämisse wurden Verfahren entwickelt, die physische (sich verändernde) Merkmale des Patienten zur Erzeugung von kryptographischen Schlüsseln zwischen IMD und Programmer verwenden. Fallstricke bei diesen Verfahren sind Entropieverlust und die Erzeugung kryptographischer Schlüssel aus verrauschten (noisy) Messungen physikalischer Patientenmerkmale. Der Einsatz physiologischer Signale zur Schlüsselerzeugung/zum Schlüsselaustausch wird ausführlich in Kapitel 2 erläutert.
Andere Verfahren, die sich nach dem touch-to-access Prinzip richten, nutzen einen künstlichen physikalischen Datenkanal, um Daten (zur Schlüsselgenerierung oder zum Schlüsselaustausch) zwischen Patient und Programmer zu übertragen. Sie verwenden u.a. Schallwellen und Vibrationen für den Schlüsselaustausch. Wir behandeln diese Verfahren in Kapitel 7.
Verfahren, die ähnlich wie bei den am touch-to-access Prinzip orientierten Protokolle auf eine Prüfung der Nähe des Programmers zum IMD setzen, sind die sog. Distance-Bounding-Protokolle. Diese messen die Entfernung zwischen IMD und Programmer, meistens durch die Umlaufzeit von Signalen zwischen beiden, um die Nähe des Kommunikationspartners sicherzustellen. In Kapitel 6 geben wir einen Überblick.

Eine große Gefahr für IMDs sind Angriffe auf die Batterie des IMDs, bei denen ein Angreifer permanent dem IMD eine Kontaktaufnahme des Programmers vortäuscht, auf die dieser jedes Mal antworten muss. Dieses Thema mit den dazugehörigen Abwehrmaßnahmen behandeln wir in Kapitel 4.
Eine Möglichkeit, ein hohes Maß an Security zu erreichen und dabei die Batterie des IMDs zu schonen ist der Einsatz eines externen Zwischengerätes zwischen IMD und Programmer, das als Proxy zwischen diesen fungiert. Dieses Gerät muss vom Patienten immer mitgeführt werden. Wir stellen dieses Konzept in Kapitel 3 vor.
In Kapitel 5 behandeln wir Ansätze, durch Erkennung ungewöhnlicher Zugriffsmuster auf das IMD Angriffe zu erkennen und im Idealfall abzuwehren.
Echte ausgeführte Hacks auf IMDs werden in Kapitel 8 vorgestellt.

2 Nutzung von physiologischen Signalen zur Verschlüsselung

Die Verwendung von physiologischen Signalen (PS) des Patienten zur Schlüsselerzeugung zwischen zwei Geräten wurde erstmals 2003 von Cherukuri et al. vorgeschlagen [CVG03]. PS sind Körpersignale, die sich mit der Zeit verändern und die einen gewissen Grad an Zufälligkeit besitzen.
Das am häufigsten verwendete PS für die Schlüsselgenerierung ist das EKG. In der Literatur werden auch andere PS wie das PhotoPlethysmoGramm (PPG), Blutzucker, Blutdruck, Temperatur, Hämoglobin und Blutfluss vorgeschlagen [Yao+11].
Poon et al. schlagen in [PZB06] vor, die Zeit zwischen den Herzschlägen des Patienten, die sie als Inter-Puls-Interval (IPI) bezeichnen, als Datenquelle für die Schlüsselerzeugung zu verwenden. Das IPI weist nach [PZB06] zwei wünschenswerte Eigenschaften auf:

1. das IPI bietet ein hohes Maß an Zufälligkeit
2. das IPI kann überall am Körper durch Berühren der Haut des Patienten gemessen werden

Bei einem PS-basierten Schlüsselvereinbarungsprotokoll messen Programmer und IMD für einen gewissen Zeitraum das gewählte PS und erzeugen daraus einen gemeinsamen kryptografischen Schlüssel.
Für ein klassisches Verschlüsselungsverfahren geht man davon aus, dass der verwendete kryptographische Schlüssel gleichverteilt zufällig gezogen wird [IA20] und dass er auf beiden Seiten der Kommunikation identisch ist. Die Werte der PS-Messungen sind jedoch in der Regel wegen des Rauschens des gemessenen Signals nicht gleich, sondern nur ähnlich, und haben eine unbekannte Verteilung, sind also nicht gleichverteilt. Man muss also einen kryptographischen Schlüssel aus unterschiedlichen, aber ähnlichen physiologischen Messwerten erzeugen.
Dies wird in einigen Arbeiten zur Erzeugung von Schlüsseln aus PS ad hoc gemacht, z.B. durch einfache Bitparität oder Verwerfen von sich zu sehr unterscheidenden Messwerten bei IMD und Programmer. Andere Arbeiten haben eine strategischere Herangehensweise und verwenden eine Fuzzy-Kryptoprimitive: die Verknüpfung von Fehlerkorrekturfunktionen mit Kryptographie zur Verwendung „unscharfer“ (engl. fuzzy) Schlüssel in der Kryptographie.
Juels et al. waren die ersten, die zwei praktische Umsetzungen dieser Fuzzy-Kryptoprimitive vorschlugen: das Fuzzy Commitment Schema [JW99] und darauf aufbauend der Fuzzy Vault [Jue02] [JS06]. Dodis et el entwickeln in [DRS04] auf dem Fuzzy Vault aufbauend den Fuzzy Extraktor zur Umwandlung verrauschter PS-Messdaten in unscharfe

kryptographische Schlüssel. Genauere Beschreibungen von Fuzzy Commitment, Fuzzy Vault und Fuzzy Extraktor liefern wir jeweils in Abschnitt 2.1 , Abschnitt 2.2 und Abschnitt 2.3. In mehreren Artikeln wurde vorgeschlagen, die Fuzzy Kryptoprimitive zur Schlüsselerstellung und/oder Authentifizierung zu verwenden. In [CVG03] stellen Cherukuri et al. 2003 ein Fuzzy-Commitment-basiertes Schlüsselverteilungsprotokoll vor, das sie Biosec nannten. Dieses und Kritik daran behandeln wir in Abschnitt 2.4. Weitere Arbeiten die das Fuzzy Commitment verwenden sind u.a. [BSZ04], die als PS die Herzratenvariabilität (HRV) verwenden, ein eng mit dem IPI zusammenhängender Wert und [BH07] [Yao+10].
Miao et al. schlugen vor, das ursprüngliche Fuzzy Vault Schema so zu modifizieren, dass es auch im Kontext von Körpersensornetzwerken angewendet werden kann [Mia+09]. Dabei behaupten sie, dass ihr Verfahren eine geringere Fehlerrate in Bezug auf abgebrochene Schlüsselvereinbarungen wegen zu unterschiedlicher Messwerte der kommunizierenden Parteien hat als die theoretischen Messreihen in [PZB06]. 2008 stellten Venkatasubramanian et al [VBG08] EKA vor, ein auf dem EKG-Signal basierendes Verfahren, deren Extraktionsmethode der Messdaten allerdings diese zu sehr verfälscht und somit keinen guten Schlüsselerzeugungsalgorithmus darstellt [Ven+10]. Im gleichen Jahr präsentierten sie ein Fuzzy-Vault-basiertes Schlüsselvereinbarungsprotokoll namens PKA (engl. PPG-based key agreement) [Ven+08], welches PPG-Signale zur Erzeugung eines symmetrischen kryptografischen Schlüssel verwendet. 2010 stellten Venkatasubramanian et al schließlich ein Fuzzy-Vault-basiertes Schlüsselvereinbarungsprotokoll namens PSKA (engl. Physiological-Signal-based Key Agreement) vor [Ven+10], das auf den Erfahrungen mit EKA und PKA aufbaut. Allerdings haben Bagade et al. gezeigt, dass es möglich ist, das physiologische Signal von PSKA bis zu einer Distanz von $1,20$ m als Angreifer auszulesen [Bag+13]. Hu et al. schlugen 2013 das Verfahren OPFKA (engl. Ordered-Physiological-Feature-based Key Agreement) vor, das versucht, die Einschränkungen von PSKA zu überwinden [Hu+13]. Rostami et al. zeigten jedoch, dass OPFKA für einen Angriff anfällig ist, der die Verwendung der Hash-Funktion zur Erweiterung der Merkmalsgröße ausnutzt [Ros+13a]. OPFKA und die Angriffe darauf stellen wir in Abschnitt 2.7 vor. Weitere Verfahren, die auf dem Fuzzy Vault aufbauen findet man in [Raj+12] und [Zhe+14a].
Rostami et al. [Ros+13b] präsentierten 2013 Heart-to-Heart (H2H), ein Pairing-Protokoll das ein Commitment-Schema und TLS verwendet. Marin et al fanden jedoch einen sog. Spiegelangriff und einen MitM-Angriff gegen H2H [Mar+16b]. H2H und die Angriffe darauf werden in Abschnitt 2.6 dargestellt.
In [Xu+11] schlagen die Autoren ein das IPI verwendendes Schlüsselvereinbarungsprotokoll zwischen einem externen Schutzgerät namens Guardian und dem IMD vor, welches wir in Abschnitt 2.5 vorstellen.
2016 stellten Marin et al ein Protokoll vor, das den Fuzzy Extractor verwendet [Mar+16a].
Einige Protokolle verwenden klassische Fehlerkorrekturverfahren wie Reed-Solomon-Codes [Zha+12] oder Bose-Chaudhuri-Hocquenghem-Codes [Zhe+15] [Zag+15]. Es gibt auch Verfahren, die eine Authentifikation im Notfall über den Fingerabdruck ermöglichen wie Finger2Heart (F2H) [Zhe+19] und [Bel+19]. Hier wird vorausgesetzt, dass im Programmer ein Fingerabdruckleser integriert ist. Nach der Authentifizierung

eines Fingers des Patienten kommunizieren hier IMD und Programmer aber unverschlüsselt.
Beck et al benutzen als PS-Methode die Ballistokardiographie. Dabei misst man sich wiederholende Bewegungen/Erschütterungen des menschlichen Körpers durch den Herzschlag. Eine Auflistung aller Protokolle, die für IMD/Programmer verwendet werden können findet sich in Tabelle 2.1 in Abschnitt 2.8. In diesem Kapitel zeigen wir auch weitere mögliche Angriffspunkte PS-basierter Verfahren auf.

2.1 Das Fuzzy Commitment Schema

(Die Kapitel über Fuzzy-Verfahren orientieren sich an [Mar+16b].)
Das Fuzzy Commitment Schema aus [JW99] ist wie herkömmliche Commitment-Schemata hiding und binding. hiding bedeutet, dass das Commitment zunächst keinerlei Information über den festgelegten Wert an den Kommunikationspartner oder einen Angreifer preisgibt. binding bedeutet, dass es dem Ersteller des Commitments nicht möglich ist, seinen festgelegten Wert nachträglich zu ändern.
Im Folgenden betrachten wir einen Beispielablauf des Fuzzy Commitment Schemas: Alice möchte einen Schlüssel k mit Hilfe eines Fuzzy-Commitment-Schemas F sicher an Bob übertragen. Die von Alice und Bob verwendeten verrauschten Ausgangswerte aus einer PS-Messung bezeichne man als w und w'. Zuerst erzeugt Alice einen zufälligen Schlüssel k und fügt dann durch ein Fehlerkorrekturverfahren (ECC) eine gewisse Redundanz zu k hinzu, wodurch man den Wert $c = \mathsf{ECC}(k)$ erhält. Alice berechnet dann $\delta = c \oplus w$, um c auf dem Weg zu Bob zu maskieren. Anschließend berechnet Alice $H(c)$, wobei H eine Einwegfunktion ist (z. B. eine kryptografische Hash-Funktion). Das Commitment, das also von Alice an Bob geschickt wird hat die Form $F(c, w) = \{H(c), \delta = c \oplus w\}$.
Im Gegensatz zu herkömmlichen Commitment-Schemata kann Bob das Commitment öffnen, indem er einen beliebigen Messwert w' verwendet, der nach einer vorher festgelegten Metrik nahe bei w liegt. Wenn der Abstand zwischen c' und c kleiner ist als die maximale Anzahl an Fehlern, die das gewählte Fehlerkorrekturverfahren korrigieren kann, dann kann Bob c erfolgreich aus c' berechnen, weil dann $c' = \delta \oplus w'$ gilt. Zur Überprüfung, ob c der korrekte Wert ist, berechnet Bob $H(c)$ und vergleicht ihn mit dem von Alice im Commitment gesendeten Hashwert.

2.2 Das Fuzzy Vault Schema

Das Fuzzy-Vault-Schema aus [JS06] ist eine ordnungsinvariante Version des von Juels et al [JW99] vorgeschlagenen Fuzzy Commitment Schemas. Der Fuzzy Vault verbirgt ein Geheimnis k in einen Tresor (Vault) durch Verwendung von einer Menge von Merkmalen A (im Gegensatz zur Verwendung von nur einem Merkmal beim Fuzzy Commitment Schema). Dieser Tresor kann dann nur mit einer Menge von Merkmalen B, die in einem vorher festgelegten Abstandsmaß denen von A ähnlich sind, geöffnet werden. Die Merkmalsmengen A und B können beliebig geordnet sein. Das Ziel ist

wie beim Fuzzy-Commitment-Schema einen Schlüssel k von Alice an Bob zu schicken, ohne dass ein Angreifer den Schlüssel k ermitteln kann.
Im ersten Schritt wählt Alice die Koeffizienten eines Polynom $p(x)$ so, dass k in diesen kodiert wird. Danach wählt sie zufällig eine Menge A von Merkmalen. Jedes Element von A wird als x-Koordinatenwert auf der Kurve p interpretiert. Alice berechnet dann für jedes Element von A den zugehörigen y-Wert des Polynom p. Anschließend wählt Alice zufällige Punkte, die nicht auf p liegen, die wir Täusch-Punkte (engl. chaff points) nennen wollen. Diese Punkte sollen die Kurven-Punkte verbergen. Alice mischt die Kurven-Punkte und die Täusch-Punkte und sendet diese als Commitment für den Schlüssel k an Bob. Bob versucht nun den Tresor mit seiner Merkmalsmenge B zu öffnen. Falls bei B und A eine Anzahl an Kurvenpunkten, die weniger als ein vorgegebener Grenzwert ist, gleich sind, kann Bob größtenteils die Kurvenpunkte von Alices Polynom p identifizieren und durch eine vorgegebene Fehlerkorrekturfunktion das Polynom p rekonstruieren. Mit p kann Bob nun k berechnen.
Die Sicherheit dieses Verfahrens beruht auf der Annahme, dass die Angreifer nicht zwischen den Kurvenpunkten und den Täusch-Punkten unterscheiden können.

2.3 Fuzzy Extraktor

Dodis et al geben in [DRS04] formale Definitionen und schlagen zwei Verfahren für die Umwandlung verrauschter PS-Messwerte in kryptografische Schlüssel vor: den Secure Sketch und den Fuzzy-Extraktor.
Die Secure Sketch Methode toleriert Fehler im kryptographischen Schlüssel, deren Abstand unterhalb eines festgelegten Grenzwertes liegt, sie berücksichtigt jedoch nicht die ungleichmäßige Verteilung der Ausgangsdaten. Darauf aufbauend entwickeln die Autoren den Fuzzy-Extraktor, der nahezu gleichverteilte Zufallswerte R aus einer Eingabe w zu extrahiert, wobei je nach Abstandsmaß (Hammingdistanz, Editierabstand, Komplementärmenge) entsprechend Fehler toleriert werden. Bei einem verrauschten Eingabewert w' ist es möglich, R wiederherzustellen, wenn der Abstand von w' zu w im gewählten Abstandmaß einen festgelegten Grenzwert nicht überschreitet.
Fuzzy-Extraktoren bestehen aus zwei Funktionen: einer Generatorfunktion Gen und einer Reproduktionsfunktion Rep. Die Generatorfunktion Gen berechnet nach Eingabe von w einen Schlüssel k und Hilfsdaten P aus. Die Reproduktionsfunktion Rep berechnet nach Eingabe von w' und P den Schlüssel k.

2.4 BioSec

In [CVG03] erwähnen die Autoren, dass das EKG-Signal als Zufallsquelle nicht ausreicht, und schlagen vor, mehrere PS zu verwenden wie z.B. Glukosewert, Blutdruck, Temperatur, Hämoglobinwert oder Blutfluss. Sie erklären nicht, wie die PS ermittelt werden, sondern gehen schon von einem aus physiologischen Werten ermittelten 128-Bitstring aus. Sie verwenden das Fuzzy Commitment Schema aus [JW99] zur maskierten Übertragung eines Sitzungsschlüssels. Die Fehlerkorrektur im Fuzzy Commitment Schema korrigiert die in jedem zum BAN/IMD gehörigen Gerät unterschiedlichen

Messwerte. Abbildung 2.1 enthält die für das Protokoll BioSec benötigten Definitionen von Werten und Funktionen.

m_s	enthält eine aus einer Kombination biometrischer Daten erzeugt 128 Bit Zufallszahl
r_u	enthält eine eindeutige 128 Bit Geräte-ID
K_{commit}	speichert das Commitment $m_s \oplus r_u$
$K_{session}$	enhält den von Gerät 1 erzeugten 128 Bit Session Key[1]
$\mathsf{Enc}_K(\cdot)$	ist die eingesetzte Verschlüsselungsfunktion, die RC5 mit Schlüssel K verwendet
$\mathsf{Dec}_K(\cdot)$	ist die eingesetzte Entschlüsselungsfunktion, die RC5 mit Schlüssel K verwendet
$Data$	die zu sendenden Daten
$eData$	enspricht den verschlüsselten Daten, also $eData := \mathsf{Enc}_{K_{session}}(Data)$
MAC	berechnet den 128 Bit Message Authentification Code durch den MD5-Algorithmus
m	128 Bit Signatur von $eData$, also $m := \mathsf{MAC}(eData)$
S_{com}	enthält das Commitment auf $K_{session}$ durch den Commitment-Key K_{commit}
$H(\cdot)$	ist eine Einwegfunktion, z.B. eine kryptographische Hashfunktion
ECC	die im Fuzzy Commitment Schema eingesetzt Fehlerkorrekturfunktion

Abbildung 2.1: Abkürzungen des BioSec-Protokolls

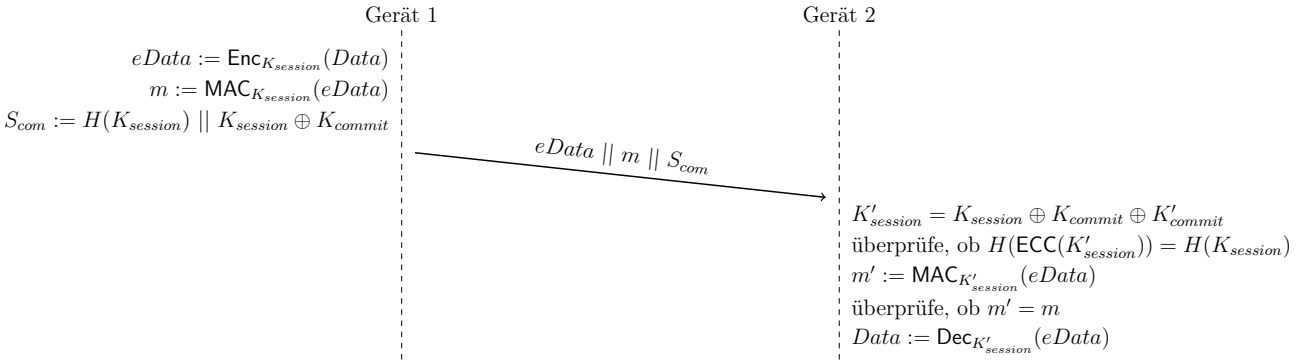

Abbildung 2.2: gleichzeitige Schlüsselverteilung und Datenübertragung des BioSec Protokolls

Die Schlüsselverteilung zwischen zwei BAN-Sensoren bzw. Programmer und IMD erfolgt durch das Protokoll in Abbildung 2.2, das pro teilnehmendem Gerät nur aus einer Nachricht besteht. Im Folgenden wollen wir die miteinander kommunizierenden Geräte mit „Gerät 1“ und „Gerät 2“ bezeichnen. Das BioSec-Protokoll funktioniert wie folgt:
Gerät 1 erzeugt einen zufälligen 128 Bit Sitzungsschlüssel $K_{session}$, der wegen der im Fuzzy Commitment Schema verwendeten intrinsischen Fehlerkorrektur nur in einen Teil des 2^{128} großen Schlüsselraumes liegt [JW99]. Es verschlüsselt dann die Daten $Data$ zu einem Chiffretext $eData$, d. h. $eData := \mathsf{Enc}_{K_{session}}(Data)$. Anschließend wird ein 128-Bit MAC m von $eData$ berechnet ($m := \mathsf{MAC}_{K_{session}}(eData)$). Um $K_{session}$ an Gerät 2 zu schicken, ohne es potentiellen Angreifern zu offenbaren, maskiert Gerät 1 $K_{session}$ mittels XORen mit K_{commit} und berechnet den Hashwert $H(K_{session})$. Dann werden $eData$, m und S_{com} an Gerät 2 geschickt, wobei $S_{com} := H(K_{session}) \mid\mid K_{session} \oplus K_{commit}$. Anschließend versucht Gerät 2, die von Gerät 1 durchgeführte Maskierungsoperation rückgängig zu machen, um $K_{session}$ mit Hilfe von $K'_{session}$ und c aus S_{com} zu erhalten. Gerät 2 berechnet also $K'_{session} =$

$K_{session} \oplus K_{commit} \oplus K'_{commit}$. Da die biometrischen Messwerte höchstwahrscheinlich unterschiedlich sind wird $K'_{session} \neq K_{session}$ gelten, wobei der Abstand der beiden Werte in der Abstandsmetrik des gewählten Fehlerkorrekturverfahrens sehr wahrscheinlich unterhalb des im Fehlerkorrekturverfahren gewählten Grenzwertes liegt, oberhalb dessen Abweichungen nicht mehr korrigiert werden können. Gerät 2 wird also mit Hilfe der Fehlerkorrekturfunktion ECC $K_{session}$ aus $K'_{session}$ wiederherstellen und dann prüfen ob die Hashwerte übereinstimmen ($H(\mathsf{ECC}(K'_{session})) = H(K_{session})$). Ist diese Bedingung erfüllt, kann Gerät 2 mit Hilfe von $K_{session}$ die Nachricht entschlüsseln.

2.4.1 Kritik an BioSec

Folgende Kritikpunkte ergeben sich u.a. aus [Mar+16b]:

- **Zufälligkeit von K_{commit}:** $K_{session}$ wird mit einer Kombination aus biometrischen Daten und der statischen ID r_u eines Patienten XOR-verknüpft, um zu verhindern, dass Angreifer diesen Wert während des Sendevorgangs von Gerät 1 zu Gerät 2 erhalten. Es ist unklar, zu welchem Zweck die statische ID verwendet wird und ob/wie der Empfängersensor diese ID kennt. Die statische ID bietet jedoch keine zusätzliche Sicherheit, da sie durch das Erfassen zweier Nachrichten und die Subtraktion ihrer K_{commit}-Werte entfernt werden kann.

- **Biometrische Zufälligkeit:** Die Autoren von [CVG03] haben selbst darauf hingewiesen, dass ein mögliches Manko ihrer Lösung das Fehlen einer ausreichenden Entropie sein könnte.

- **Wiederverwendung des Schlüssels:** Für die Gewährleistung der Vertraulichkeit und Authentizität wird der gleiche Schlüssel verwendet. Eine einfache Lösung für dieses Problem wäre die Verwendung von $K_{session}$ als Hauptschlüssel und die anschließende Ableitung zweier unabhängiger Zufallsschlüssel mit einer Standard-Schlüsselableitungsfunktion oder noch besser die Verwendung einer authentifizierten Verschlüsselung.

- **Nichtberücksichtigung der Verwendung einer Fehlerkorrekturfunktion (ECC):** Die Verwendung einer Fehlerkorrekturfunktion reduziert die effektive Schlüssellänge, da sie Redundanz hinzufügt und somit die Entropie verringert. Eine mögliche Lösung wäre, die Schlüssellänge zu erhöhen, um das gleiche Maß an Sicherheit zu erreichen, das erwartet wird, wenn keine Fehlerkorrektur verwendet wird. $K_{session}$ wird jedoch als 128-Bit-Schlüssel ohne Berücksichtigung dieses Entropieverlustes aus dem Raum der Codewörter des Fehlerkorrekturverfahrens gewählt. Dies erleichtert Brute-Force-Angriffe. Da der Hash des verwendeten Schlüssels in der Nachricht enthalten ist, kann ein erfolgreiches Raten des Schlüssels sofort überprüft werden. Darüber hinaus könnten diese Berechnungen offline durchführt werden. wobei eine Tabelle mit allen möglichen Schlüsseln und den entsprechenden Hash-Werten erstellt wird. Dann kann durch Abfangen einer einzigen Nachricht der Schlüssel $K_{session}$ ermittelt werden.

- **Verknüpfung von Schlüsselaustausch und Datentransfer:** Bei BioSec wird bei jeder übertragenen Nachricht, der Schlüssel übermittelt, bzw muss bei jedem Schlüsselaustausch auch eine Nachricht mitgeschickt werden. Eine Trennung von Schlüsselvereinbarung und Datenaustausch durch zwei verschiedene Protokolle findet nicht statt.

- **Unvollständigkeit des Protokolls:** Bei BioSec bleibt nicht nur die Erzeugung des Schlüssels, die Berücksichtigung der Reduktion der Entropie des Schlüssels durch Fehlerkorrektur sondern auch jegliche Informationen über die Initialisierung der Schlüsselerzeugung aus. Hier muss eine zeitliche Synchronisation stattfinden, für die man mangels eines gemeinsamen Schlüssels dann auch unverschlüsselt kommunizieren muss (asymetrische Kryptographie wie z.B. das Diffie-Hellman-Verfahren wird wegen der erforderlichen Rechenleistung von den Autoren ausgeschlossen). Auch wenn die Autoren erwähnen, dass ein Re-Keying für implantierte Geräte möglich sein soll, beschreiben sie nicht, wie dieses mit ihrem Verfahren stattfinden soll.

2.5 Schlüsselgenerierung mit EKG-Signal in IMDGuard

Das Schlüsselvereinbarungsprotokoll aus IMDGuard [Xu+11] extrahiert synchron in IMD und einem externen Gerät namens „Guardian“ jeweils die vier untersten Bits des IPIs. Die Autoren gehen nach Analyse in ihrem Paper davon aus, dass drei der vier Bits gleich sind, und in sehr seltenen Fällen nur 2 Bits übereinstimmen. Sie teilen ihr Verfahren in 2 Runden, wobei das Protokoll mit gleichem Schlüssel auf beiden Seiten endet, wenn nur 1 Bit pro 4-Bit-IPI-Wert fehlerhaft ist, andernfalls wird noch die zweite Runde des Verfahrens ausgeführt :
Runde 1: Für jedes IPI erhalten sowohl das IMD als auch der Guardian einen 4-Bit-Block. Beide Seiten berechnen die Parität ihres eigenen Blocks und tauschen diese Informationen aus. Wenn die Paritäten unterschiedlich sind, wird der Block verworfen. Andernfalls extrahiert jede Seite die ersten 3 Bits des Blocks; das 4. Bit wird verworfen, da die Parität ein Bit Information preisgibt. Dieser Prozess wird fortgesetzt, bis beide Seiten 129 Bits erhalten. Das IMD verschlüsselt sie dann mit der Hash-Funktion SHA-1 und sendet den Hash-Wert an den Guardian. Der Guardian vergleicht diesen Hashwert mit seinem eigenen und benachrichtigt das IMD über das Ergebnis. Wenn beide Hash-Werte übereinstimmen, ist der Algorithmus beendet. Andernfalls wird Runde 2 durchgeführt.
Runde 2: Für die 43 in Runde 1 ausgewählten IPI berechnen sowohl das IMD als auch Guardian die Parität der letzten 2 Bits eines jeden 4-Bit-Blocks und tauschen diese Informationen aus. Wiederum werden die Blöcke deren Paritäten nicht übereinstimmen verworfen. Für die verbleibenden Blöcke extrahieren beide Seiten das 2. und 3. Bit; das erste Bit wird verworfen, da die zweite Parität ebenfalls ein Bit Information preisgibt. Offensichtlich ist die Länge des Schlüssels kleiner als 128. Beide Seiten analysieren die folgenden IPIs wobei sie jetzt zwei Paritäten gleichzeitig prüfen und 2 Bits aus jedem Block, der die Paritätsprüfung besteht extrahieren. Dieser Prozess wird fortgesetzt bis beide Seiten 128 Bits haben.

2.6 Heart2Heart (H2H)

Das Heart2Heart-Protokoll (H2H) aus [Ros+13b] verwendet Public-Key-Kryptographie in Kombination mit einem Commitment-Schema und gewährt Zugriff auf das IMD nach dem Touch-To-Access-Prinzip, also für jeder Programmer, der physischen Kontakt mit dem Patienten herstellen und seine Herzfrequenz messen kann. Abbildung 2.3 gibt einen Überblick über das H2H-Pairing Protokoll.

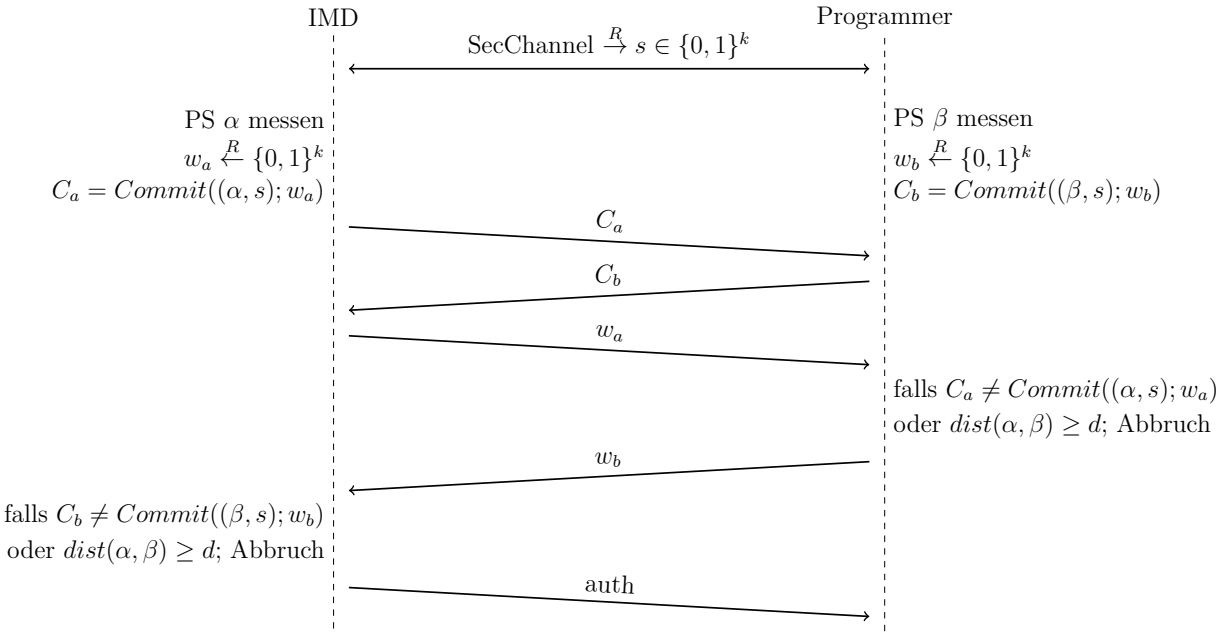

Abbildung 2.3: H2H Pairing Protokoll

H2H kann in zwei verschiedene Phasen unterteilt werden:

1. Aufbau eines sicheren Kanals zwischen IMD und Programmer
2. Authentifizierung beider Geräte durch Nachweis des Zugriffs auf das EKG-Signal

Zuerst wird ein sicherer, aber nicht authentifizierter Kanal zwischen dem IMD und dem Programmer über TLS aufgebaut. Dieser Kanal bietet Vertraulichkeit, Integrität und Freshness. Freshness bedeutet, dass bei jedem erneuten Zugriff des Programmers auf das IMD die Authentifikation durchgeführt wird und somit auch jedes Mal ein neuer Sitzungsschlüssel erstellt wird. Das IMD nimmt die Rolle eines TLS-Clients ein, während der Programmer als TLS-Server fungiert. Der Programmer legt dem IMD sein Zertifikat vor, das IMD überprüft dieses jedoch nicht. Die Autoren wollen damit die Belastung durch eine Public-Key-Infrastruktur (PKI) vermeiden. Die TLS-Sitzung gibt eine eindeutige und zufällige Zahl s aus, die nicht geheim gehalten werden muss, z. B. den Hash des TLS-Sitzungsschlüssels. Die Autoren verwenden hier TLS nur als Black-Box für einen sicheren Kanals, der dann auch noch eine Zufallszahl s liefert und nennen diesen Algorithmus SecChannel.
In der Authentifizierungsphase authentifiziert das IMD den Programmer mittels des

Touch-to-Access-Prinzips. Zu diesem Zweck nehmen das IMD und der Programmer jeweils eine IPI-Lesung vor, die mit α und β bezeichnet wird, und generieren dann eine Zufallszahl w_a bzw. w_b. Das Commitment-Schema gibt C_a bzw. C_b aus und ermöglicht es jedem der Geräte, sich auf seinem IPI-Wert (α oder β) zu committen, während es diese Werte mit Hilfe von w_a bzw w_b maskiert. Jedes Gerät commitet sich gleichzeitig auch auf s, um die Wiederverwendung von α oder β durch Angreifer auf einem anderen Kanal zu verhindern. Nachdem C_a und C_b ausgetauscht wurden, kann das IMD das Commitment C_a durch Senden von w_a öffnen. Anschließend überprüft es, ob α, w_a und s zur Erzeugung von C_a verwendet wurden, und prüft dann noch, ob der Abstand zwischen α und β kleiner ist als ein vorgegebener Schwellenwert d. Wenn alle diese Bedingungen erfüllt sind, sendet der Programmer w_b, um das Commitment C_b zu öffnen, woraus das IMD β extrahiert. Analog zum Programmer zuvor prüft das IMD nun, ob β, w_b und s verwendet wurden, um C_b zu erzeugen, und ob der Abstand zwischen α und β unter dem Schwellwert liegt. Wenn diese Bedingungen erfüllt sind, akzeptiert das IMD das Gerät mit dem es den TLS-Kanal aufgebaut hat als authentischen legitimen Programmer und schickt diesem eine Authentifikationsbestätigung.

2.6.1 Kritik an und Angriffe auf Heart2Heart

Rostami et al. [Ros+13b] geben in ihrem Paper einen Sicherheitsbeweis an, allerdings existieren mehrere Angriffe gegen das Verfahren, u.a. ein MitM-Angriff (s.u.).
Die Autoren gehen weder auf Details über das verwendete TLS-Protokoll noch das verwendete Commitment Schema ein, was eine Sicherheitsbewertung des Verfahrens erschwert.
Die Sicherheit von H2H beruht auf der Annahme, dass ein Angreifer keinen physischen Kontakt mit dem Patienten herstellen kann, wobei die Autoren auch explizit davon ausgehen, dass es für Angreifer unmöglich ist, die IPI eines Patienten aus der Entfernung zu ermitteln.
Dadurch, dass beim H2H-Protokoll das IMD zuerst komplett einen TLS-Handshake und danach noch das Authentifikationsprotokoll mit einem unbekannten Kommunikationspartner durchführt, ist es natürlich anfällig für einen Battery-Draining-Angriff.
Spiegelangriff: Marin et al. benutzen in [Mar+16b] einen sog. Spiegelangriff (engl. reflection attack), welcher ausnutzt, dass das H2H-Protokoll in beide Richtungen (vom IMD zum Programmer und umgekehrt) symmetrisch ist. Hierbei kann sich ein Angreifer ohne Kenntnis der IPI gegenüber dem IMD als Programmer ausgeben (siehe Abbildung 2.4).

Bei diesem Angriff baut der Angreifer zunächst, wie im H2H-Protokoll vorgegeben, einen sicheren, aber unauthentifizierten TLS-Kanal zum IMD auf. In der Authentifizierungsphase besteht das Ziel des Angreifers darin, den Wert s aus dem vorangegangenen TLS-Protokoll zu authentifizieren, was laut H2H-Protokoll nur durch den Nachweis der Kenntnis der IPI des Patienten erreicht werden kann. Hier schickt der Angreifer einfach die vom IMD gesendeten Nachrichten an diesen zurück (also gilt dann $C_b = C_a$ und $w_b = w_a$). Das IMD hat nun den Angreifer als legitimen Programmer authentifiziert. Diese Schwachstelle lässt sich leicht beheben, indem das IMD C_b zurückweist, wenn es

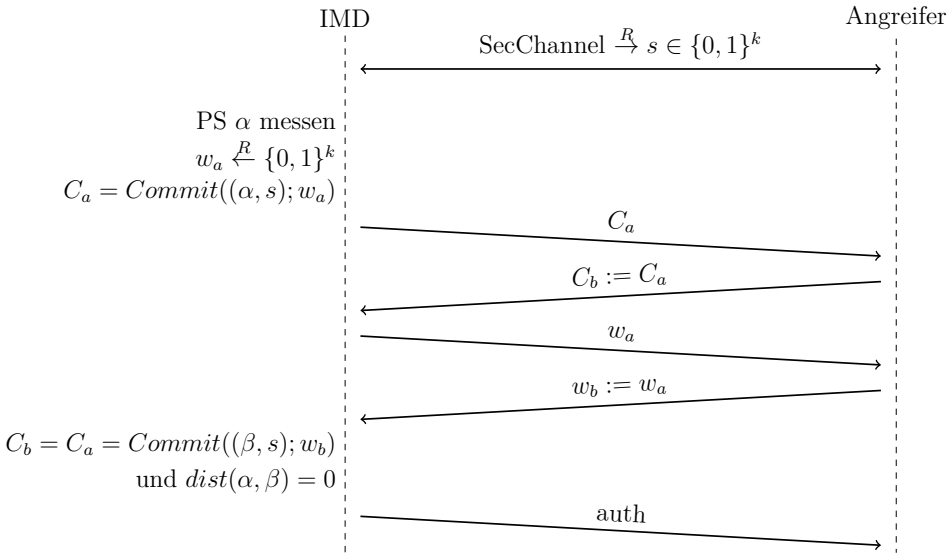

Abbildung 2.4: Spiegelangriff auf das H2H Pairing Protokoll

mit C_a identisch ist, oder indem das Protokoll so verändert wird, dass es nicht mehr symmetrisch ist.

Man-in-the-Middle-Angriff: Marin et al. fanden auch einen MITM-Angriff, bei dem der Angreifer dem Geräteprogrammierer vorgaukelt, dass er mit dem legitimen IMD kommuniziert [Mar+16b]. Der Angriff ist in Abbildung 2.5 dargestellt und funktioniert wie folgt:

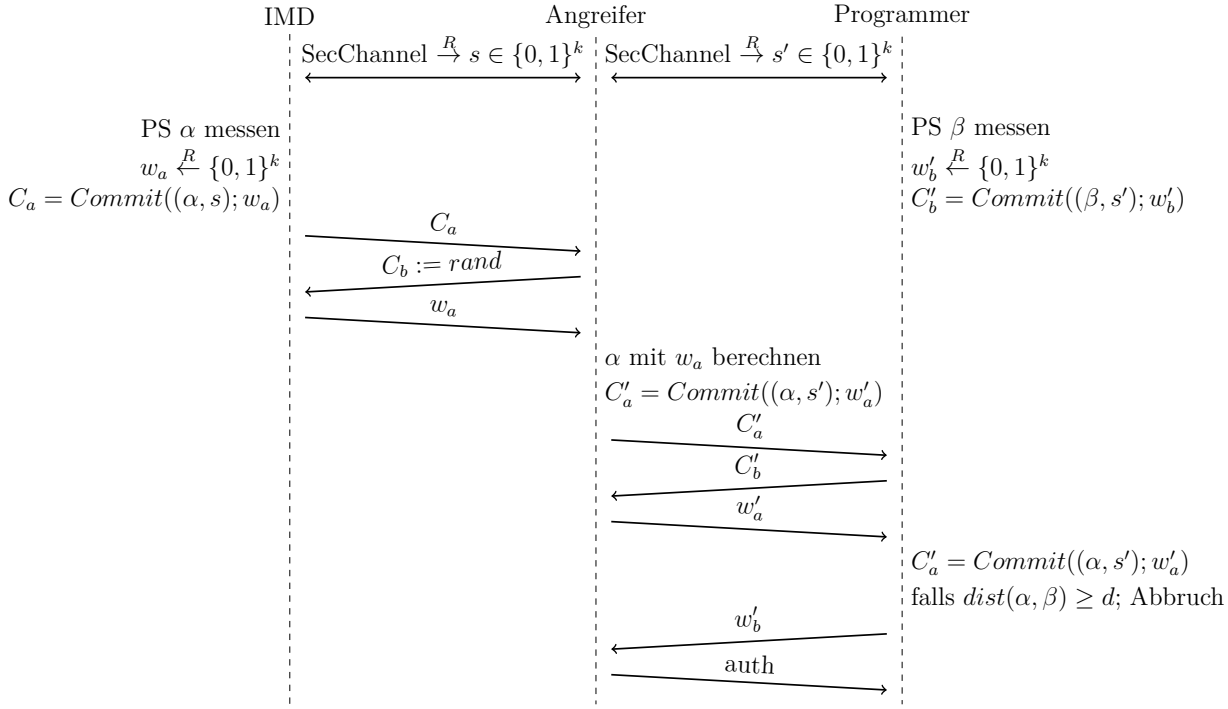

Abbildung 2.5: MitM-Angriff auf das H2H Pairing Protokoll

Der Angreifer führt zunächst das reduzierte TLS-Protokoll zum Aufbau eines verschlüsselten Kanals mit dem IMD bzw. dem Geräteprogrammierer durch und erhält dadurch die mit IMD bzw. Programmer gemeinsamen Werte s und s'. In der Authentifizierungsphase erstellt das IMD zunächst sein Commitment C_a und sendet es an den Angreifer. Nach Erhalt von C_a sendet dieser einen zufälligen Wert C_b als Commitment an das IMD. Im nächsten Schritt sendet das IMD w_a, um die Öffnung des Commitments C_a zu ermöglichen, was dem Angreifer ermöglicht, den IPI-Messwert α des IMDs zu berechnen. Er erstellt ein neues Commitment C'_a, das α enthält, und sendet es an den Programmer. Dieser erstellt und sendet dann sein Commitment C'_b an den Angreifer. Nach dem Empfang von C_b sendet der Angreifer w'_a, was es dem Programmer ermöglicht, das Commitment C'_a zu öffnen. Nach dem gleichen Verfahren sendet der Programmer w'_b an den Angreifer,der nun das Commitment C'_b öffnen kann. Schließlich kann der Angreifer einfach mit einer Auth-Nachricht antworten, ohne das Commitment C'_b öffnen zu müssen. Nach Erhalt der Auth-Nachricht ist der Programmer überzeugt, dass er mit dem IMD kommuniziert.
Dieser Angriff könnte ein Reverse-Engineering des Kommunikationsprotokolls zwischen Programmer und IMD erleichtern.

2.7 OPFKA

2013 stellten Hu et al das Verfahren OPFKA (engl. ordered physiological feature-based key agreement) vor [Hu+13]. Dieses Verfahren verwendet wie das Fuzzy Vault Schema sog. Täuschpunkte (engl. chaff points) zum Verschleiern der PS-Messwerte - allerdings werden hier die Messwerte mit den Täuschpunkten unverschlüsselt versendet!
Das OPFKA-Protokoll (siehe Abbildung 2.6) verläuft in drei Phasen:

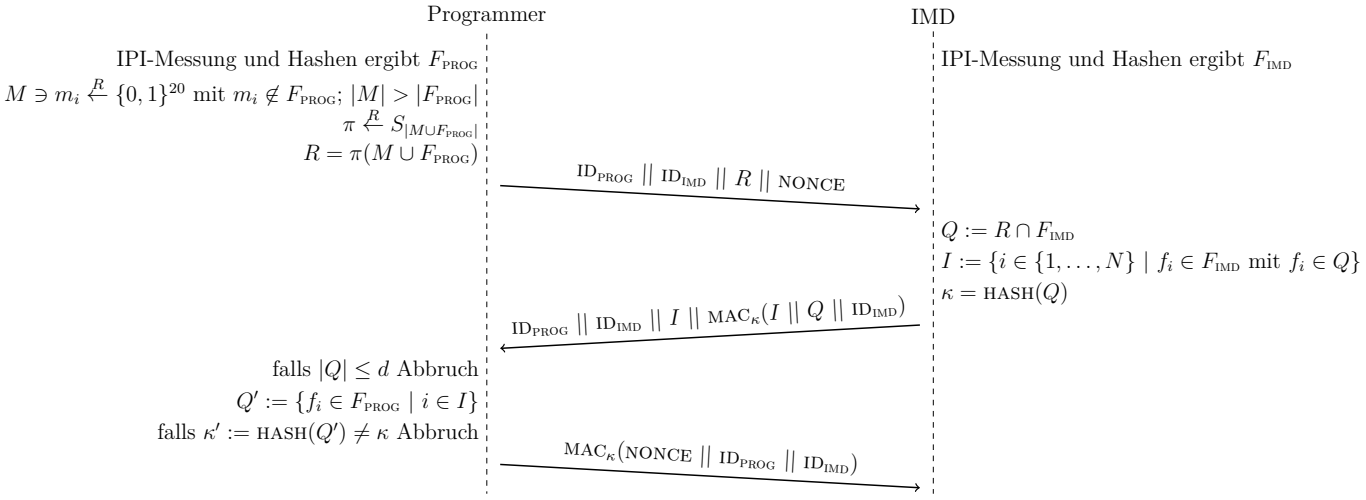

Abbildung 2.6: OPFKA Pairing Protokoll

1. Messung der physiologischen Signale. Diese werden noch verändert und dann als Merkmale (engl. features) bezeichnet.

2. Erstellung des sog. Coffers, einer Menge von Zufallswerten und den Merkmalen.
3. gegenseitige Authentifikation von IMD und Programmer durch Nachweis der Kenntnis der EKG-Werte

Zunächst führen IMD und Programmer zunächst synchrone lokale EKG-Messungen durch und übersetzen diese Messwerte in eine zeitlich geordnete Folge von sog. Merkmalen (engl. features). Sie schlagen für die Messung der Signale zwei alternative Verfahren vor.

erweiterte FFT-Methode: Hier verwenden die Autoren die erweiterte FFT Methode aus [Ven+10] zur Extraktion von 30 13-Bit Werten aus dem PS-Messwerten, wobei hier die Extraktion von (vollständigen) EKG-Daten oder von PPG-Daten vorgeschlagen wird. Dieses Verfahren ist schnell, man benötigt für die Messung der EKG-Daten nur 4 Sekunden, für die Messung der PPG-Daten 12, 8 Sekunden. Die Autoren wenden nun eine Hash-Funktion mit einem Salt s auf diese Werte an und nehmen die ersten 20 Bit der Ausgabe als Merkmal. Dies soll die Gefahr der Kollision von Messwerten mit den in Schritt 2 erstellten Zufallswerten verringern.

IPI-Methode: Bei der IPI-Methode sammeln IMD und Programmer gleichzeitig IPIs für etwa 1 bis 1,5 Minuten bis sie jeweils 90 Messwerte erhalten haben. Jedes IPI wird zunächst in eine binäre 4-Bit-Darstellung umgewandelt, und drei benachbarte IPIs werden zu einem 12-Bit-Merkmal verkettet. Die Autoren schätzen die Wahrscheinlichkeit, dass die geheimen Merkmale von IMD und Programmer übereinstimmen auf etwa 42 %. Auch hier erweitern sie die 12-Bit-Merkmale mittels Hash-Funktion und Salt auf 20 Bit. Hierbei wird die Reihenfolge der Elemente beibehalten.

Bei dieser PS-Extraktionsmethode müssen nur die IPIs gesammelt werden (und keine FFT berechnet werden). Die IPI-Methode benötigt deshalb im Vergleich zur FFT-Methode weniger Rechenleistung auf Kosten einer längeren Messzeit.

Die gemessenen gehashten Merkmalsvektoren der Länge N bezeichnen wir mit $F_{\text{IMD}} = \{f^1_{\text{IMD}}, f^2_{\text{IMD}}, \ldots, f^N_{\text{IMD}}\}$ bzw. $F_{\text{PROG}} = \{f^1_{\text{PROG}}, f^2_{\text{PROG}}, \ldots, f^N_{\text{PROG}}\}$ für IMD und Programmer.

Coffer-Erzeugung: Für die Erzeugung des Coffers R erzeugt der Programmer eine Menge M von Täuschpunkten mit $|M| > |f_{\text{PROG}}|$, wobei jeder Täuschpunkt ein zufälliger 20 Bit Wert ist und nicht in F_{PROG} liegen darf ($f^i_{\text{PROG}} \notin M$). Auf $F_{\text{PROG}} \cup M$ wird nun eine zufällige Permutation $\pi \in S_{|F_{\text{PROG}} \cup M|}$ angewendet und wir erhalten den Coffer $R = \pi(F_{\text{PROG}} \cup M)$, wobei $|R| = N + |M|$. Die Autoren schlagen vor, die Größe von M in Abhängigkeit von dem gewünschten Sicherheitsniveau zu wählen.

Authentifikation: Für den Authentifikationsprozess sendet der Programmer $\text{ID}_{\text{PROG}} \mid\mid \text{ID}_{\text{IMD}} \mid\mid R \mid\mid \text{NONCE}$ an das IMD, wobei ID_{PROG} und ID_{IMD} die IDs von Programmer und IMD sind und NONCE eine Nonce. Nachdem das IMD den Coffer R erhalten hat, vergleicht er R mit seinen eigenen Merkmalen und speichert die übereinstimmenden Werten in einer Menge Q. Die Positionen dieser Werte in seinem eigenen Merkmalsvektor F_{IMD} werden in einer Indexmenge I gespeichert. Danach erzeugt er den geheimen Schlüssel $\kappa = \text{HASH}(Q)$ unter Verwendung der übereinstimmenden Merkmale. Das IMD sendet nun $\text{ID}_{\text{PROG}} \mid\mid \text{ID}_{\text{IMD}} \mid\mid I \mid\mid \text{MAC}_\kappa(I \mid\mid Q \mid\mid \text{ID}_{\text{IMD}})$ an den Programmer.

Dieser identifiziert nun die gemeinsamen Merkmale anhand der Menge I. Wenn $|Q|$

größer als ein vorher festgelegter Schwellenwert d ist, berechnet der Programmer den Schlüssel $\kappa' = \text{HASH}(Q)$. Falls die Schlüssel auf beiden Seiten übereinstimmen, also $\kappa' = \kappa$ sendet er die Authentifikationsbestätigung $\text{MAC}_\kappa(\text{NONCE} \mid\mid \text{ID}_{\text{PROG}} \mid\mid \text{ID}_{\text{IMD}})$ an das IMD.
Damit der Programmer den Schlüssel κ berechnen kann, müssen die indizierten Merkmale mit denen des Empfängers übereinstimmen. Damit authentifizieren sich Programmer und IMD gegenseitig.

2.7.1 Angriff durch Reduktion der Menge der Cofferwerte

Rostami et al [Ros+13a] nutzen einen Konstruktionsfehler bei der Verwendung der Hashfunktion auf die gemessenen PS zur Reduktion der möglichen Kofferwerte. Der Angriff geht von einer physiologischen Messung der IPI-Werte aus, funktioniert aber analog auch für die Messung mit der FFT-Methode.
Wenn das IMD im Authentifizierungsschritt ein Merkmal auswählt, das in $R \cup F_{\text{IMD}}$, aber nicht in F_{PROG} liegt, kann der Programmer κ nicht berechnen und das Protokoll schlägt fehl. Um die Rate solcher Fehlschläge zu verringern wollten die Autoren von [Hu+13] die Menge der Merkmalswerte vergrößern. Die Anwendung einer Hash-Funktion erweitert jedoch die Anzahl der Merkmalswerte für einen festen Definitionsbereich D nicht. Sei $D = \{0,1\}^{12}$ die Menge der möglichen Werte für ein 12-Bit-Merkmal f. Der Hash von f wird als $H(s, f)$ für einen vorher vereinbarten Salt s berechnet. $B = \{H(s, f)\}_{f \in D}$ bezeichne den Wertebereich von $H(s, \cdot)$. Dann ist es einfach zu sehen, dass $|B| \leq |D| = 2^{12}$ ist, da $H(s, \cdot)$ eine deterministische Funktion ist. Die überwiegende Mehrheit der Täuschwerte in R werden also ungültige Merkmalswerte sein, die außerhalb von B liegen. Bezeichnen wir mit $\hat{B} = R \cap B$ die Menge der Werte im Coffer, die gültige Merkmalswerte sind. Es gilt $F_{\text{PROG}} \subseteq \hat{B}$. Die Wahrscheinlichkeit, dass ein zufällig ausgewählter Täuschwert $f' \in \{0,1\}^{20} \backslash F_{\text{PROG}}$ in $\hat{B}$ liegt ist $\frac{|R|}{(2^{20}-N)} < \frac{|R|}{(2^{20})} < \frac{2^{12}}{(2^{20})} = 2^{-8} < 0,004$. Durch den Ausschluss ungültiger Täuschwerte (die nicht in $\hat{B}$ liegen) kann ein Angreifer seinen Suchraum bei einem Brute-Force-Angriff auf den Schlüssel κ stark einschränken, wie in Algorithmus 1 gezeigt. (Hier bezeichnet Π_n die Menge der Permutationen über $\mathbb{Z}_n$ und $\pi \in \Pi_n$ ist eine Permutation.)

Algorithmus 1 : Schlüsselsuchalgorithmus für verkleinerten Coffer

Input : I, $m := I \mid\mid Q \mid\mid \text{ID}_{\text{IMD}}$, $\mu := \text{MAC}_\kappa(I \mid\mid Q \mid\mid \text{ID}_{\text{IMD}})$, Coffer R, $\hat{B}$, $n := |I|$
Output : Schlüssel κ

```
for alle (f_0, f_1, ..., f_n - 1) ∈ B̂ do
    for alle π ∈ Π_n do
        κ' ← HASH(f_π(0) || f_π(1) ... || f_π(n-1))
        if MAC_κ' = μ then
            κ' ausgeben; halt
```

2.7.2 Adaptiver Angriff auf OPFKA

Rostami et al [Ros+13a] beschreiben auch einen adaptiven Angriff, der mittels wiederholter abgebrochener Kommunikation mit dem IMD einzelne Merkmalswerte durch Binärsuche ermitteln und so den Schlüssel berechnen kann.
Sei die Coffergröße $M = 4096 = 2^{12}$, $D = \{0,1\}^{12}$ die Menge der möglichen Werte für ein 12-Bit-Merkmal f und $B = \{H(s,f)\}_{f \in D}$ bezeichne den Wertebereich der Hashfunktion $H(s,\cdot)$, die die IPI-Messungen zu 20-Bit-Werten transformiert. Der Coffer enthalte nun die Zielmenge der Anwendung der Hashfunktion H auf alle möglichen Ausgangswerte von D, es gilt also $R = B$.
Der Angreifer konstruiert nun einen Coffer C wie folgt. R wird in zwei gleich große Mengen R_0 und R_1 (der Größe 2^{11}) aufgeteilt. Es wird ein Coffer C konstruiert, der R_0 und eine Teilmenge $R_1' \subseteq R_1$ (s.u.) enthält. Mit hoher Wahrscheinlichkeit sind mindestens d Merkmalswerte aus F^{IMD} in R_0 enthalten. Daher wird das IMD auf die Übertragung von C mit einer Reihe von Indizes I für jede Wahl von R_1' reagieren. Bei einer ersten Übertragung setzt der Angreifer nun $R_1' = R_1$. Mit hoher Wahrscheinlichkeit wird F^{IMD} mindestens einen Merkmalswert in R_1' mit dem Index i enthalten. Durch Rekursion auf die Hälften von R_1' und Beobachtung, ob $i \in I$ in der Antwort des IMD enthalten ist, kann der Angreifer eine binäre Suche durchführen und f_j^{IMD} mit $log_2|D| = 12$ Übertragungen lernen. Durch die Wahl verschiedener Anfangspartitionen (R_0, R_1) kann der Angreifer q Merkmalswerte in F^{IMD} lernen und sich erfolgreich als legitimer Programmer ausgeben.
Nach Rostami et al sind auch Angriffsvarianten mit einem kleiner Wert von M oder auch eine Parallelisierung des Angriffs möglich, bei der nach mehreren IMD-Merkmalswerten gleichzeitig gesucht wird. Der hier beschriebene Angriff setzt die Fähigkeit voraus, das IMD relativ schnell abzufragen, und wird ermöglicht, weil OPFKA keinen Drossel- oder Abschaltmechanismus enthält. Eine einfache Gegenmaßnahme besteht darin, dass das IMD nach einer fehlgeschlagenen Schlüsselvereinbarung mit einem Programmer Verbindungen verweigert, bis es einen neuen Satz von IPIs gesammelt hat. Dies birgt natürlich das Risiko von Denial-of-Service-Angriffen und verzögertem legitimem Zugriff auf das IMD, falls durch IPI-Messunterschiede das Protokoll abgebrochen werden muss.

2.8 Fazit

Die Security der Verwendung von physiologischen Signalen zur Schlüsselerzeugung und Authentifikation zwischen IMDs und mit diesen am oder im Körper befindlichen Geräten beruht auf der Grundannahme, dass eine Messung von PS nur bei direktem Körperkontakt (touch-to-access) möglich ist. Dies ist ein aktiver Forschungsbereich, in dem gezeigt werden konnte, dass eine Extraktion von PS aus Videosignalen prinzipiell möglich ist. So ermitteln Poh et al in [PMP11] die Herz- und Atemfrequenz, Tarassenko et al [Tar+14] Herz- und Atemfrequenz sowie Sauerstoffsättigung. Kwon et al [Kwo+12] extrahieren die Herzfrequenz mit Hilfe eines Smartphonevideos des Gesichtes. Weitere Arbeiten berechnen aus den Videodaten, die dann als iPPT (engl. imaging photoplethysmography) bezeichnet werden, auch noch Blutdruck und Sauer-

stoffsättigung [Zhu+22]. Eine Extraktion des EKG-Signals aus Videoaufnahmen war bisher nicht möglich. Diese Verfahren zeigen auf, dass die Annahme der Nichtmessbarkeit von PS aus der Entfernung auf wackligen Beinen steht.
Ein weiterer möglicher Angriffswinkel auf PS-basierte Verfahren ist die Qualität des Zufalls von PS. Dieser hängt unter anderem auch von der Auswahl der für den Zufall verwendeten Ausgangsdaten aus PS ab. So deklarieren Poon et al [PZB06] die untersten 4 Bits des IPI als „quasi“ gleichverteilt, Xu et al [Xu+11] schränken die Gleichverteilungsannahme auf die untersten 3 Bits ein. In [OPP20] analysieren die Autoren über 160000 EKGs und bestimmen die besten Bits des extrahierten IPI-Signals bzgl. Entropie. Sie zeigen, dass die vier niederwertigsten Bits (LSB) des IPI bzgl. Entropie nicht die beste Wahl sind.
Alle gerade genannten Arbeiten über den Einsatz des EKG-Signals nutzen die Daten der PhysioNet Herzdatenbanken[2] [Gol+00]. Chang et al führten hingegen direkte Messexperimente am menschlichen Körper durch und kommen zu dem Schluss, dass wegen der Messunterschiede des EKG-Signals an unterschiedlichen Stellen des Körpers dieses für den Einsatz als gemeinsames Geheimnis zwischen verschiedenen Geräten am Körper schlecht geeignet ist [Cha+12].
Die Extraktion von PS benötigt eine gewisse Zeit, um eine ausreichende Entropie der Ausgangsdaten für einen kryptographischen Schlüssel einer gewissen Länge (z.B. 128 Bit) sicherzustellen. Hier stehen Utility und Security im Widerspruch, da der Patient und eventuell auch der Arzt für die PS-Messung länger warten muss. Mit einer Messung von PS über mehrere Minuten wäre es möglich, zufällige sichere Schlüssel zu generieren. Beispielsweise wäre es bei einer Routinekontrolle eines Herzschrittmachers unpraktisch, wenn der Arzt den Programmer mehrere Minuten auf dem Brustkorb des Patienten halten und solange warten müsste, bis er die Herzschrittmacherdaten auslesen und diesen programmieren könnte.
Außerdem sind die Messdaten von PS, die von verschiedenen Geräten am bzw. im Körper gemessen werden nicht gleich, so dass hier entweder so lange gemessen werden muss bis beide Parteien durch Vergleich der Werte ausreichende Zufallsbits für die Schlüsselerzeugung erlangen oder die Verfahren setzen einen Fehlerkorrekturmechanismus ein wie bei den auf der Fuzzy Kryptoprimitive aufbauenden Verfahren. Hier stellt sich dann die Frage, inwieweit ressourcenbeschränkte Geräte wie IMD und implantierte Sensoren diese ausführen können. Die Komplexität der dabei verwendeten Fehlerkorrekturverfahren könnte in Abhängigkeit von der gewählten Codefamilie und den Parametern in einigen Fällen vom IMD bzw. Sensor zu viel Energie verbrauchen [Mar+16b].
In Tabelle 2.1 geben wir einen Überblick über alle PS-basierten Protokolle, die von einem IMD und Programmer verwendet werden können. Bei nicht übereinstimmenden Schlüsseln lassen viele Protokolle direkt weitere Zugriffe zu, obwohl dies eine Brute-Force-Attacke begünstigt. Die Messzeit der PS wäre prinzipiell ein Zeichen für die Praktikabilität des Verfahrens, wobei viele Papers diese einfach angeben, weil sie diese aus PS-Datenbanken übernommen haben. In der Praxis geht man davon aus, dass man PS ca. 30 Sekunden messen muss, um ausreichend Entropie zur Erzeugung von

[2] https://physionet.org/data/#ecg

128 Zufallsbits zu erreichen, wobei neuere Untersuchungen zeigen, dass dafür eher eine Messzeit von 60 Sekunden für eine ausreichende Entropie benötigt wird [Ort+19].

	verwendete physiol. Signale							
Paper (Name des Protokolls)	andere	IPI	EKG	PPG	Maßnahmen bei nicht übereinstimmenden Schlüsseln	Messzeit	#Samples	verwendetes Verfahren zur Korrektur von PS-Messfehlern
[CVG03] (BioSec)	✓	✗	✓	✗	keine	✗	✗	Fuzzy Commitment
[BSZ04]	HRV	✗	✗	✗	keine	✗	✗	Fuzzy Commitment
[BH07]	✗	✓	✗	✗	keine	? mit 1 kHz	✗	eigenes Fuzzy Commitment
[VBG08] (EKA)	✗	✗	✓	✗	keine	5 sec mit 135 Hz	625	#Messwerte groß genug bzw. mit Hashfkt. Wert auf Keylänge bringen
[Ven+08] (PKA)	✗	✗	✗	✓	keine	12,8 sec mit 60 Hz	768	Fuzzy Vault
[Bao+08]	✗	✗	✓	✓	keine	40 sec mit 1 kHz	✗	Bit-Shift
[Mia+09]	✗	✗	✓	✗	keine	3 sec mit 360 Hz	1080	Fuzzy Vault
[Ven+10] (PSKA)	✗	✗	✓	✓	keine	gleiche Werte wie PKA für PPG gleiche Werte wie EKA für EKG		Fuzzy Vault
[Yao+10] (ESKE)	✗	✗	✓	✗	keine	✗	✗	Fuzzy Commitment
[Yao+11] (Bioscrypt)	✗	✗	✓	✗	gibt es nicht durch die Konstruktion des Verfahrens	✗	✗	Messfehler werden bei Schlüsselberechnung komplett herausgerechnet
[Xu+11] (IMDGuard)	✗	✗	✓	✗	gibt es nicht durch die Konstruktion des Verfahrens	✗	✗	PS-Werte solange messen bis genug Schlüsselbits ausgelesen wurden ECC: Bitparität
[Raj+12]	bel. PS	(✓)	(✓)	(✓)	keine	4 sec mit 125 Hz	256 (nach FFT)	eigenes Fuzzy Vault Verfahren
[Zha+12] (ECG-IJS)	✗	✗	✓	✗	keine	4 sec mit 120 Hz	256 (nach FFT)	Reed-Solomon-Code
[Hu+13] (OPFKA)	✗	✓	✗	✓	keine	ca. 12,8 sec für PPG ca. 60-90 sec für IPI	90	#Messwerte groß genug
[Ros+13b] (H2H)	✗	✓	✗	✗	warte einen PS-Lesezyklus	ca. 15 sec	✗	Neyman-Pearson-Test der Werte
[Zhe+14b] (ESDS)	✗	✗	✓	✗	keine	? mit 125 Hz	✗	verbesserter Fuzzy Vault
[Zhe+15] (EDE)	✗	✗	✓	✗	gibt es nicht durch die Konstruktion des Verfahrens	✗	✗	PS-Werte solange messen bis genug Schlüsselbits ausgelesen wurden ECC: BCH und Bitparität
[Zag+15] (ELPA)	✗	✗	✓	✗	keine	✗	✗	BCH (Bose-Chaudhuri-Hocquenghem-Code) in Kombination mit LPC (Linear Prediction Coding)
[Mar+16a]	✗	✓	✗	✗	direkte Wiederholung des Protokolls	✗	✗	Fuzzy Extractor
[KA18] (SGenP)	✗	✓	✗	✗	direkte Wiederholung des Protokolls	✗	90	#Messwerte groß genug bzw. Protokoll wiederholen
[Bai+18]	✗	✗	✓	✗	gibt es nicht durch die Konstruktion des Verfahrens	4 sec + x	✗	offener Broadcast der Messwerte
[KM19] (ESKG)	✗	✗	✓	✗	Abbruch des Protokolls	1 Minute	✗	keines
[KSL19] (SKA-PS)	✗	✓	✗	✗	Abbruch des Protokolls	? mit 125 Hz	✗	#Messwerte groß genug
[Sey19] (SKA-PSAR)	✗	✓	✗	✗	direkte Wiederholung des Protokolls	? mit 125 Hz	✗	#Messwerte groß genug
[Lin+19] (H2B)	✗	✓	✗	✗	gibt es nicht durch die Konstruktion des Verfahrens	✗	✗	PS-Werte solange messen bis Fehlerrate in PS-Bits unter Schranke ECC durch compressive sensing [Bar07]
[Zhe+19] (F2H)	Fingerabdruck	✗	✗	✗	keine	✗	✗	Minutia Cylinder-Code (MCC) [CFM10]
[Bel+19]	Fingerabdruck	✗	✓	✗	keine	✗	✗	Senden verarbeiteter Messwerte als Security Token
[BZS21] (MEDISCOM)	BCG	✓	(✓)	✗	keine	✗	✗	nicht berücksichtigt
[Zha+21a] (H2K)	✗	✓	✗	✓	direkte Wiederholung des Protokolls	✗	✗	Secure Sketch

Tabelle 2.1: Authentifikation/Schlüsselvereinbarung mit physiologischen Signalen

3 Verbesserung der Sicherheit von IMDs durch externes Gerät

Eine weitere Möglichkeit, einen IMD gegen Angriffe zu schützen ist der Einsatz eines externen Geräts, das alle Geräte, die mit dem IMD kommunizieren wollen, authentifiziert. Solch ein Gerät wurde erstmals 2008 in [DFK08] von Denning et al vorgeschlagen. Die Autoren definieren für solch ein mit dem IMD gekoppeltes externes Gerät eine neue Klasse von externen Schutzgeräten, die sie Kommunikationscloaker nennen. Im Folgenden werden wir diese Geräte kürzer mit Cloaker bezeichnen.
Ein Cloaker soll über mehr Rechenleistung als das IMD verfügen und (natürlich) über Funk kommunizieren können. Bei Einsatz eines Cloakers ignoriert das IMD nun alle direkten Kommunikationsversuche von externen Geräten außer dem Cloaker[1]. Da aber immer Programmer existieren können, die das Cloaker-Konzept nicht integriert haben, muss nun das Problem gelöst werden, dass medizinisches Personal im Notfall direkt auf das IMD mit einem „naiven“ Programmer zugreifen können muss (und somit die Safety gewährleistet ist), der sich gegenüber dem Cloaker nicht authentifizieren kann. Als Lösung präsentieren Denning et al das Fail-Open-Konzept, bei dem im Notfall eine unverschlüsselte direkte Kommunikation mit dem IMD möglich ist.
Fail-Open wird von den Autoren im Falle des Cloakers so definiert, dass dieser Security für das IMD gewährleistet solange er vom Patienten getragen wird, das IMD aber mit allen externen Programmern kommuniziert, sobald der Cloaker entfernt wird. Hierbei muss der Patient den Cloaker dauerhaft mit sich führen, um Security zu gewährleisten. Während normaler Routineuntersuchungen erlaubt der Cloaker vorab autorisierten Programmern den Zugriff. Im Notfall können Mediziner den Cloaker entfernen und erhalten somit direkten Zugriff auf das IMD.
Denning et al wollen nicht das genaue Design von Cloakern vorgeben, sondern ein Konzept für die Entwicklung von Cloakern vorstellen. Ein Cloaker-basiertes IMD-System besteht aus den Komponenten IMD, Programmer und Cloaker. Bei der Entwicklung eines solchen Systems benötigt man nun Kommunikationsprotokolle für die Kommunikation zwischen IMD, Programmer und Cloaker bei Anwesenheit des Cloakers und ein (Sicherheits-)Konzept für den Nachweis der Ab-/Anwesenheit des Cloakers. Die (unverschlüsselte) direkte Kommunikation zwischen IMD und Programmer bei Abwesenheit des Cloakers nutzt bereits bestehende Protokolle.
Die Kommunikation zwischen Cloaker und IMD sollte verschlüsselt und authentifiziert stattfinden und Maßnahmen (wie z.B. Counter) verwenden um Replay- und Reordering-Angriffe zu verhindern. Da Cloaker und IMD für den dauerhaften gemeinsamen Einsatz konzipiert schlagen die Autoren die Verwendung von symmetrischen

[1] deshalb haben Denning et al den Begriff Cloaker (engl. Tarnkappe) ausgewählt, da dieser bei Anwesenheit das IMD für externe Geräte praktisch unsichtbar macht

Verschlüsselungsverfahren vor.
Für die Initialisierung der Kommunikation zwischen Programmer und IMD über den Cloaker bestehen folgende zwei Möglichkeiten:
Das IMD hört auf Sitzungsanfragen des Programmers und fragt bei einer Sitzungsanfrage den Cloaker als Orakel nach der Verifikation der Authentizität des Programmers (Ansatz 1, siehe Abbildung 3.1). Die Autoren befürchten hier einen Battery-Draining-

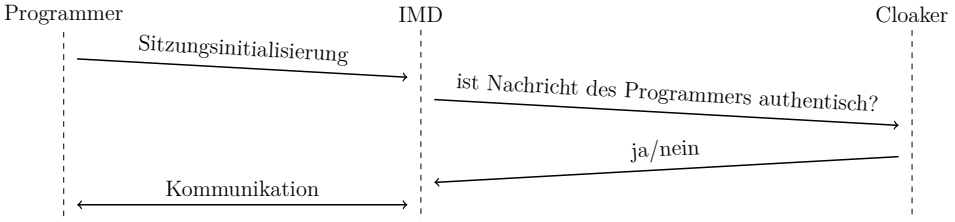

Abbildung 3.1: Ansatz 1 für die Initialisierung der Kommunikation von Programmer und IMD bei Anwesenheit des Cloakers

Angriff auf die Batterie des IMD und bevorzugen Ansatz 2, bei dem der Cloaker zuerst die Autorisierung des Programmers prüft, bevor eine Kommunikation zwischen IMD und Programmer stattfindet (siehe Abbildung 3.2).
Da die Batterie des Cloakers austauschbar ist und dieser über größere Rechenleis-

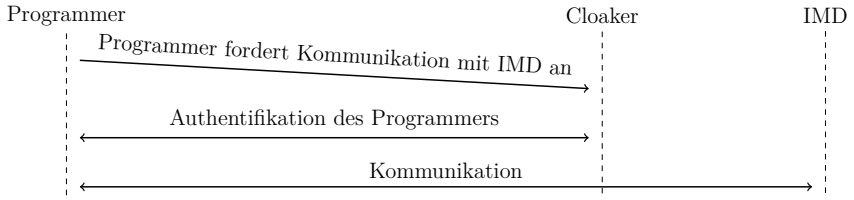

Abbildung 3.2: Ansatz 2 für die Initialisierung der Kommunikation von Programmer und IMD bei Anwesenheit des Cloakers

tung als das IMD verfügt kann rechenintensive Public Key Kryptographie eingesetzt werden. Die Autoren schlagen als mögliche Realisierung vor, dass auf dem Cloaker die öffentlichen Schlüssel der autorisierten externen Programmer gespeichert werden, wobei die Programmer dann die dazugehörigen privaten Schlüssel in sicherer Hardware speichern sollten. Dies hat den Nachteil, dass sobald das behandelnde Krankenhaus einen neuen Programmer kauft, alle Cloaker-Geräte alle Patienten ein Schlüsselupdate für diesen neuen Programmer erhalten müssen.
Für die Kommunikation zwischen Programmer und IMD nach der Authentifikation des Programmers erwägen Denning et al zwei Möglichkeiten: Bei Ansatz A (siehe Abbildung 3.3) fungiert der Cloaker als Proxy zwischen Programmer und IMD. Dies erlaubt es ihm die ganze Kommunikation zwischen Programmer und IMD zu loggen, welche dann für forensische und Analysezwecke zur Verfügung steht. Eine vom Cloaker unbemerkte Kommunikation zwischen externem Gerät und IMD ist hier unmöglich.
Ansatz B ermöglicht eine direkte Kommunikation zwischen Programmer und IMD, indem der Cloaker symmetrische Sitzungsschlüssel an Programmer und IMD verteilt (siehe Abbildung 3.4). Ein möglicher Vorteil wäre hier eine niedrigere Latenz und dass

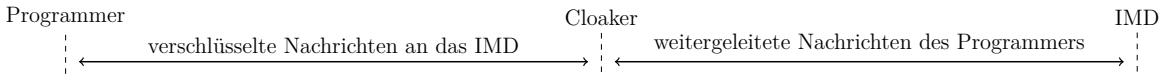

Abbildung 3.3: Ansatz A für die Kommunikation von Programmer und IMD nach Authentifikation

bei Entfernung des Cloakers die Kommunikationssitzung zwischen Programmer und IMD nicht unterbrochen wird.
Ein Angriffspunkt auf das Cloaker-Konzept ist die Vortäuschung der Abwesenheit

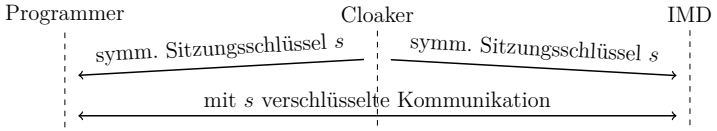

Abbildung 3.4: Ansatz B für die Kommunikation von Programmer und IMD nach Authentifikation

des Cloakers. Denning et al betrachten zwei Ansätze für den Anwesenheitsnachweis des Cloakers: einen zustandsloser Nachweis,bei dem jedesmal, wenn das IMD eine Kontaktaufnahme eines Programmers wahrnimmt, dieses die Anwesenheit des Cloakers bei diesem abfragt. Dieser Ansatz hat den Vorteil, dass man nicht die ganze Zeit Keep-Alive-Nachrichten verschicken muss.
Hier muss man verhindern, dass ein Angreifer selektiv die Nachrichten vom Cloaker zum IMD jammt. Hier könnte als Abwehrmaßnahme die Anfrage des IMD an den Cloaker an zufälligen Zeitpunkten stattfinden. Seitens des Cloakers könnte eine Abwehr durch Auslösen eines Alarmsignals bei Detektion eines Jamming-Signal stattfinden. Andere Abwehrmöglichkeiten des Cloakers wären Änderung der Übertragungsstärke/Übertragungscharakteristik oder ein aktives Jammen der Kommunikation des Angreifers mit dem IMD.
Bei einem zustandsbehafteten Nachweis der Anwesenheit des Cloakers speichert das IMD den Zustand der Anwesenheit des Cloakers und updatet diesen je nach Eintreffen oder Ausbleiben von Keep-Alive Nachrichten. Die Autoren betrachten folgende Keep-Alive-Nachrichten-Varianten: Das IMD fragt den Cloaker nach dessen Anwesenheit und erhält danach dessen bestätigende Antwort oder der Cloaker sendet Keep-Alive-Nachrichten an das IMD.
Egal, welche der beiden Varianten man wählt stellt das Zeit-Intervall zwischen den Keep-Alive-Nachrichten immer einen Kompromiss zwischen Safety und Batterieentladung da: kürzere Zeitintervalle führen zu einem schnelleren Fail-Open des IMD im Notfall während ein längeres Zeitintervall die Batterie des IMD schont.
Eine im Notfall gefährliche Situation, die die Safety des IMDs aushebelt ist die, falls der Cloaker vom Patienten getragen wird und dieser vom Notfallpersonal nicht gefunden wird, da er z.B. einer vom Patienten mitgeführten Tasche oder in einer der Taschen seiner Kleidung verborgen ist. Eine Lösung wäre hier festzulegen, dass der Cloaker an einem festen Ort am Patienten, z.B. als Armband oder um den Hals getragen wird.
Verwandt hiermit ist eine Angriffsszenario, bei dem der Angreifer unbemerkt vom Patienten den am Körper getragenen Cloaker zerstört. Dieser wähnt sich die ganze

Zeit geschützt, obwohl jetzt das IMD dauerhaft im Fail-Open-Modus ist.
Im Jahr 2011 schlugen Xu et al. [Xu+11] IMDGuard vor. IMDGuard dient als Authentifizierungsproxy zwischen IMD und Programmer, wobei initial IMD und IMDGuard einen gemeinsamen Schlüssel durch Messen der EKG-Signale des Patienten vereinbaren. IMDGuard verwendet ein sog. defensives Jamming zur Abwehr von Angriffen, die versuchen, vor dem IMD die Anwesenheit des Cloakers zu verbergen, um ein Fail-Open zu erzwingen. Eine genauere Beschreibung der Protokolle und Abwehrmaßnahmen findet man im Kapitel zu IMDGuard.
Einige externe Geräte zum Schutz des IMD verwenden nur physikalische Maßnahmen zu dessen Schutz und können ohne Veränderung des IMD und ohne dessen Kenntnis eines solchen Gerätes eingesetzt werden. Diese Geräte fallen also nur im erweiterten Sinne unter den Cloaker-Begriff, da sie die mit dem IMD kommunizierenden Geräte nur aufgrund ihrer physikalischen Eigenschaften „authentifizieren“. Diese Geräte stellen wir in Abschnitt 3.2 vor.
2013 stellten Zhang et al in [ZRJ13] ein externes Gerät namens „Medmon“ vor, das auf der Frequenz des IMD lauscht und externe Signale jammt, wenn diese bestimmte Abweichungen des physikalischen Signals oder des Verhaltens des Programmers aufweisen. Eine genauere Beschreibung bieten wir in Abschnitt 3.3.
Erwähnen wollen wir hier noch einen Ansatz, der unsere Einschränkung auf Verfahren, die direkt und ohne öffentliche Netzwerke mit dem IMD kommunizieren, verletzt:
Im Jahr 2018 schlugen Fu et al. [Fu+18] ein Physiological Obfuscated Keys (POKS) verwendetes Verfahren vor, bei dem allerdings auf einer IC-Karte (engl. integrated circuit) schon im Voraus ein geheimer Schlüssel gespeichert ist und dauerhaft Zugriff auf einen Server im Krankenhaus gewährleistet sein muss.

3.1 IMDGuard

In [Xu+11] wird zum ersten Mal ein umfassendes Security-Protokoll für das in [DFK08] vorgeschlagene **Fail-Open**-Konzept entwickelt, welches das externe Cloaker-Gerät als **Guardian** bezeichnet. Die Autoren konzipieren den Guardian als tragbares armbanduhrähnliches Gerät, wobei bei IMDGuard sowohl das IMD als auch der Guardian EKG-Signale erfassen können müssen.
Der Guardian erfüllt drei grundlegende Funktionen:

1. Kontrolle über den Modus des IMDs (**regulär** oder **Notfall**). Wenn der Patient den Guardian trägt, sollte das IMD im regulären Modus sein. In diesem wird der Programmer vom Guardian authentifiziert und IMD und Programmer erhalten vom Guardian danach einen Sitzungsschlüssel.
   Bei Abwesenheit des Guardian soll das IMD in den Notfallmodus schalten. Somit kann ein Arzt im Notfall den Guardian entfernen um sofort uneingeschränkten Zugriff auf das IMD zu erhalten.

2. Authentifikation des Programmer für das IMD, was die Batterie des IMD schont und dessen Komplexität verringert.

3. Abwehr von Spoofing-Angriffen auf das IMD: Wenn ein Angreifer das IMD durch Jamming aller Nachrichten des Guardian in den Notfallmodus schalten will verhindert der Guardian die Kommunikation des IMD mit dem Programmer durch Jammen der Nachrichten. Da der Angreifer über beliebig starke Sender verfügen könnte, kann der Guardian diesen nicht jammen, aber für das Jammen des IMD-Signals reicht die Energie des Guardian - auch da ihm Timing und Frequenz des IMD-Signals bekannt sind.

Diese Funktionen werden durch die folgenden Protokolle realisiert:

1. ein initiales Pairing-Protokoll zwischen IMD und Guardian durch Analyse des EKG-Signals des Patienten (siehe Kapitel 2.5),

2. ein Access Control Protokoll für die Authentifikation vom Programmer gegenüber dem Guardian (siehe Kapitel 3.1.1) *und*

3. ein Kommunikationsprotokoll zwischen Programmer und IMD, das im Notfall (bei Abwesenheit des Guardian) einen direkten unverschlüsselten Zugriff auf das IMD erlaubt (siehe Kapitel 3.1.3).

Im Angreifermodell von Xu et al gewinnt ein Angreifer, wenn er in Anwesenheit des Guardian unbemerkt eines der folgende Ziele erreicht: Programmieren des IMD oder Auslesen von Daten des IMD. Sie treffen außerdem Annahmen bzgl. des Angreifers:

1. Er führt keine DoS-Attacken aus. Dies wird angenommen und als zusätzliche Aufgabe für darauf aufbauende Entwürfe betrachtet.

2. Er kann nicht unbemerkt das EKG-Signal des Patienten erfassen. Dies gründet auf der Annahme, dass zum Auslesen des EKG die Messvorrichtung am oder im Körper des Patienten lokalisiert sein muss.

3. Er kann den Guardian dem Patienten nicht unbemerkt entwenden, da dieser ja vom Patient direkt am Körper getragen wird.

4. Im (medizinischen) Notfall existiert kein Angreifer, da der Zugriff auf das IMD für medizinisches Notfallpersonal (Safety) in diesem Fall Vorrang hat.

Im folgenden werden die Kommunikationsprotokolle zwischen IMD, Guardian und Programmer vorgestellt. Die in den Abbildungen verwendeten Bezeichnungen werden in Abbildung 3.5 erläutert. Das Schlüsselvereinbarungsprotokoll zwischen Guardian und IMD durch Auslesen des EKGs wird in Abschnitt 2.5 beschrieben.
Wir setzen hier voraus, dass der Guardian eine Liste der legitimen Programmer und ihre zugehörigen Public Keys enthält. Diese kann z.B. von einer legitimierten Stelle im Krankenhaus eingespielt werden.

Das zwischen Guardian und Programmer eingesetzte Public Key Verfahren verwendet Kryptographie auf elliptischen Kurven mit der Kurve SECP160R1 [WSL06].

IMD	das IMD
GUARD	der Guardian
PROG	der Programmer
$\text{NONCE}[i]_{\text{GERÄT}}$	die $i$te Nonce, die von GERÄT erzeugt wurde mit GERÄT $\in \{\text{IMD}, \text{GUARD}, \text{PROG}\}$
$\text{HASH}(\cdot)$	eine standardisierte kryptographische Hashfunktion, z.B. SHA-1
SK_{IMD}	zwischen IMD und Guardian schon vereinbarter symmetrischer geheimer Schlüssel
$\text{PK}_{\text{GERÄT}}$	der öffentliche Schlüssel von GERÄT mit GERÄT $\in \{\text{GUARD}, \text{PROG}\}$
$\text{SK}_{\text{GERÄT}}$	der geheime Schlüssel von GERÄT mit GERÄT $\in \{\text{GUARD}, \text{PROG}\}$
SSK_i	der $i$te temporäre symmetrische Session-Key für IMD und Programmer
t_i	Wartezeit des Timers T_i
ID	die ID des IMDs

Abbildung 3.5: Abkürzungen für die IMDGuard-Protokolldiagramme

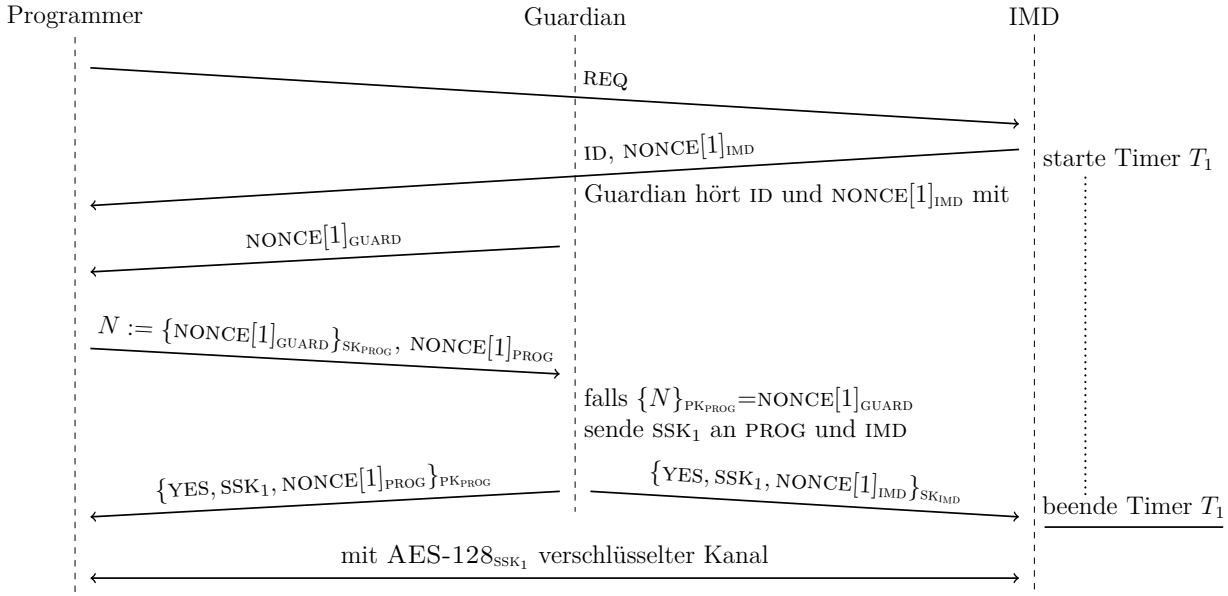

Abbildung 3.6: erfolgreiche Etablierung eines verschlüsselten Kanals zwischen Guardian und Programmer

3.1.1 Authentifikationsprotokoll des Programmers gegenüber dem Guardian

Dieses Protokoll wird in Abbildung 3.6 dargestellt. Dabei wacht das IMD in regelmäßigen Abständen auf, um zu prüfen, ob eine Anfrage vom Programmer vorliegt. Bei Vorliegen einer Anfrage sendet das IMD seine ID und eine Nonce $\text{NONCE}[1]_{\text{IMD}}$ an den Programmer zurück. Diese soll Replay-Attacken verhindern. Dann startet das IMD einen Timer, nach dessen Ablauf das IMD davon ausgeht, dass der Guardian nicht vorhanden ist (dieses Protokoll wird in Kapitel 3.1.2 behandelt). Bei Anwesenheit des Guardian hört dieser die vom IMD gesendete ID und die Nonce $\text{NONCE}[1]_{\text{IMD}}$ mit und beginnt daraufhin mit der Authentifizierung des Programmers.
Er sendet die Nonce $\text{NONCE}[1]_{\text{GUARD}}$ an den Programmer. Dieser schickt als Signatur diese Nonce $\text{NONCE}[1]_{\text{GUARD}}$ verschlüsselt mit seinem privaten Schlüssel SK_{PROG} und eine eigene Nonce $\text{NONCE}[1]_{\text{PROG}}$ an den Guardian zurück. Dieser prüft mit dem Public Key PK_{PROG} die Gültigkeit der Signatur. Wenn diese gültig ist erstellt er einen

temporären Sitzungsschlüssel SSK_1. Anschließend wird sowohl an den Guardian als auch an das IMD eine verschlüsselte Nachricht gesendet, die sowohl die erfolgreiche Authentifizierung der Kommunikation zwischen den beiden Geräten (YES), als auch den symmetrischen Sitzungsschlüssel SSK_1 und die zugehörige Nonce für jedes Gerät enthält. Ab jetzt kommunizieren Programmer und IMD direkt unter Verwendung von AES-128.
Falls die Signatur des Programmers ungültig ist informiert der Guardian das IMD (mit $\{\text{NO}, \text{NONCE}[1]_{\text{IMD}}\}_{\text{SK}_{\text{IMD}}}$) und den Programmer über die fehlgeschlagene Authentifikation des Programmers (siehe Abbildung 3.7).

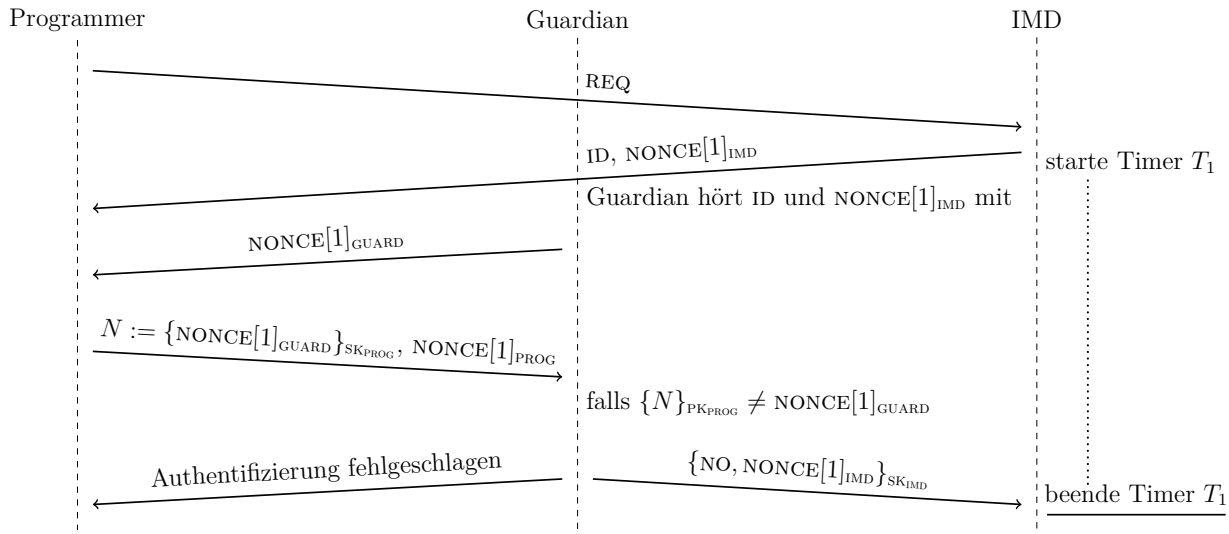

Abbildung 3.7: Protokoll Guardian lehnt Programmer ab

3.1.2 Protokoll, falls der Guardian nicht vorhanden ist (Notfallprotokoll)

Falls im Authentifizierungsprotokoll des Programmers gegenüber dem Guardian (siehe Abbildungen 3.6 und 3.7) der Guardian nicht antwortet tritt beim IMD nach der Zeit t_1 der Timeout des Timers T_1 ein und dieser schickt nun eine Nonce $\text{NONCE}[2]_{\text{IMD}}$ an den Programmer, wartet danach die Zeit t_2 und schickt danach eine weitere $\text{NONCE}[3]_{\text{IMD}}$. Der Programmer schickt nun die geXORten Noncen gehasht an das IMD zurück. Falls dieser Wert korrekt ist schaltet das IMD in den Notfallmodus und kommuniziert nun mit dem Programmer unverschlüsselt (siehe Abbildung 3.8). Mit dem zweimaligen Senden einer Nonce soll ein Spoofing-Angriff verhindert werden (siehe Kapitel 3.1.3).

3.1.3 Resistenz des Protokolls gegen Spoofing-Attacken

Der Angreifer kann versuchen, die Kommunikation zwischen IMD und Guardian zu stören und so das IMD in den Notfallmodus zu versetzen. Der Angreifer sendet also eine REQ-Nachricht an das IMD und direkt danach ein Jamming-Signal (roter Bereich in Abbildung 3.9), damit keine Kommunikation zwischen IMD und Guardian möglich

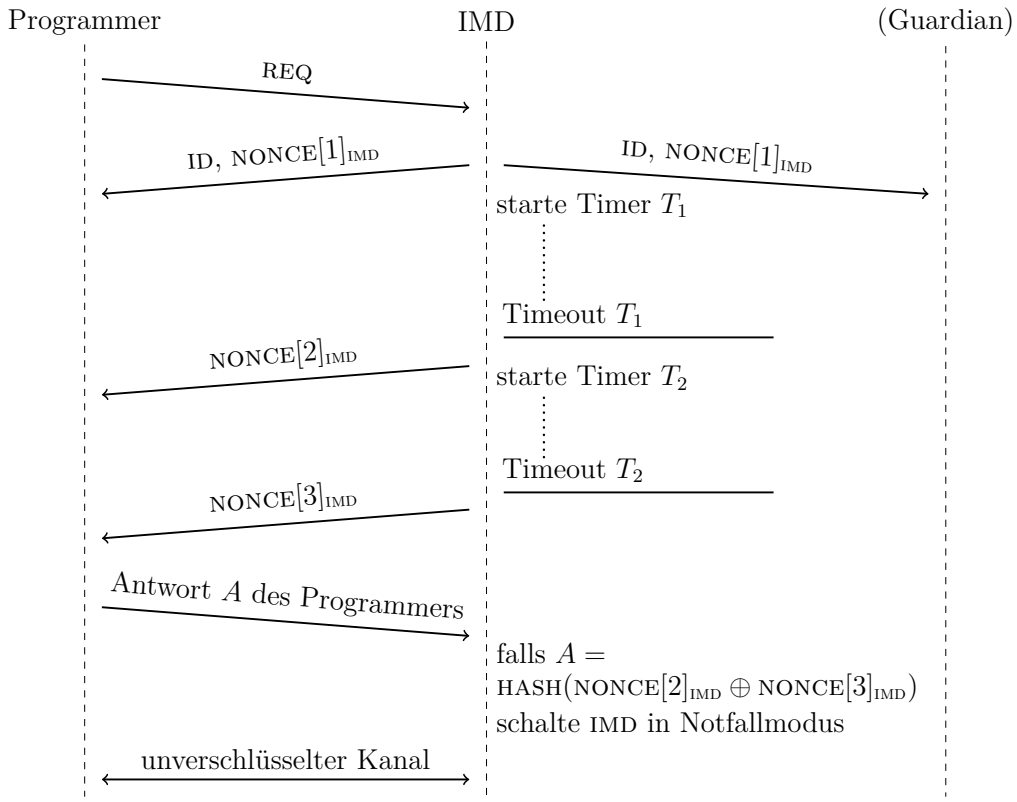

Abbildung 3.8: Notfallprotokoll, falls Guardian nicht vorhanden ist

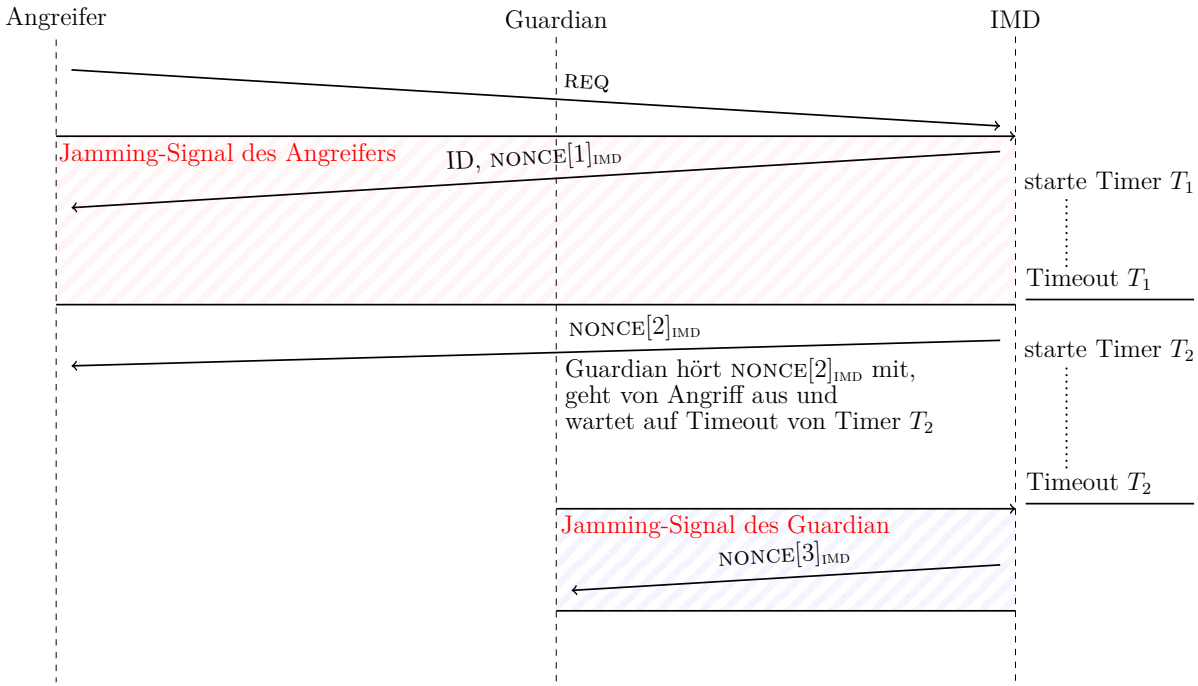

Abbildung 3.9: Abwehr eines Spoofing-Angriffes bei IMDGuard

ist und wartet auf Ablauf des Timers T_1 des IMDs. Währenddessen sendet das IMD als Antwort seine ID und eine Nonce $\text{NONCE}[1]_{\text{IMD}}$ an den vermeintlichen Programmer zurück welche aufgrund des Jamming-Signals des Angreifers nicht vom Guardian bemerkt wird. Nun ist das IMD im Notfallmodus und versucht einen unverschlüsselten Kanal zum Angreifer (den das IMD für einen Programmer hält) aufzubauen. das IMD sendet also seine zweite Nonce $\text{NONCE}[2]_{\text{IMD}}$ an den Angreifer, was der Guardian mithört. Dieser geht nun von einem Angriff aus und wartet bis der Timer T_2 des IMD abgelaufen ist und jammt dann dessen Signal (die Nonce $\text{NONCE}[3]_{\text{IMD}}$) an den Angreifer (blauer Bereich in Abbildung 3.9). Wir stellen das Jamming-Signal und die vom IMD gesendete dritte Nonce nur im Bereich zwischen Guardian und IMD dar, weil die Nachricht des IMD durch das Jamming-Signal des Guardians nicht über diesen hinaus kommt. das IMD erhält dann vom Angreifer keine gültige Antwort auf die dritte Nonce und geht wieder in den Normalmodus.
Der Guardian jammt deshalb das Signal des IMD, weil nicht davon ausgegangen werden kann, dass die Signalstärke des Guardian stark genug ist, um die Antworten des Angreifers zu jammen. Der Guardian kennt außerdem die exakte Laufzeit t_2 des Timers T_2 des IMDs und kann somit ein exakt getimetes kurzes Signal zum Jammen verwenden was seinen Energieverbrauch gering hält.

3.1.4 Kritik

Positiv anzumerken ist, dass durch den Einsatz des Schlüsselaustauschprotokolls über EKG-Signal zwischen IMD und Guardian jederzeit ein Rekeying möglich ist. Die Protokolle werfen allerdings Fragen auf bzw. beinhalten möglicherweise folgende Schwachstellen:
möglicher Angriff im Notfall: Im Notfall könnte ein unbemerkt am Patienten angebrachtes Gerät mit höherer Frequenz als das IMD Noncen senden und somit den Zugriff auf das IMD mit einem legitimen Programmer verhindern.
unnötiges Hashen: Im Notfallprotokoll schickt der Programmer die geXORten Noncen gehasht zurück - welche Funktion das Hashen hat ist hier unklar.
möglicher Angriff auf Spoofing-Abwehrprotokoll: Der Angreifer sendet sofort nach dem Jammen eine Nonce, die vom Guardian dann als Nonce des IMD interpretiert wird und dessen Timer verschiebt. Hier besteht für den Angreifer die Chance, dass das Jamming-Signal des Guardian beginnt, nachdem das IMD die zweite Nonce und dieses Signal endet bevor das IMD die dritte Nonce schickt. Falls der Angreifer erfolgreich ist, kann es das IMD nun in den Notfallmodus schalten.

3.2 Geräte mit ausschließlich physischem Schutz

Gollakota et al. [Gol+11] schlugen ein externes Gerät namens Shield zur Autorisierung von externen Geräten vor, welches als Proxy zum IMD fungiert.Hier gibt es im IMD keine Erfassung der Anwesenheit des externen Geräts, die Schutzwirkung funktioniert ohne jede Veränderung eines implantierten IMDs. Der Schutz des IMD geschieht rein physikalisch durch Jammen aller Nachrichten, die vom IMD gesendet werden. Das Shield-Gerät fungiert als Störsender und Empfänger und ist so aufgebaut, dass nur es

selbst Nachrichten des IMDs empfangen kann bei gleichzeitigem Jammen des IMD-Signals. Dies wird durch die Kopplung von Empfangsantenne mit der Sendeantenne, die das Jamming-Signal sendet, erreicht. Da Shield die Signale unverschlüsselt und ohne Störsignal an das IMD schickt, können diese natürlich mitgehört werden.
Diese Sicherheitslücke will [Kul17] schließen und schlägt einen Sicherheitsgürtel mit Bezeichnung „Secure Belt“ vor, der drei Störsender mit Empfänger und drei Richtantennen enthält, die an optimalen Stellen angebracht sind, um die Nichtabhörbarkeit zu gewährleisten. Alle Sendungen der Richtantennen sind punktgenau auf das IMD gerichtet, so dass Sendungen vom Gürtel an das IMD nicht abgehört werden. Die Sender und Antennen sind mit einem Steuergerät am Gürtel verbunden. Dieses lässt Sendungen an das IMD zeitlich zufällig zwischen den Richtantennen umschalten, was vom IMD nicht bemerkt wird. Wenn das IMD Signale an den Sicherheitsgürtel sendet werden dessen Signale zeitlich zufällig von den verschiedenen Störsendern gejammt und gleichzeitig von allen empfangen. Mit der MRC-Technik (engl. Maximum Ratio Combining) wird dann das Störsignal im Steuergerät am Gürtel herausgerechnet. Diese MRC-Technik ermöglicht den Empfang schwächerer Signale was es dem IMD gestattet mit geringerer Leistung zu senden. Dies bietet mehr Sicherheit gegen Eavesdropping und kann auch die Batterielebensdauer des IMD erhöhen. [Kul17] konzentriert sich vollständig auf das physikalische Design des Gürtels; es fehlt eine Beschreibung, wie die Kommunikation zwischen Programmer und Secure Belt ablaufen soll.
2019 stellte Kulac in [Kul19] einen Verbesserung des Secure Belt vor: das Security Jacket. Dies ist eine am Körper getragene Jacke, die als Faradayscher Käfig fungiert und auf der ein Gitternetz an kombinierten Störsender-Empfängern angebracht ist. Hier wird nun wie beim Secure Belt an zufälligen Zeitpunkten von den Sender-Empfängern jeweils zu mehreren gleichzeitig gesendet und Jamming-Signale ausgesendet, wobei die Jamming Signale so gewählt sind, dass sie sich genau beim IMD-Empfangsmodul auslöschen. Beim Senden von Signalen an das IMD wird Beamforming eingesetzt, so dass das Signal vom Security Jacket genau am Ort des IMD maximale Signalstärke hat. Durch den Einsatz von einer größeren Anzahl an Sendern und Empfängern kann die Sendeleistung sowohl des Security Jackets als auch des IMD verringert werden. Das Security Jacket ist durch die Notwendigkeit dauerhaft diese Jacke tragen zu müssen (u.a. auch im Sommer oder in Innenräumen) eher ein theoretisches Konzept.

3.3 Medmon

2013 stellten Zhang et al in [ZRJ13] ein externes Gerät namens „Medmon“ vor, das auf der Frequenz des IMD lauscht und externe Signale jammt, wenn diese bestimmte Abweichungen des physikalischen Signals oder des Verhaltens von Geräten, die Nachrichten an das IMD senden, aufweisen. Durch die Messung der physikalischen Werte RSSI (engl. received signal strength indicator), TOA (engl. time of arrival), DTOA (engl. differential time of arrival) und AOA (engl. angle of arrival) werden Richtung und Winkel des eintreffenden Signals gemessen. Als Verhaltenswerte werden Programmierwerte für das IMD und deren Veränderung über die Zeit, sowie deren Häufigkeit und Zeitpunkt der Programmierung betrachtet.

Für all diese Werte werden Grenzwerte festgelegt, bei deren Überschreitung Medmon je nach Einstellung entweder Alarm gibt oder das Signal gleich jammt.
So betrachten die Autoren das Beispiel einer implantierten Insulinpumpe mit implantiertem Glukosesensor und externem Programmiergerät für den Patienten. Hier sendet der Glukosesensor regelmäßig Glukosedaten an die Insulinpumpe, die dann aufgrund dieser Werte entsprechend Insulin abgibt. Hier wäre ein außerhalb des Zeitrasters ankommendes Wertsignal des Glukosesensors ein Hinweis auf einen Angriff.
Die Festlegung von physikalischen Grenzwerten findet in einer Trainingsphase statt, bei der der Patient das Gerät in seinem täglichen Alltag trägt. Hier werden jegliche an das IMD gerichtete Signale als legitime Signale gewertet und daraus dann die Grenzwerte für RSSI, TAO, DTAO und AOA für Medmon ermittelt. Die Grenzwerte für die Verhaltensprüfung werden manuell einprogrammiert.

3.4 Fazit

Alle in diesem Kapitel vorgestellten externen Geräte ermöglichen in Notfallsituationen den Zugang zum IMD durch ihr Fail-Open Entwurfsprinzip: durch einfaches Entfernen des externen Geräts ist ein unverschlüsselter direkter Zugriff auf das IMD möglich. Dies ist allerdings auch ein Angriffspunkt und eine Möglichkeit des unbeabsichtigten Fail-Open, falls ein Patient vergisst, das externe Gerät mit sich zu führen. Ein Diebstahl oder auch die (vom Patienten) unbemerkte Zerstörung des Geräts wäre auch denkbar. Im Folgenden fassen wir die in diesem Kapitel eingesetzten Eigenschaften der Kommunikationsprotokolle mit den externen Geräten und die eingesetzten Schutzmaßnahmen in Tabelle 3.1 zusammen.

Tabelle 3.1: Übersicht über die Protokolleigenschaften der externen Schutzgeräte

Paper (Name des ext. Geräts)	Protokoll-variante $1/2^{\alpha}$	fungiert Cloaker-gerät als Proxy?	wie überprüft das IMD Anwesenheit des Cloakers?	Schutzmaßnahmen des ext. Geräts
[Xu+11] (IMDGuard)	2	nein	zustandslos	Schlüsselvereinbarungsprotokoll + Authentifikationsprotokoll + Jamming
[Gol+11] (Shield)	2	ja	IMD weiß nichts von ext. Gerät	Jamming beim Empfang von allen IMD-Signalen
[Kul17] (Secure Belt)	2	ja	IMD weiß nichts von ext. Gerät	Richtantennen +Jamming
[Kul19] (Security Jacket)	2	ja	IMD weiß nichts von ext. Gerät	Jamming +Richtantennen mit Beamforming
[ZRJ13] (Medmon)	Medmon ist für alle anderen Geräte unsichtbar	nein	IMD weiß nichts von ext. Gerät	Anomalieerkennung +Jamming bzw. Alarm

$^{\alpha}$ Protokollvariante 1 = IMD fragt Cloaker nach Authentifikation für Programmer (Erstkontakt des Programmers ist das IMD)
Protokollvariante 2 = Programmer muss sich zuerst bei Cloaker authentifizieren (Erstkontakt des Programmers ist der Cloaker)

4 Zero-Power-Defense gegen Battery-Draining-Angriffe

Stajano et al [SA00] gehören zu den ersten Autoren, die Angriffe auf die Batterielebensdauer von Sensoren durch wiederholte Kontaktaufnahme betrachteten. Wir wollen diese Angriffsart in Anlehnung an [AY16] Battery-Draining-Angriffe (BDAs) nennen.
Halperin et al [Hal+08a] waren 2008 die ersten, die Gegenmaßnahmen gegen Battery-Draining-Angriffe vorschlugen und damit den Begriff Zero-Power-Defense prägten. Ihre Sorge war, das neu eingeführte Security-Maßnahmen für IMDs wie z.B. der Einsatz von Kryptographie bei der Authentifizierung von externen Kommunikationsgeräten einen neuen Angriffsvektor in Gestalt der BDAs eröffnet bzw. solche Angriffe durch erhöhten Energieverbrauch erleichtert. Ziel ihrer Zero-Power-Verteidigungsansätze ist Verhinderung von Angriffen, Authentifikation gegenüber dem IMD ohne Stromverbrauch für diesen und Benachrichtigung des Patienten durch einen Alarm falls sicherheitsrelevante Kommunikation mit dem IMD stattfindet.
Sie entwerfen drei Zero-Power-Verteidigungsmechanismen, die sie Zero-Power-Notification (ZPN), Zero-Power-Authentication (ZPA) und „spürbaren Schlüsselaustausch“ (SKE, engl. sensible key exchange) nennen und entwerfen mit diesen ein Authentifikationsprotokoll für IMD und Programmer.
Zero-Power-Notification: ZPN nutzt Funkenergie vom Programmer, um ein RFID-Gerät mit Piezoelement mit Energie zu versorgen, das den Patienten akustisch auf sicherheitsrelevante Ereignisse wie z.B. die Programmierung des IMDs aufmerksam macht. Diese Kommunikation kommt also ohne Verwendung der IMD-Batterie aus. Die Autoren verwenden hierfür ein RFID-Embedded-System namens WISP (engl. Wireless Identification and Sensing Platform) [Smi+06], an das sie ein Piezo-Element angeschlossen haben und das sie aufgrund dessen als WISPer bezeichnen. Wenn WISPer Anfragen über Funk erhält gibt es ein Zirpgeräusch von sich und informiert so den Patienten über die Funksendung an das IMD. Sie verwenden für das vom Piezo-Element erzeugte Geräusch einen 4-kHz-Ton, damit der Patient auch in lauteren Umgebungen einen Funkzugriff auf das IMD durch die entstehende Vibration bemerken kann und so gewarnt wird.
Zero-Power-Authentication (ZPA): Mit ZPA kann ein IMD überprüfen, ob es mit einem authentifizierten Programmer kommuniziert. Voraussetzung für dieses Kommunikationsprotokoll zwischen Programmer und dem WISPer-Teil des IMD sind allerdings vorverteilte Schlüssel. So soll in allen kommerziellen Programmern ein Hauptschlüssel K_m und in jedem IMD ein aus der Seriennummer I und dem Hauptschlüssel K_m mit Hilfe einer kryptographisch starken Pseudozufallsfunktion berechneter IMD-spezifischer Schlüssel $K_{\text{IMD}} = f(K_m, I)$ gespeichert sein. Die Autoren schlagen vor,

den Wert K_m in sicherer Hardware auf dem Programmer zu speichern.
Im Protokoll (siehe Abbildung 4.1) sendet der Programmer eine Authentifizierungsanfrage an WISPer, welcher mit der Seriennummer I und einer Nonce N antwortet. Der Programmer berechnet nun den IMD-spezifischen Schlüssel $K_{\text{IMD}} = f(K_m, I)$ und sendet dann als Antwort $R = RC5(K_{\text{IMD}}, N)$ an das WISPer-Modul. Dieses berechnet denselben Wert und vergleicht den vom Programmer erhaltenen Wert mit ihrem Ergebnis. Bei Gleichheit sendet er dem Programmer die Bestätigung der Authentifikation und dem IMD, dass ein Programmer erfolgreich authentifiziert wurde. Es wurde RC5 als Verschlüsselungsalgorithmus eingesetzt, da dieser damals auf einem RFID-Gerät mit durch ein Funksignal erlangter Energie ausführbar war.
Die Autoren sehen dieses Protokoll als Bootstrapping-Mechanismus für bessere Authentifizierungsmethoden, die dann auch die Batterie des IMD nutzen. Durch den Einsatz des WISPer-Moduls laufen Battery-Draining-Angriffe ins Leere.
Dieses Verfahren hat durch den Einsatz von allen beteiligten Parteien bekannten Schlüsseln das Problem der sicheren Schlüsselverteilung im Vorhinein. Falls der Master-Schlüssel K_m aus einem Programmer extrahiert werden kann wäre jegliche Kommunikation mit IMDs kompromittiert. Dieses System bietet auch keine Möglichkeiten für ein Schlüsselwiderruf bzw. Rekeying.

spürbarer Schlüsselaustausch (SKE): Bei SKE wird ein kryptographischer Schlüssel

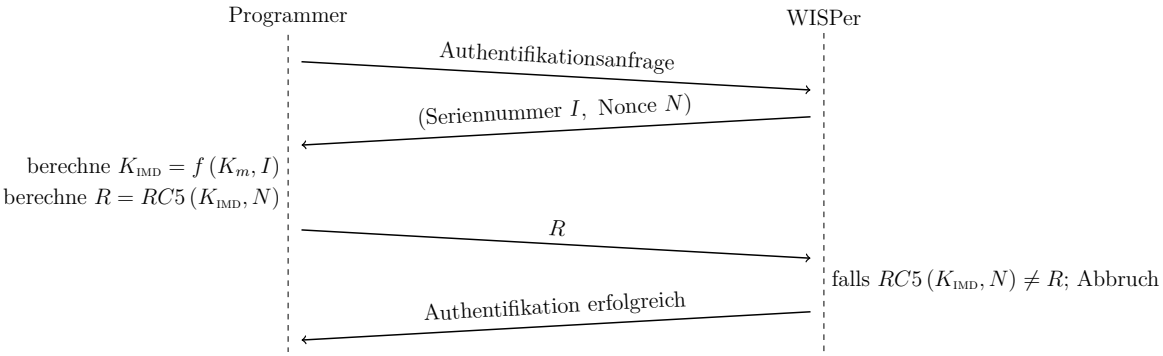

Abbildung 4.1: Authentifikationsprotokoll zwischen dem Programmer und einem am IMD angeschlossenen WISPer-Gerät

über Schall/Vibration vom IMD an den Programmer gesendet.
Zuerst sendet der Programmer ein Funksignal zur Stromversorgung der WISPer-Komponente an das IMD. Diese generiert dann einen Zufallswert, der als Sitzungsschlüssel verwendet wird, und sendet diesen als Schallwelle aus, welche von einem im Programmer integrierten Mikrofon in Kontakt mit dem Körper des Patienten in der Nähe des Implantats leicht erfasst werden kann. Die Autoren konnten mit ihren Messgeräten kein brauchbares Signal mehr über die Luft messen, sobald das Mikrofon von der Körperoberfläche entfernt wurde. Durch die Schwingung des Piezo-Elements nimmt der Patient jeden Authentifizierungsversuch wahr. Bei Experimenten von Halperin et al. dauerte die Übertragung eines 128-Bit Schlüssel $0,4$ Sekunden.
Nach dem Empfang des Schlüssels durch den Programmer kann eine sichere verschlüsselte Funkkommunikation zwischen IMD und Programmer erfolgen.
Die Autoren weisen selbst darauf hin, dass ein Abhören des Schlüssels über das

Audiosignal mit Spezialhardware oder durch Messen der elektrischen Abstrahlung des Piezo-Elements möglich sein könnte. Dies wird von Halevi und Saxena in [HS10] bestätigt, die mit einem Allzweckmikrofon den vom Piezoelement akustisch gesendeten Schlüssel aus $0,9$ Metern Entfernung mit einer Erfolgswahrscheinlichkeit von 99,88 % auslesen konnten. Bei Einsatz eines Parabolmikrofons konnten einzelne Schlüsselbits sogar aus einer Entfernung von $3,6$ Metern noch mit einer Erfolgswahrscheinlichkeit von jeweils 80 % ausgelesen werden.
Halperin et al gehen (unrealistischerweise) davon aus, dass jeder Programmer weltweit genau dieses Protokoll zur Authentifizierung verwendet. Falls aber medizinisches Personal einen solchen Programmer nicht zur Verfügung hätte wäre eine Kommunikation mit einem solchen IMD unmöglich.
In [LAK10] stellten Liu et al. ein Authentifizierungsprotokoll für BAN vor, das dem von Halperin et al. sehr ähnlich ist. So ist auch hier (angepasst auf das IMD-Szenario) ein RFID-Gerät zur Authentifizierung an das IMD angegliedert und IMD und Programmer haben ein gemeinsamen Schlüssel. Liu. et al entwerfen ein RFID-Modul Design, das AES mit CBC zur Erzeugung von n aufeinanderfolgenden WACs (s.u.) verwendet.
Zuerst sendet der Programmer einen sog. Wake-Up-Code (WAC) an das RFID-Modul, welches diesen prüft und bei Erfolg das IMD aktiviert, das eine Bestätigungsmeldung an den Programmer sendet. Falls diese nicht beim Programmer ankommt sendet dieser noch maximal weitere $n-1$ WACs mit Hilfe des AES-CBC-Designs. Sobald die Bestätigungsmeldung des IMD erfolgreich ankommt, können IMD und Programmer verschlüsselt kommunizieren.
Auch hier besteht das Problem, dass IMD und Programmer im Vorhinein einen gemeinsamen geheimen Schlüssel besitzen müssen.
Strydis et al. [Str+13] stellen eine neue Systemarchitektur für IMDs vor, bei der ein RFID-Sicherheits-Koprozessor im IMD integriert wird. Dieser Sicherheitsprozessor ist unabhängig vom primären IMD-System.
Die Autoren schlagen ein Authentifikationsprotokoll für IMD und Programmer vor, welches wie in allen vorher beschriebenen Protokollen einen gemeinsamen geheimen Schlüssel zwischen IMD und Programmer voraussetzen. Da sie einen RFID-Prozessor verwenden und somit nicht viel Energie für das Authentifikationsprotokoll zur Verfügung haben entscheiden sich Strydis et al für die Verwendung der energiesparenden Chiffre MISTY1 [MO00] und eines MAC-Algorithmus für die Nachrichtenauthentifikation. Um Sicherheitslücken vorzubeugen stützen Strydis et al das Protokolldesign auf ISO/IEC 9798 Teil 2. Die im Protokolldiagramm (siehe Abbildung 4.3) verwendeten Abkürzungen sind in Abbildung 4.2 erläutert. Im Folgenden wird der Ablauf des Protokolls (siehe Abbildung 4.3) zur gegenseitigen Authentifizierung von IMD und Programmer beschrieben:
Zuerst sendet der Programmer eine Authentifikationsanfrage an das IMD, welches daraufhin eine Nonce N_{IMD} erzeugt und an den Programmer zurücksendet. Dieser erzeugt nun ebenfalls eine Nonce N_{PROG} und berechnet einen MAC der Noncen, der ID des IMD und des zu sendenden Befehls. Er schickt die Noncen, den MAC und den mit dem gemeinsamen Schlüssel verschlüsselten Befehl an das IMD. Das IMD entschlüsselt den Befehl und berechnet ebenfalls den MAC und vergleicht diesen mit

IMD	das IMD
PROG	der Programmer
ID_{IMD}	die ID/Seriennummer des IMDs
ID_{PROG}	die ID/Seriennummer des Programmers
κ	gemeinsamer Schlüssel von IMD und Programmer
$\mathsf{Enc}_K(\cdot)$	ist die eingesetzte Verschlüsselungsfunktion, die MISTY1 mit Schlüssel K verwendet
$\mathsf{Dec}_K(\cdot)$	ist die eingesetzte Entschlüsselungsfunktion, die MISTY1 mit Schlüssel K verwendet
N_K	eine von Gerät K berechnete Nonce
CMD	ein gesendeter Befehl
ANS	eine gesendete Antwort
MAC_κ	berechnet ein MAC mit dem Schlüssel κ

Abbildung 4.2: Abkürzungen des Authentifikationsprotokoll zwischen dem Programmer und dem Sicherheitsprozessor aus Strydis et al.

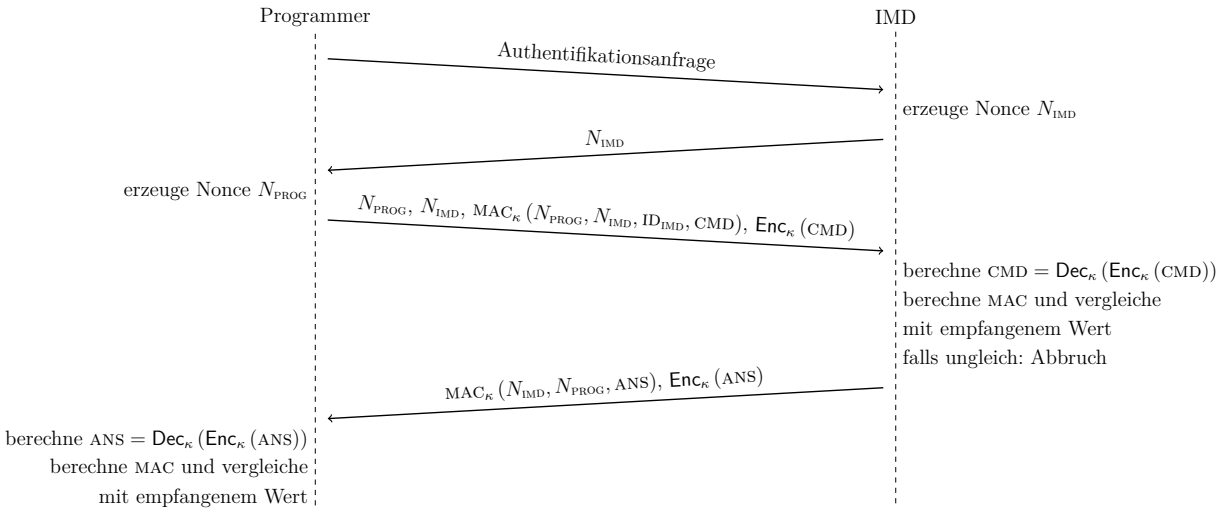

Abbildung 4.3: Authentifikationsprotokoll zwischen dem Programmer und dem Sicherheitsprozessor aus Strydis et al.

dem empfangenen MAC. Falls diese unterschiedlich sind bricht das IMD das Protokoll ab, andernfalls aktiviert das IMD den Hauptprozessor mit Batterieversorgung und berechnet die MAC der Noncen und einer Antwortnachricht an den Programmer und schickt diesen mit der mit κ verschlüsselten Antwort an den Programmer. Dieser entschlüsselt die Antwort, berechnet selbst die MAC und vergleicht diese mit dem empfangenen Wert. Falls die Werte gleich sind, sind IMD und Programmer nun beide authentifiziert.

Yang et al. [Yan+14a] realisieren den Entwurf von Halperin et al. in Hardware, d.h. sie verwenden auch an Programmer und IMD vorverteilte Schlüssel und es authentifiziert sich nur der Programmer gegenüber dem IMD. Sie definieren zusätzlich noch das Übertragungsformat der Nachrichten und führen eine Shutdown-Nachricht am Ende der Protokollsitzung zwischen IMD und Programmer ein, um Replay-Attacken zu verhindern.

Ellouze et al. [Ell+13] [Ell+18] verwenden das gleiche WISP-Modul wie [Hal+08a]. Im Gegensatz zu [Hal+08a] bietet ihr vorgeschlagenes Protokoll eine beidseitige Authentifizierung von IMD und Programmer. Zur Schlüsselgenerierung verwenden sie das physische Signale verwendende Protokoll OPFKA aus [Hu+13] (siehe Abschnitt 2.7),

welches jedoch anfällig für Angriffe ist [Ros+13a].
2021 stellten Siddiqui et al. [Sid+21] ein Pairing-Protokoll vor, das Ultraschall im MHz-Bereich für den Schlüsselaustausch verwendet. Sie konnten zeigen, dass im Gegensatz zum „spürbaren Schlüsselaustausch“ von [Hal+08a] welcher Ultraschall mit einer Frequenz von 4 KHz verwendet, Ultraschall im MHz-Bereich eine maximale Abhörreichweite außerhalb des menschlichen Körpers von wenigen Millimetern hat. Deshalb muss der Ultraschallkopf des Programmers für die Datenübertragung auf der Haut des Patienten anliegen. Der Programmer sendet nun über diesen Kanal den Master-Schlüssel unverschlüsselt an den IMD. Am IMD ist ein Ultraschallwandler installiert, der diese Ultraschallwellen aufnimmt, sie in Strom umwandelt und gleichzeitig den Schlüssel speichert und den IMD aus dem Schlafmodus aufweckt. Dieser ist dann für verschlüsselte Funkkommunikation mit dem Programmer bereit.
Eine an ZPD-Techniken angelehntes Gebiet ist der Einsatz von wiederaufladbaren Akkus in IMD. Dies ist ein großes Forschungsgebiet und Khan et al. [Kha+20] gibt einen umfassenden Überblick über die dabei eingesetzten Techniken. Einige Energieübertragungstechniken werden schon für Implantate wie Cochlea- und Augenimplantate eingesetzt, die weiter an der Oberfläche des Körpers implantiert sind. Die Forschung bzgl. Aufladbarkeit von Implantaten im Herzen und im Gehirn läuft, allerdings besteht hier noch die Gefahr von Gewebeschäden durch die übertragene Energie. Auch für eine Energieübertragungstechnik, die Ultraschallwellen zur Energieübertragung verwendet, bedarf es noch Untersuchungen der Langzeiteffekte der Ultraschallwellen auf das Gewebe.
Die bei den ZPD-Protokollen eingesetzten Zero-Power-Technologien wie RFID oder die Ultraschallwandler-Konstruktion von Siddiqi et al. bieten eine funktionierende Abwehr gegen Battery-Draining-Angriffe, Angriffe gegen diese Abwehrtechniken sind bisher nicht bekannt. Diese Technologien stellen also einen geeigneten Baustein für IMDs dar, um Battery-Draining-Angriffe abzuwehren. Im Folgenden fassen wir die in diesem Kapitel eingesetzten ZPD- und Schlüsselaustauschmethoden in Tabelle 4.1 zusammen.

Paper	ZPD-Technik	Schlüsselaustausch-Technik
[Hal+08a]	RFID-Chip am IMD	Vibration mit 4 kHz
[LAK10]	RFID-Chip am IMD	sog. Wake-Up-Codes (WACs) für Sitzungsschlüssel, verteilter Master-Key wird vorausgesetzt
[Str+13]	RFID-Koprozessor im IMD	Schlüsselaustauschprotokoll nach ISO/IEC 9798 Teil 2 für Sitzungsschlüssel, gemeinsamer Schlüssel von IMD und Programmer wird vorausgesetzt
[Yan+14a]	RFID-Chip am IMD	gemeinsamer Schlüssel von IMD und Programmer wird beim Produktionsprozess in diesen gespeichert
[Ell+13] [Ell+18]	RFID-Chip am IMD	PS-Schlüsselerzeugungsprotokoll aus [Hu+13] (siehe Abschnitt 2.7)
[Sid+21]	Ultraschallwandler	Übertragung des Schlüssels im Klartext mit sicherem MHz-Ultraschallsignal über den Körper als Medium

Tabelle 4.1: Übersicht über die ZPD-Techniken

5 Anomalieerkennung

Eine weitere Abwehrmaßname gegen Angriffe wie schädigende Eingriffe in die Therapie oder Battery-Draining-Angriffe auf ein IMD ist die Anomalieerkennung. Dabei sucht man nach ungewöhnlichen Zugriffsmustern auf das IMD wie z.B. Ort, Zeit, Wochentag oder Häufigkeit des Zugriffs, auch physikalische Eigenschaften der Sendung wie Signalstärke, Einfallwinkel oder ungewöhnlichen Inhalten der an das IMD gesendeten Befehle mit plötzlichen starken Änderungen an den im IMD eingestellten Therapiewerten.
Hier kann man nun Grenzwerte festlegen und einstellen, so daß z.B. bei implantierten Insulinpumpen mehrere Insulinboli direkt hintereinander einen möglichen Angriff nahelegen. Diese Werte können im Vorhinein z.B. vom Mediziner eingestellt werden. Einige Ansätze setzen auch Techniken aus dem maschinellen Lernen wie SVMs (Support-Vektor-Maschinen) oder neuronale Netze zum Training der als „normal“ geltenden Werte ein. Nachdem die Anomalieerkennung beim Patient eine gewisse Zeit in seinem Alltag im Trainingsmodus eingesetzt wird, kann man nach dieser Phase die Erkennung scharf schalten. Je nach Konzept wird dann bei Erkennung von ungewöhnlichen Zugriffsmustern auf das IMD Alarm gegeben oder der Zugriff auf das IMD verweigert bzw. muss dann eine Authentifikation zur Durchführung des Zugriffs auf das IMD stattfinden.
Hei et al. [Hei+10] waren 2010 die ersten, die durch Anomalieerkennung insbesondere Angriffe auf den Ressourcenverbrauch des IMD (engl. ressource depletion attacks) abwehren wollten. Sie beobachten die fünf Zugriffsmerkmale Befehlsart, Zeitintervall zwischen zwei gleichen Befehlen, Aufenthaltsort des IMDs, aktuelle Uhrzeit und Wochentag und verwenden SVMs zur Festlegung der Grenzen für normale und ungewöhnliche Zugriffe. Sie schlagen wegen der enormen Rechenleistung und Akkuladung sowie des Vorhandenseins eines GPS-Moduls vor, die SVMs als Smartphone-App zu realisieren. Mit Hilfe der Klassifizierung der IMD-Zugriffe in Echtzeit soll ein ressourcenverbrauchender Zugriff auf das IMD durch einen Angreifer verhindert werden.
Bei Kontaktaufnahme durch ein Lesegerät sendet das IMD zunächst eine kurze Verifizierungsanfrage an das Smartphone des Patienten, welches dann den Klassifizierungsalgorithmus ausführt. Falls dieser den Zugriff auf das IMD als normal klassifiziert sendet die Smartphone-App ihr OK für die Fortsetzung der Kommunikation mit dem Programmer an das IMD. Bei Einstufung des Zugriffs auf das IMD als Angriff sendet die App einen Abbruch-Befehl an das IMD. Bei einem uneindeutigen Klassifizierungswert gibt die App Alarm und nun muss der Patient in einem Auswahldialog der App dem Zugriff auf das IMD zustimmen oder diesen ablehnen.
Hei et al. geben an, das ihr SVM-basiertes Verfahren für den IMD-Zugang nicht für den Notfall entworfen wurde und empfehlen für den Notfall entweder eine automatische Erkennung von Notfällen im IMD (z.B. durch abnormale physiologische Messwerte) oder den Einsatz von Backdoors wie z.B. ein gemeinsamer Hauptschlüssel für den

Zugang zu einer Gruppe von IMDs, welcher dann in den Krankenhäusern und allen Krankenwagen zur Verfügung steht und so Zugriff auf das IMD gewährleistet. Dieses System ist unrealistisch, sobald der Patient im Urlaub ist, wäre kein Zugriff auf seinen IMD mehr möglich.
Auch das einfache Konzept der Nachfrage des IMD an die Klassifizierungs-App mit Antwort ist z.B. durch Jamming, Replay- oder Reordering-Angriffe leicht zu knacken, Hei et al. geben auch keine genaueren Protokollbeschreibungen an.
Ein System zur nachträglichen Anomalieerkennung durch Audits für ein implantiertes Insulinpumpensystem stellen Henry et al. [Hen+13] vor. Sie erfassen permanent die Darmgeräusche des Patienten durch die das System feststellt, ob der Patient isst. Gleichzeitig werden die vom System abgegebenen Insulingaben erfasst. Diese Daten werden in der Insulinpumpe für nachträgliche forensische Untersuchungen gespeichert. Wenn die Menge der Insulinabgabe nicht mit der Nahrungsaufnahme des Patienten zusammenpasst kann ein Forensiker diese anomale Insulinzufuhr als möglichen Indikator für eine Sicherheitsverletzung oder einen Fehler in der Insulinpumpen-Steuerung erkennen. So interpretieren die Autoren ungewöhnliche Bolusgaben, die nicht mit einer Mahlzeit übereinstimmen entweder als einen Hinweis auf eine vom Patienten nach einer Mahlzeit vergessene Bolusgabe, eine Funkstörung oder einen Angriff.
Dieses System ist so wie es präsentiert wird nur im Nachhinein für forensische Untersuchungen nützlich. Um akute Angriffe zu verhindern, müsste es gemeinsam mit einem aktiven Schutzsystem eingesetzt werden.
Zhang et al. [ZRJ13] schlagen MedMon vor, ein externes Gerät, das ungewöhnliche Zugriffe auf das IMD sowohl anhand von Ort/Zeit/Wochentag als auch durch physikalischen Eigenschaften des zugreifenden Funksignals erkennt. Eine genauere Beschreibung findet sich in Abschnitt 3.3.
Hei et al. stellten 2015 in [Hei+15a] ein Erkennungssystem namens PIPAC für erhöhte Basal- und Bolusdosen für implantierte Insulinpumpen vor. Das System verwendet SVMs zum Lernen der normalen Infusionsmengen und -rate. Die erzeugten Regressionsmodelle legen dynamische Grenzen für die Infusionsdosis fest und ermöglichen somit das Erkennen von anomalen Dosismengen. Ziel des Systems ist die Abwehr von Angriffen auf den Patienten durch akute Überdosierung oder über einen langen Zeitraum stattfindende geringe Überdosierung (welche ebenfalls erhebliche Schädigungen des Gesundheitszustandes des Patienten herbeiführen können). Bei Angriffsdetektion gibt es weiterhin normale Insulindosen (im Bereich der dynamischen Grenzen des Systems) statt der erhöhten Dosen ab.
Gao et al. [GT17] verwenden SVMs bzw. den ID3-Algorithmus, um Entscheidungsbäume für sieben ausgewählte Eigenschaften von an das IMD gesendete Signale zu erzeugen und so ungewöhnliche Zugriffsmuster zu erkennen. Bei Erkennung eines Angriffs soll ein vom Patienten mitgeführtes elektronisches Gerät wie z.B. sein Smartphone eine Warnung gesendet bekommen. Dabei lassen Gao et al. ihren Anomalieerkennungsalgorithmus auf einem externen Gerät ausführen, das sich zwischen Programmer und IMD befindet und das bei Erkennung eines Angriffs die Kommunikation des Angreifers zum IMD blockiert. Wie die Kommunikation zwischen diesem Gateway und IMD bzw. Programmer ablaufen soll beschreiben sie nicht.
In [Rat+17] und [Rat+18] nutzen Rathore et al. ein MLP (engl. multi layer percep-

tron), um zu prüfen, ob der vom Glukosesensor an die Insulinpumpe übermittelte Glukosewert im Normalbereich liegt oder einen Angriff darstellen könnte. In [Rat+17] werden 5 Eingabeneuronen mit 3 Hidden Layers verwendet, in [Rat+18] werden 10 Eingabewerte verwendet. Die Autoren zeigen in Experimenten, dass ihre Klassifikation dem der vorhergehenden SVM-Klassifizierer überlegen ist.
Die Techniken zur Anomalieerkennung können konstruktionsbedingt nur einen Teil der Angriffe abwehren, so dass beim Einsatz dieser andere Abwehrmaßnahmen wie Authentifikation zur Erkennung unbefugter Zugriffe auf das IMD eingesetzt werden sollten. Im Folgenden fassen wir die in diesem Kapitel eingesetzten Anomalieerkennungstechniken und den Ort der Anomalieerkennung in Tabelle 5.1 zusammen.

Paper	Anomalie-Erkennungstechnik	Ort der Anomalieerkennung
[Hei+10]	SVM (engl. support vector machine)	Smartphone
[Hen+13]	keine, nur Speicherung von Zugriffsdaten und Darmgeräuschen zur nachträglichen forensischen Untersuchung	implantierte Insulinpumpe
[ZRJ13]	Grenzwerte physikalischer Signaleigenschaften werden trainiert, die Trainingsmethode ist nicht angegeben, andere Grenzwerte werden manuell eingestellt	Smartphone
[Hei+15b]	SVM	nicht angegeben
[GT17]	SVM + ID3-Algorithmus [Qui86]	externes Gerät
[Rat+17] [Rat+18]	MLP (engl. multi layer perceptron)	in IMD integrierter Spezial-Chip

Tabelle 5.1: Übersicht über die Anomalieerkennungstechniken

6 Entfernungs-prüfende-Protokolle (engl. distance bounding protocols)

(Dieses Kapitel orientiert sich an [Rus+14].)
Distance-Bounding-Protokolle (DBPs) messen die Entfernung zwischen IMD und Programmer, um die Nähe des Kommunikationspartners sicherzustellen. Das Security-Konzept bei DBPs ist die Annahme, dass ein Angreifer Signale nicht schneller Senden kann als das verwendete Signal und Medium physikalisch zulassen.
In [Ras+09] verwenden Rasmussen et al. Ultraschall, um die Umlaufzeit der Signale zwischen IMD und Programmer zu messen. Zur Schlüsselgenerierung kommt eine gegen Distance-Shortening Angriffe sichere Variante des Diffie-Hellman-Protokolls zum Einsatz. Da die Geschwindigkeit von Ultraschall [Par+94] im menschlichen Körper schneller ist als in der Luft (1500 $\mathrm{km\,h^{-1}}$ im Vergleich zu 340 $\mathrm{km\,h^{-1}}$) und die Autoren eine Maximaldistanz zwischen Programmer und IMD von 5 cm vorschlagen kann dieses Verfahren zu den touch-to-access Verfahren gezählt werden.
Shi et al. [Shi+13] schlagen ein Verfahren vor, das die Fluktuation der Signalstärke des empfangenen Funksignals (RSS, engl. received signal strength) misst, um festzustellen, ob sich ein Kommunikationspartner am oder im Körper befindet oder sich in größerer Entfernung aufhält. Dieses Verfahren funktioniert, weil die RSS-Fluktuation zwischen zwei Geräten am/im selben Körper stabiler ist als zwischen einem Körpergerät und einem externen Gerät. Dieses Protokoll benötigt mindestens zwei Geräte am/im Körper, eignet sich also nur für Implantate, die mit dazugehörigen Sensoren/Aktoren am/im Körper ein BAN bilden.
Jurik et al. [Jur+11] messen kontinuierlich EKG-Signale und senden diese verschlüsselt an einen Kommunikationspartner. Durch das periodische Senden der Signale soll die Nähe der Kommunikationspartner sichergestellt werden. Es wird allerdings nur geprüft, ob ein Herzschlag übertragen wird und ob regelmäßig Schläge ankommen. Eine Replay-Attacke, auch aus größerer Entfernung, erscheint hier realistisch.
Camara et al. [Cam+21] stellen ein Protokoll vor, das LLLT (engl. low-level light therapy), einen externen out-of-band Kanal über schwaches Laserlicht nutzen, um einen Sitzungsschlüssel zu übertragen. Die Nähe zwischen Programmer und IMD wird dann durch Messen und Vergleichen des QRS-Komplexes des Herzens evaluiert.
Eine Entfernungsmessung allein weist nur physische Nähe nach, was höchstens als „schwache Authentifizierung“ gewertet werden kann. Ein Nachweis der Identität der Geräte durch Authentifikation muss zusätzlich zum Nachweis der Nähe noch erbracht werden.
Eine Übersicht über Angriffe auf Disctance-Bounding-Protokolle findet sich bei Clulow et al [Clu+06] und Cremers et al [Cre+12], eine Gesamtübersicht über DBPs mit

Angriffsanalysen geben Avoine et al. [Avo+18].
In Tabelle 6.1 findet sich eine tabellarische Übersicht der DBPs.

Paper	verwendetes Medium	Distanz-Prüfmechanismus
[Ras+09]	Ultraschall + Funk	Bit-Umlaufzeit-Messung
[Jur+11]	Funk	Vorhandensein des Herzschlags
[Shi+13]	Funk	RSS-Fluktuaktionen (engl. received signal strength)
[Cam+21]	schwacher Laser + EKG	Abgleich vom QRS-Komplex des EKGs

Tabelle 6.1: Übersicht über die Distance-Bounding-Protokolle

7 Nutzung anderer Kommunikationskanäle: OOB-Verfahren (engl. out-of-band)

Einige Verfahren zum sicheren Schlüsselaustausch von IMD und Programmer verwenden einen externen Kommunikationskanal, nutzen also sog. OOB-Kommunikation (engl. out-of-band). Diese stützen sich meistens auf das touch-to-access Prinzip und gehen davon aus, dass das übermittelte Signal außerhalb des menschlichen Körpers nicht messbar ist. So sehen die Autoren dieser Arbeiten diese künstlichen physikalischen Datenkanäle als inhärent sicher an. Im Folgenden stellen wir insbesondere Papers vor, die solch eine OOB-Kommunikation als Hauptmerkmal ihres Verfahrens herausstellen, andere Arbeiten, die solch einen Kanal „nebenbei“ verwenden, erwähnen wir natürlich.
So verwenden Halperin et al. [Hal+08a] in ihrem ZPD-Verfahren Schallwellen für die Übermittlung einer Nonce (siehe Kapitel 4).
Rasmussen et al. [Ras+09] nutzen Ultraschall als Medium für ein Schlüsselaustauschprotokoll (siehe auch Kapitel 6).
Schechter [Sch10] schlug 2010 die Verwendung eines ultraviolettes Tatoos am Körper zur Schlüsselhinterlegung vor, wobei hier ein Austausch des Schlüssels nicht möglich ist. Hier muss auch der Schutz des Schlüssels beim Sonnenbaden und bei Verletzungen bedacht werden. Schechter schlägt hier den Einsatz von Fehlerkorrekturverfahren und eine Zweitanbringung des Schlüssels an der Fußsohle vor.
Kim et al. [Kim+15] schlagen einen Schlüsselaustausch per Vibrationen vor. Dieser hat eine Übertragungsgeschwindigkeit von 20 Bit/sec. Sie untersuchen diesen Kommunikationskanal und weisen nach, dass ab einem Abstand von 20 cm von der Haut des Patienten das im Körper übertragene Vibrationssignal im Umgebungsrauschen verschwindet und nicht mehr messbar ist. Sie schlagen als Nachweis der korrekten Übermittlung des Schlüssels vom Programmer zum IMD vor, dass das IMD eine feste, vorher eingespeicherte Nachricht c an den Programmer sendet. Ein „fremder“ Programmer ohne Zugriff auf c hätte also keinen Zugriff auf das IMD.
Zhao et al. [Zha+21b] verbessern die Übertragungsrate auf dem Vibrationskanal auf 40 Bit/sec durch den Einsatz eines CNNs (engl. convolutional neural network) mit 10 Layern, schlagen allerdings kein Protokoll vor.
Siddiqi et al. stellen in [Sid+21] ein Schlüsselaustauschprotokoll vor, das Ultraschall verwendet und ZPD-Techniken einsetzt (siehe Kapitel 4).
Camara et al. [Cam+21] stellen ein Schlüsselaustauschprotokoll vor, das niedrigenergetische Laserstrahlen als Kommunikationsmedium verwendet.
In Tabelle 7.1 geben wir eine Übersicht über die verwendeten OOB-Verfahren.

Paper	Kommunikationskanal	Geschwindigkeit der Datenübertragung
[Hal+08a]	Schall	320 Bit/sec
[Ras+09]	Ultraschall	1000 Bit/sec
[Sch10]	ultraviolettes Licht	nicht angegeben
[Kim+15]	Vibration	20 Bit/sec
[Zha+21b]	Vibration	40 Bit/sec
[Sid+21]	Ultraschall	nicht angegeben
[Cam+21]	LLLT (low-level laser therapy)	nicht angegeben

Tabelle 7.1: Übersicht über die OOB-Verfahren

8 durchgeführte Hacks gegen IMDs

Viele IMDs heutzutage haben oft keine ausreichenden Security-Mechanismen und sind so anfällig für Angriffe über die Funkschnittstelle. So waren Halperin et. al [Hal+08a] die ersten, die mit Hilfe eines Software-Defined-Radios (SDRs) einen IMD ohne Verwendung eines Programmers programmieren konnten. Sie analysierten das proprietäre Protokoll zwischen dem Programmer und dem zugehörigen ICD, welche komplett unverschlüsselt kommunizierten, und konnten durch Aufzeichnung und Replay diesen fernsteuern. Dies gelang den Autoren bis zu einer Entfernung von 10 cm.
Li et al. [LRJ11] nutzten auch ein SDR, um das Protokoll einer implantierten Insulinpumpe zu rekonstruieren. Auch hier verlief die Kommunikation komplett unverschlüsselt. Als „Sicherheitsmechanismus “ verwendete die Insulinpumpe einen Counter und verwarf ankommende Pakete, wenn sie den gleichen Counterwert hatten. Li et al zeichneten einfach zwei aufeinanderfolgende Nachrichten auf und sendeten sie alternierend. Damit konnten sie die Insulinabgabe beliebig erhöhen. In ihren Experimenten gelang das bis zu einer Entfernung von 20 Metern.
Der Sicherheitsforscher Barnaby Jack demonstrierte 2011 einen Angriff auf implantierbare Insulinpumpen [22], bei dem eine implantierte Insulinpumpe ihren gesamten Vorrat an Insulin ausschüttete, was einer tödlichen Dosis entspricht. 2012 demonstrierte er auf der RSA Sicherheitskonferenz, dass ein Zugriff auf implantierte Insulinpumpen im Umkreis von über 90 Metern möglich ist [Par12].
2016 konnten Marin et al. [Mar+16c] das Kommunikationsprotokoll von mehreren ICDs rekonstruieren. Bei diesem kam zum ersten Mal eine Schutzmaßnahme zum Einsatz, die versendeten Nachrichten des Protokoll wurden mit einer LFSR-Sequenz (engl. Linear Feedback Shift Register) geXORt. Diese LFSR-Sequenz war aber bei allen Kommunikationssitzungen und bei den über 10 von den Autoren untersuchten ICD-Modellen konstant. Sie stellten fest, dass die ICDs nicht gegen Replay-Angriffe geschützt waren, so dass Angreifer Replay-Angriffe durchführen können ohne die Protokollspezifikationen zu kennen. Diese Angriffe können bis zu einer Entfernung von 5 Metern durchgeführt werden.
Im gleichen Jahr rekonstruierten Marin et al. [Mar+16d] das Kommunikationsprotokoll einer Insulinpumpe. Die Sendungen fanden weiterhin im Klartext statt. Außerdem hatte sich der Sicherheitsmechanismus des Countervergleichs seit der Analyse von Li et al. nicht geändert. Somit war ein Replay-Angriff durch das Senden zweier aufeinanderfolgender aufgezeichneter Nachrichten weiterhin möglich. Die Autoren konnten auch beliebig die Insulinpumpe aus- und einschalten und beliebig hohe Insulindosen verabreichen lassen. Die Autoren geben als Entfernung für einen erfolgreiche Angriff bis zu 5 Meter an, weisen aber darauf hin, dass sich diese noch erheblich vergrößern lasse.
2018 schließlich gelang Marin et al. [Mar+18] das Reverse-Engineeren des Kommuni-

kationsprotokolls eines Neurostimulators. Auch hier fand die Kommunikation ohne jegliche Verschlüsselung statt. Die Ausführung von Replay-Angriffen war uneingeschränkt möglich, alle Einstellungen des Neurostimulators konnten verändert werden. Durch die Kenntnis des rekonstruierten Protokolls konnten sie auch beliebige selbst erstellte Nachrichten senden. Der von den Autoren untersuchte Neurostimulator prüft bei eingehenden Nachrichten durch Überprüfen der Seriennummer, ob er adressiert ist. Marin et al. fanden heraus, dass dies durch Senden einer 0 als Seriennummer umgangen werden kann und somit jedes Gerät ohne Kenntnis seiner Seriennummer umprogrammiert werden kann. Sie konnten auch Daten über Diagnose, Symptome, Krankheit und Therapie des Patienten aus den Geräten auslesen. Die maximale Entfernung aus der die Autoren erfolgreich Angriffe durchführen konnten betrug 10 cm. Wir geben in Tabelle 8.1 einen Überblick über die erfolgreichen Hacks auf IMDs.

Paper/Artikel	IMD-Typ	Angriff möglich aus Entfernung	Verschlüsselung der Kommunikation
[Hal+08a]	ICD	10 cm	keine
[LRJ11]	Insulinpumpe	20 Meter	keine
[Par12]	Insulinpumpe	90 Meter	keine
[Mar+16c]	ICD	5 Meter	XOR mit konstantem LFSR
[Mar+16d]	Insulinpumpe	> 5 Meter	keine
[Mar+18]	Neurostimulator	10 cm	keine

Tabelle 8.1: Übersicht über erfolgreiche Hacks auf IMDs

9 Zusammenfassung und Ausblick

In den vorangegangenen Kapiteln haben wir einen Überblick über die vorhandene Forschung zur Sicherung des Funkkanals zwischen dem Programmer und dem IMD gegeben. Die in den wissenschaftlichen Artikeln angegebenen Lösungen versuchen, die Security des IMD-Systems ohne große (negative) Auswirkungen auf die Utility zu verbessern. Sie verwenden u.a. physiologische Signale zur sicheren Schlüsselerzeugung, externe Geräte als Proxy zwischen IMD und Programmer, Anomalieerkennung zur statistischen Abwehr von Angriffen, Methoden zur Sicherstellung einer geringen Entfernung zum Kommunikationspartner, künstliche Kommunikationskanäle wie Sound und Vibrationen oder RFID-Technologie zur Authentifikation oder zur Abwehr von Angriffen auf die Batterie des IMD.
In Anlehnung an das SoK-Paper von Rushanan et al [Rus+14] wollen wir in Tabelle 9.1 einen Überblick über die vorgestellten wissenschaftlichen Artikel nach ihrem Zweck bzgl. Abwehr von Angriffen (siehe Abschnitt 1.6) geben. Bei den Angriffsarten in der Tabelle haben wir Seitenkanalangriffe außen vor gelassen, da alle betrachteten Paper bis auf [Ras+09] keine Abwehr gegen solche Angriffe in Erwägung ziehen. Außerdem haben wir die verwendeten kryptographischen Primitive aufgelistet, wobei auffällt, dass die wissenschaftlichen Artikel zur Security von IMDs oft ohne diese auskommen. Dies ist dem notwendigerweise niedrigen Ressourcenverbrauch der IMDs geschuldet, denen man keine rechenleistungsintensive Kryptographie zumuten möchte:
So verwenden viele Artikel als sicher angenommene Seitenkanäle (engl. out-of-band channel, wir verwenden in der Tabelle die Abkürzung OOB) wie Ultraschall im Körper des Patienten oder gehen von schon verteilten Schlüsseln zwischen IMD und Programmer aus. Andere wiederum messen physiologische Signale wie z.B. den Herzschlag, um so aus einem gemeinsam gemessenen ähnlichen Wert einen Schlüssel für die Kommunikation mittels symmetrischer Algorithmen zu berechnen. Einige Arbeiten bewerten auch die frisch gemessenen einmal verwendeten physiologischen Signale als sichere Abwehr gegen Replay- und Reordering-Angriffe. Viele Verfahren verwenden auch physische Sicherheitsmerkmale, wie die Messung des Abstandes zum Kommunikationspartner (z.B. durch Messung der Umlaufzeit des Funksignals) oder die Ermittlung des Einfallswinkels und der Stärke des einkommenden Funksignals. Wie zu erwarten war, sind alle vorgeschlagenen Verfahren bis auf die auf die Zero-Power-Defense spezialisierten Arbeiten für Battery-Draining-Angriffe anfällig. Hier gibt es eine Ausnahme, eine Möglichkeit im Stategiepapier von Denning et al. [DFK08] sieht ein externes Gerät als Proxy zwischen IMD und Programmer vor, das eine Kommunikation mit dem IMD erst nach Authentifizierung erlaubt. Wir haben Ausdrücke und Werte in der Tabelle in Klammer gesetzt, falls diese Maßnahme nur teilweise umgesetzt wurde. Die in der Tabelle erwähnten Sicherungsmaßnahmen gegen Angriffe werden teilweise als solche in den Papern erwähnt. Wir haben keine komplette Sicherheitsanalyse der

Verfahren durchgeführt, um zu untersuchen, ob diese Verfahren wirklich gegen die erwähnten Angriffe sicher sind. Diese Einschätzungen entsprechen den Artikeln selbst (falls ein Artikel eine Maßnahme als sicher einschätzt haben wir derselben Maßnahme bei vergleichbaren Verfahren die gleiche Schutzwirkung zugemessen).
Diese Einschätzungen können aber auch sehr trügen. So sind Rostami et al. [Ros+13b] die Einzigen, die einen Sicherheitsbeweis für ihr Heart-2-Heart-Protokoll vorstellen. Leider gibt es sowohl einen sog. Spiegelangriff gegen das Verfahren, bei dem man durch Replay der vom IMD erhaltenen Nachrichten am Schluß gegenüber dem IMD authentifiziert ist, als auch eine MitM-Attacke (siehe Unterabschnitt 2.6.1). Weitere Ansätze für die kryptographische Sicherung der für die Kommunikation zwischen IMD und Programmer entworfenen Protokolle bieten [Str+13] und [Hei+15b], die den Standard ISO/IEC 9798-2 für den Entwurf ihrer Protokolle verwenden, welcher Authentifikationsverfahren bereitstellt.

9.1 Forschungsperspektiven zur Verbesserung der IMD-Sicherheit

Für die Zukunft der Forschung auf dem Gebiet der Security von IMDs sollten Entwurfsverfahren für sichere Authentifikations- und Schlüsselaustauschprotokolle bestehen, die den geforderten geringen Stromverbrauch als auch den Notfall berücksichtigen. Dieser ist einer der Hauptangriffspunkte für Zugriffsprotokolle von IMDs - im medizinischen Notfall muss medizinisches Personal sofort Zugriff auf das IMD erhalten können, dies geschieht dann meistens (auch in den vorgestellten Protokollen) unverschlüsselt und ohne Authentifikation. Hier hat sich als Lösung das touch-to-access-Prinzip etabliert, das wohl einen realistischen Kompromiss zwischen Security im Normalfall und Utility im Notfall darstellt. Die Prävention von Angriffen aus der Ferne, die solch einen Notfallmodus ausnutzen stellt einen wichtigen Forschungsansatz dar.
In Anlehnung an den Angriff von touch-to-access-Protokollen gibt es auch die Attacken auf die bis zu einem gewissen Grad als sicher geltenden physiologischen Signale zur Schlüsselerzeugung. Hier konnte auch schon gezeigt werden, dass (elektrische) Herzschläge aus der Ferne per Video gemessen werden können. Hierauf kann man aufbauen und forschen, bis zu welcher Entfernung diese und auch andere Körperwerte wie z.B. Blutdruck oder EEG-Werte ausgelesen werden können, denn im Gegensatz dazu, was der Titel der Arbeit von Calleja et al. [CPT15] „Electrical heart signals can be monitored from the moon“ andeutet, haben die Forscher den Herzschlag aus einer Entfernung von 60 cm ermittelt. Der einzige vom Herzen gemessene Standardwert, der noch nicht ohne Kontakt zum Körpers gemessen werden konnte, ist das vollständige EKG. Für eine erhöhte Sicherheit von generierten Schlüsseln können mehrere physiologische Signale gleichzeitig gemessen werden.
Die Verwendung von physiologischen Signalen zur Erzeugung kryptographischer Schlüssel benötigt den Einsatz unscharfer kryptografischer Primitive, wie die auf der Fuzzy-Kryptoprimitive aufbauenden Verfahren. Hier muss untersucht werden, welche Verfahren sich für IMDs bzgl. Zeit-, Speicher- und Strombedarf besonders

eignen.
Auf dem Gebiet der entfernungsprüfenden Protokolle kann man versuchen, die bekannten Angriffe auf solche Protokolle (siehe Ende Kapitel 6) anzuwenden oder zu verbessern und andererseits die Signale oder auch Protokolle so geschickt zu wählen, dass diese Angriffe unterbunden werden.
Bzgl. der Angriffsdetektion durch Anomalieerkennung kann man versuchen, die Erkennungsrate von Angriffen zu verbessern oder den Einsatz anderer Erkennungstechniken wie z.B. KI erproben. Auch die Erforschung weiterer möglicher Angriffsmerkmale könnte zu einer höheren Erkennungsrate führen.
Weitere mögliche Forschungsfelder für die Security von IMDs wären die Erforschung von Seitenkanalangriffen auf ein IMD, die Erforschung von Angriffen auf die analoge Schnittstelle des IMD oder auch Angriffe auf die Firmwareupdatemechanismen von IMDs.

			verwendete Krypto-Primitive					Protokoll/Methode schützt vor Angriffsart				
Paper (Name des Protokolls/ Konzepts)	eingesetzte Schutztechnik	Protokollart	Einweg-Hashfkt.	symmetr. Krypto	asymm. Krypto	digitale Signatur	Commitment	Abhören	MitM	Replay	Reordering	Battery-Draining
[Xu+11] (IMDGuard)	phys. Signal ext. Gerät	Schlüsselerzeugung Authentifikation	SHA-1	AES-128	✓	PKK als Signatur	✗	Verschl.	PKK	Noncen	Noncen	✗
[DFK08] (Cloaker)	ext. Gerät	Authentifikation	✗	✓	✓	✗	✗	Verschl.	✗	✗	✗	(✓)
[Gol+11] (Shield)	ext. Gerät	✗	✗	?	?	✗	✗	(Jamming) Verschl.	?	✗	✗	✗
[Kul17] (secure belt)	ext. Gerät	✗	✗	✗	✗	✗	✗	Jamming	✗	✗	✗	✗
[Kul19] (security jacket)	ext. Gerät	✗	✗	✗	✗	✗	✗	Jamming	✗	✗	✗	✗
[ZRJ13] (Medmon)	ext. Gerät Anomalieerk.	✗	✗	✗	✗	✗	✗	✗	Anomalie-erkennung	Anomalie-erkennung	Anomalie-erkennung	✗
[Hal+08a]	ZPD OOB	Authentifikation	✗	RC5	✗	✗	✗	Ultraschall 4 kHz, Verschl.	✗	Noncen	✗	RFID
[LAK10]	ZPD	Authentifikation	✗	AES-CBC	✗	✗	✗	Verschl.	✗	Counter	Counter	RFID
[Str+13]	ZPD	Authentifikation	✗	MISTY1	✗	✗	✗	Verschl.	✗	Noncen	Noncen	RFID
[Yan+14b]	ZPD	Authentifikation	✓	✓	✗	✗	✗	✗	✗	Noncen	Noncen	RFID
[Ell+13] [Ell+18]	phys. Signal ZPD	Schlüsselerzeugung Authentifikation	✓	AES	✗	✗	✗	✗	✗	Noncen	Noncen	RFID
[Sid+21] (SecureEcho)	ZPD OOB	Schlüsselaustausch	✗	✗	✗	✗	✗	OOB	OOB	OOB	OOB	✗
[Hei+10]	Anomalieerk.	✗	✗	✗	✗	✗	✗	✗	Anomalie-erkennung	Anomalie-erkennung	Anomalie-erkennung	✗
[Hen+13]	Anomalieerk.	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
[Hei+15b] (PIPAC)	Anomalieerk.	keine	✗	✗	✗	✗	✗	✗	Anomalie-erkennung	Anomalie-erkennung	Anomalie-erkennung	✗
[GT17]	ext. Gerät Anomalieerk.	✗	✗	✗	✗	✗	✗	✗	Anomalie-erkennung	Anomalie-erkennung	Anomalie-erkennung	✗
[Rat+17] [Rat+18]	Anomalieerk.	✗	✗	✗	✗	✗	✗	✗	Anomalie-erkennung	Anomalie-erkennung	Anomalie-erkennung	✗
[Ras+09]	Distance-Bounding OOB	Schlüsselerzeugung	✗	✗	DH	✗	✗	OOB	OOB	OOB Noncen	OOB Noncen	✗
[Jur+11]	phys. Signal Distance-Bounding	Nachrichten-austausch	✗	✗	✗	✗	✗	Verschl.	✗	✗	✗	✗
[Shi+13]	Distance-Bounding	Authentifikation	✗	✗	✗	✗	✗	✗	Distanz-messung	Distanz-messung	Distanz-messung	✗

			verwendete Krypto-Primitive					Protokoll/Methode schützt vor Angriffsart				
Paper (Name des Protokolls/ Konzepts)	eingesetzte Schutztechnik	Protokollart	Einweg-Hashfkt.	symmetr. Krypto	asymm. Krypto	digitale Signatur	Commitment	Abhören	MitM	Replay	Reordering	Battery-Draining
[Cam+21] (ACIMD)	phys. Signel OOB	Schlüsselaustausch Authentifikation	✓	?	?	✗	✗	OOB	OOB	OOB Noncen	OOB Noncen	✗
[CVG03] (BioSec)	phys. Signal	Schlüsselaustausch Nachrichten-austausch	✓	RC5	✗	MD5	Fuzzy Commitment	Verschl.	✗	Live-PS	Live-PS	✗
[Ros+13b] (H2H)	phys. Signal	Authentifikation	✗	✗	✗	✗	✓	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[Hu+13] (OPFKA)	phys. Signal	Schlüsselerzeugung Authentifikation	✓	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[BSZ04]	phys. Signal	Schlüsselaustausch Nachrichten-austausch	✓	✓	✗	✗	Fuzzy Commitment	Verschl.	✗	Live-PS	Live-PS	✗
[BH07]	phys. Signal	Schlüsselaustausch	SHA-1	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[VBG08] (EKA)	phys. Signal	Schlüsselerzeugung	✓	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Ven+08] (PKA)	phys. Signal	Schlüsselerzeugung	✗	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[Bao+08]	phys. Signal	Schlüsselerzeugung	✗	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Mia+09]	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Ven+10] (PSKA)	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[Yao+11] (ESKE)	phys. Signal	Schlüsselaustausch	✓	✓	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Yao+11]	phys. Signal	Schlüsselaustausch	✓	✓	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[Raj+12]	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[Zha+12] (ECG-IJS)	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[ZRJ14] (ESDS)	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[Zhe+15] (EDE)	phys. Signal	Schlüsselaustausch Nachrichten-austausch	✓	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[Zag+15] (ELPA)	phys. Signal	Schlüsselaustausch	✗	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Mar+16a]	phys. Signal	Schlüsselerzeugung	✓	✗	✗	✗	✗	✗	✗	Live-PS Noncen	Live-PS Noncen	✗
[KA18] (SGenP)	phys. Signal	Schlüsselerzeugung	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗

Paper (Name des Protokolls/ Konzepts)	eingesetzte Schutztechnik	Protokollart	verwendete Krypto-Primitive					Protokoll/Methode schützt vor Angriffsart				
			Einweg-Hashfkt.	symmetr. Krypto	asymm. Krypto	digitale Signatur	Commitment	Abhören	MitM	Replay	Reordering	Battery-Draining
[Bai+18]	phys. Signal	Schlüsselaustausch	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
[KM19] (ESKG)	phys. Signal	Schlüsselaustausch	✗	secure force [EC13]	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[KSL19] (SKA-PS)	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Sey19] (SKA-PSAR)	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Lin+19] (H2B)	phys. Signal	Schlüsselaustausch	✓	✓	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Zhe+19] (F2H)	phys. Signal	Authentifikation	✗	✓	✗	✗	✗	Verschl.	✗	✗	✗	✗
[Bel+19]	phys. Signal	Authentifikation	✗	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[BZS21] (MEDISCOM)	phys. Signal	Schlüsselaustausch Authentifikation	✗	✓	✗	✗	✗	Verschl.	✗	Sequenznr.	Sequenznr.	✗
[Zha+21a] (H2K)	phys. Signal	Schlüsselaustausch	✓	✗	✗	✗	✗	✗	✗	Live-PS	Live-PS	✗
[Kim+15] (SecureVibe)	OOB	Schlüsselaustausch	✗	✗	✗	✗	✗	OOB	OOB	OOB	OOB	Low-Pwr Vibr.-sensor

Tabelle 9.1: IMD Security Bedrohungen mit Abwehrmaßnahmen und eingesetzte Krypto-Primitive

Literatur

[16] *MW Is Short St. Jude Medical (STJ:US)*. Muddy Waters Research. 2016. URL: https://www.muddywatersresearch.com/research/stj/mw-is-short-stj/ (besucht am 12. 02. 2023).

[22] *Barnaby Jack*. In: *Wikipedia*. 28. Nov. 2022. URL: https://en.wikipedia.org/w/index.php?title=Barnaby_Jack&oldid=1124351661 (besucht am 13. 03. 2023).

[Abe+03] Thomas Abell u. a. „Gastric Electrical Stimulation for Medically Refractory Gastroparesis“. In: *Gastroenterology* 125.2 (1. Jan. 2003), S. 421–428. ISSN: 00165085. DOI: 10.1016/S0016-5085(03)00878-3.

[Avo+18] Gildas Avoine u. a. „Security of Distance-Bounding: A Survey“. In: *ACM Computing Surveys* 51.5 (25. Sep. 2018), S. 94. DOI: 10.1145/3264628.

[AY16] Riham Altawy und Amr M. Youssef. „Security Tradeoffs in Cyber Physical Systems: A Case Study Survey on Implantable Medical Devices“. In: *IEEE access : practical innovations, open solutions* 4 (1. Jan. 2016), S. 959–979. DOI: 10.1109/ACCESS.2016.2521727. pmid: null.

[Bag+13] Priyanka Bagade u. a. „Protect Your BSN: No Handshakes, Just Namaste!“ In: *2013 IEEE International Conference on Body Sensor Networks*. 2013 IEEE International Conference on Body Sensor Networks. Mai 2013, S. 1–6. DOI: 10.1109/BSN.2013.6575512.

[Bai+18] Tong Bai u. a. „A Lightweight Method of Data Encryption in BANs Using Electrocardiogram Signal“. In: *Future Generation Computer Systems* 92 (31. Jan. 2018), S. 800–811. DOI: 10.1016/j.future.2018.01.031. pmid: null.

[Bao+08] Shu-Di Bao u. a. „Using the Timing Information of Heartbeats as an Entity Identifier to Secure Body Sensor Network“. In: *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society* 12.6 (1. Jan. 2008), S. 772–779. DOI: 10.1109/TITB.2008.926434. pmid: 19000958.

[Bar07] Richard G. Baraniuk. „Compressive Sensing [Lecture Notes]“. In: *IEEE Signal Processing Magazine* 24.4 (Juli 2007), S. 118–121. ISSN: 1558-0792. DOI: 10.1109/MSP.2007.4286571.

[Bel+19] Taha Belkhouja u. a. „Biometric-Based Authentication Scheme for Implantable Medical Devices during Emergency Situations“. In: *Future Generation Computer Systems* 98 (1. Sep. 2019), S. 109–119. ISSN: 0167-739X. DOI: 10.1016/j.future.2019.02.002. URL: https://www.sciencedirect.com/science/article/pii/S0167739X18325792 (besucht am 27. 02. 2023).

[Ben03] Alim-Louis Benabid. „Deep Brain Stimulation for Parkinson's Disease.“ In: *Current Opinion in Neurobiology* 13.6 (1. Dez. 2003), S. 696–706. DOI: 10.1016/j.conb.2003.11.001. pmid: 14662371.

[BH07] Francis Minhthang Bui und Dimitrios Hatzinakos. „Biometric Methods for Secure Communications in Body Sensor Networks: Resource-Efficient Key Management and Signal-Level Data Scrambling“. In: *EURASIP Journal on Advances in Signal Processing* 2008 (2007), S. 1–16.

[BJH16] A. J. Burns u. a. „A Brief Chronology of Medical Device Security“. In: *Communications of the ACM* 59.10 (1. Jan. 2016), S. 66–72. ISSN: 0001-0782. DOI: 10.1145/2890488.

[Bow22] Stephen Bowditc. *Cochlear Implant Surgery and Rehabilitation*. 30. Nov. 2022. URL: https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/cochlear-implant-surgery (besucht am 20. 01. 2023).

[BSZ04] Shu-Di Bao u. a. „A Novel Key Distribution of Body Area Networks for Telemedicine“. In: *BCS International Academic Conference* (1. Dez. 2004), S. 1–17. DOI: 10.1109/biocas.2004.1454091.

[BZS21] Nils Beck u. a. „BCG & ECG-based Secure Communication for Medical Devices in Body Area Networks“. In: *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)* (22. März 2021), S. 207–212. DOI: 10.1109/percomworkshops51409.2021.9430964.

[Cam+21] Carmen Camara u. a. „Access Control for Implantable Medical Devices“. In: *IEEE Transactions on Emerging Topics in Computing* 9.3 (1. Jan. 2021), S. 1126–1138. DOI: 10.1109/TETC.2020.2982461.

[CFM10] Raffaele Cappelli u. a. „Minutia Cylinder-Code: A New Representation and Matching Technique for Fingerprint Recognition“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.12 (Dez. 2010), S. 2128–2141. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2010.52.

[Cha+12] Sang-Yoon Chang u. a. „Body Area Network Security: Robust Key Establishment Using Human Body Channel“. In: (6. Aug. 2012), S. 5–5.

[Chi+13] Girish Chitnis u. a. „A Minimally Invasive Implantable Wireless Pressure Sensor for Continuous IOP Monitoring“. In: *IEEE Transactions on Biomedical Engineering* 60.1 (1. Jan. 2013), S. 250–256. DOI: 10.1109/tbme.2012.2205248. pmid: 22736631.

[CIS17] CISA. *Abbott Laboratories' Accent/Anthem, Accent MRI, Assurity/Allure, and Assurity MRI Pacemaker Vulnerabilities | CISA*. 2017. URL: https://www.cisa.gov/uscert/ics/advisories/ICSMA-17-241-01 (besucht am 08. 02. 2023).

[CIS18] CISA. *Medtronic MyCareLink Patient Monitor | CISA*. 2018. URL: https://www.cisa.gov/uscert/ics/advisories/ICSMA-18-179-01 (besucht am 08. 02. 2023).

[CIS21] CISA. *Medtronic Conexus Radio Frequency Telemetry Protocol (Update C) | CISA*. 2021. URL: https://www.cisa.gov/uscert/ics/advisories/ICSMA-19-080-01 (besucht am 08. 02. 2023).

[Clu+06] Jolyon Clulow u. a. „So Near and Yet So Far: Distance-Bounding Attacks in Wireless Networks“. In: *Security and Privacy in Ad-Hoc and Sensor Networks*. Hrsg. von Levente Buttyán u. a. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2006, S. 83–97. ISBN: 978-3-540-69173-0. DOI: 10.1007/11964254_9.

[CPT15] Alejandro Calleja u. a. „Electrical Heart Signals Can Be Monitored from the Moon: Security Implications for IPI-Based Protocols“. In: *null* (24. Aug. 2015), S. 36–51. DOI: 10.1007/978-3-319-24018-3_3. pmid: null.

[Cre+12] Cas Cremers u. a. „Distance Hijacking Attacks on Distance Bounding Protocols“. In: *IEEE Symposium on Security and Privacy* (20. Mai 2012), S. 113–127. DOI: 10.1109/sp.2012.17.

[CVG03] S. Cherukuri u. a. „Biosec: A Biometric Based Approach for Securing Communication in Wireless Networks of Biosensors Implanted in the Human Body“. In: *2003 International Conference on Parallel Processing Workshops, 2003. Proceedings.* IEEE Comput. Soc, 1. Jan. 2003, S. 432–439. ISBN: 0-7695-2018-9. DOI: 10.1109/ICPPW.2003.1240399.

[DD03] John P. DiMarco und John P. DiMarco. „Implantable Cardioverter–Defibrillators“. In: *The New England Journal of Medicine* 349.19 (6. Nov. 2003), S. 1836–1847. DOI: 10.1056/nejmra035432. pmid: 14602883.

[DFK08] Tamara Denning u. a. „Absence Makes the Heart Grow Fonder: New Directions for Implantable Medical Device Security“. In: *Proceedings of the 3rd Conference on Hot Topics in Security.* HOTSEC'08. USA: USENIX Association, 1. Jan. 2008.

[DRS04] Yevgeniy Dodis u. a. „Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data“. In: *Advances in Cryptology - EUROCRYPT 2004.* Hrsg. von Christian Cachin und Jan L. Camenisch. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2004, S. 523–540. ISBN: 978-3-540-24676-3. DOI: 10.1007/978-3-540-24676-3_31.

[EC13] Mansoor Ebrahim und Chai Wai Chong. „Secure Force: A Low-Complexity Cryptographic Algorithm for Wireless Sensor Network (WSN)“. In: *2013 IEEE International Conference on Control System, Computing and Engineering.* 2013 IEEE International Conference on Control System, Computing and Engineering. Nov. 2013, S. 557–562. DOI: 10.1109/ICCSCE.2013.6720027.

[Ell+13] Nourhene Ellouze u. a. „Securing Implantable Cardiac Medical Devices“. In: *Proceedings of the 3rd International Workshop on Trustworthy Embedded Devices - TrustED '13.* Hrsg. von Ahmad-Reza Sadeghi u. a. New York, New York, USA: ACM Press, 1. Jan. 2013, S. 35–42. ISBN: 978-1-4503-2486-1. DOI: 10.1145/2517300.2517307.

[Ell+18] Nourhene Ellouze u. a. „Powerless Security for Cardiac Implantable Medical Devices: Use of Wireless Identification and Sensing Platform“. In: *Journal of Network and Computer Applications* 107 (1. Jan. 2018), S. 1–21. ISSN: 10848045. DOI: 10.1016/j.jnca.2018.01.009. pmid: null.

[ER89] M. W. Eichin und J. A. Rochlis. „With Microscope and Tweezers: An Analysis of the Internet Virus of November 1988“. In: *Proceedings. 1989 IEEE Symposium on Security and Privacy.* IEEE Comput. Soc. Press, 1. Jan. 1989, S. 326–343. ISBN: 0-8186-1939-2. DOI: 10.1109/SECPRI.1989.36307.

[Fer19] Manny Fernandez. *Epilepsy Foundation Was Targeted in Mass Strobe Cyberattack - The New…* archive.is. 17. Dez. 2019. URL: https://archive.is/1pXWV (besucht am 08. 02. 2023).

[Fu+18] Chenglong Fu u. a. „POKs Based Low Energy Authentication Scheme for Implantable Medical Devices.“ In: *arXiv: Cryptography and Security* (27. März 2018).

[Gol+00] Ary L. Goldberger u. a. „PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals“. In: *Circulation* 101.23 (13. Juni 2000), S. 215–220. ISSN: 0009-7322, 1524-4539. DOI: 10.1161/01.cir.101.23.e215. pmid: 10851218. URL: https://www.ahajournals.org/doi/10.1161/01.CIR.101.23.e215.

[Gol+11] Shyamnath Gollakota u. a. „They Can Hear Your Heartbeats: Non-Invasive Security for Implantable Medical Devices“. In: *Proceedings of the ACM SIGCOMM 2011 Conference*. SIGCOMM '11. New York, NY, USA: Association for Computing Machinery, 1. Jan. 2011, S. 2–13. ISBN: 978-1-4503-0797-0. DOI: 10.1145/2018436.2018438.

[GT17] Sida Gao und Geethapriya Thamilarasu. „Machine-Learning Classifiers for Security in Connected Medical Devices“. In: *2017 26th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 1. Jan. 2017, S. 1–5. ISBN: 978-1-5090-2991-4. DOI: 10.1109/ICCCN.2017.8038507.

[Gup12] Sarbari Gupta. *CSRC Presentation: Implantable Medical Devices - Cyber Risks and Mitigation | CSRC*. CSRC | NIST. 2012. URL: https://csrc.nist.gov/presentations/2012/implantable-medical-devices-cyber-risks-and-miti (besucht am 08. 02. 2023).

[Hag18] Jonathen Hagedorn. *A Review of Neuromodulation Advancements*. 2018. URL: https://www.asra.com/guidelines-articles/original-articles/professional-issues/professional-issues/legacy-b-blog-posts/2018/02/07/a-review-of-neuromodulation-advancements (besucht am 14. 01. 2023).

[Hal+08a] Daniel Halperin u. a. „Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses“. In: *2008 IEEE Symposium on Security and Privacy (Sp 2008)*. IEEE, 1. Jan. 2008, S. 129–142. ISBN: 978-0-7695-3168-7. DOI: 10.1109/SP.2008.31.

[Hal+08b] Daniel Halperin u. a. „Security and Privacy for Implantable Medical Devices“. In: *IEEE Pervasive Computing* 7.1 (1. Jan. 2008), S. 30–39. ISSN: 1536-1268. DOI: 10.1109/MPRV.2008.16. pmid: null.

[Hau+00] Morten Kristian Haugland u. a. „An Implantable Foot Drop Stimulator“. In: (1. Jan. 2000).

[Hei+10] Xiali Hei u. a. „Defending Resource Depletion Attacks on Implantable Medical Devices“. In: *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*. IEEE, 1. Jan. 2010, S. 1–5. ISBN: 978-1-4244-5636-9. DOI: 10.1109/GLOCOM.2010.5685228.

[Hei+15a] Xiali Hei u. a. „Patient Infusion Pattern Based Access Control Schemes for Wireless Insulin Pump System“. In: *IEEE Transactions on Parallel and Distributed Systems* 26.11 (1. Jan. 2015), S. 3108–3121. ISSN: 1045-9219. DOI: 10.1109/TPDS.2014.2370045. pmid: null.

[Hei+15b] Xiali Hei u. a. „Patient Infusion Pattern Based Access Control Schemes for Wireless Insulin Pump System“. In: *IEEE Transactions on Parallel and Distributed Systems* 26.11 (Nov. 2015), S. 3108–3121. ISSN: 1558-2183. DOI: 10.1109/TPDS.2014.2370045.

[Hen+13] Nathan L. Henry u. a. „Using Bowel Sounds to Create a Forensically-Aware Insulin Pump System“. In: *HealthTech* (12. Aug. 2013), S. 8–8.

[HS10] Tzipora Halevi und Nitesh Saxena. „On Pairing Constrained Wireless Devices Based on Secrecy of Auxiliary Channels: The Case of Acoustic Eavesdropping“. In: *null* (4. Okt. 2010), S. 97–108. DOI: 10.1145/1866307.1866319. pmid: null.

[Hu+13] Chunqiang Hu u. a. „OPFKA: Secure and Efficient Ordered-Physiological-Feature-Based Key Agreement for Wireless Body Area Networks“. In: *null* (14. Apr. 2013), S. 2274–2282. DOI: 10.1109/infcom.2013.6567031. pmid: null.

[IA20] Institut für Theoretische Informatik und Arbeitsgruppe für Kryptographie und Sicherheit. *Skript Zur Stammvorlesung Sicherheit*. 2020.

[IY16] Mohd Noor Islam und Mehmet R. Yuce. „Review of Medical Implant Communication System (MICS) Band and Network“. In: *ICT Express* 2.4 (1. Jan. 2016), S. 188–194. ISSN: 24059595. DOI: 10.1016/j.icte.2016.08.010.

[JS06] Ari Juels und Madhu Sudan. „A Fuzzy Vault Scheme“. In: *Designs, Codes and Cryptography* 38.2 (1. Feb. 2006), S. 237–257. ISSN: 1573-7586. DOI: 10.1007/s10623-005-6343-z. URL: https://doi.org/10.1007/s10623-005-6343-z (besucht am 15. 02. 2023).

[Jue02] Ari Juels. „A Fuzzy Vault Scheme“. In: *Proceedings of the 2002 IEEE International Symposium on Information Theory*. Bd. 408. 2002.

[Jur+11] Andrew D. Jurik u. a. „Securing Mobile Devices with Biotelemetry“. In: *2011 Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)* (30. Aug. 2011), S. 1–6. DOI: 10.1109/icccn.2011.6006008.

[JW99] Ari Juels und Martin Wattenberg. „A Fuzzy Commitment Scheme“. In: *Proceedings of the 6th ACM Conference on Computer and Communications Security*. CCS '99. New York, NY, USA: Association for Computing Machinery, 1. Nov. 1999, S. 28–36. ISBN: 978-1-58113-148-2. DOI: 10.1145/319709.319714. URL: https://doi.org/10.1145/319709.319714 (besucht am 15. 02. 2023).

[KA18] Priti Kumari und Tricha Anjali. „Symmetric-Key Generation Protocol (SGenP) for Body Sensor Network“. In: *2018 IEEE International Conference on Communications Workshops (ICC Workshops)* (20. Mai 2018), S. 1–6. DOI: 10.1109/iccw.2018.8403548.

[KC22] Emmanuel Kwarteng und Mumin Cebe. „A Survey on Security Issues in Modern Implantable Devices: Solutions and Future Issues“. In: *Smart Health* 25 (1. Jan. 2022), S. 100295. ISSN: 23526483. DOI: 10.1016/j.smhl.2022.100295.

[Kha+20] Sadeque Reza Khan u. a. „Wireless Power Transfer Techniques for Implantable Medical Devices: A Review“. In: *Sensors (Basel, Switzerland)* 20.12 (1. Jan. 2020). DOI: 10.3390/s20123487. pmid: 32575663.

[KI09] Aravind Kailas und Mary Ann Ingram. „Wireless Communications Technology in Telehealth Systems“. In: *2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology*. 2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology. Mai 2009, S. 926–930. DOI: 10.1109/WIRELESSVITAE.2009.5172574.

[Kim+15] Younghyun Kim u. a. „Vibration-Based Secure Side Channel for Medical Devices“. In: *Proceedings of the 52nd Annual Design Automation Conference*. New York, NY, USA: ACM, 1. Jan. 2015, S. 1–6. ISBN: 978-1-4503-3520-1. DOI: 10.1145/2744769.2744928.

[KM19] M. V. Karthikeyan und J. Martin Leo Manickam. „ECG-Signal Based Secret Key Generation (ESKG) Scheme for WBAN and Hardware Implementation“. In: *Wireless Personal Communications* 106.4 (1. Juni 2019), S. 2037–2052. DOI: 10.1007/s11277-018-5924-x.

[KSL19] Duygu Karaoğlan Altop u. a. „SKA-PS: Secure Key Agreement Protocol Using Physiological Signals“. In: *Ad Hoc Networks* 83 (1. Feb. 2019), S. 111–124. ISSN: 1570-8705. DOI: 10.1016/j.adhoc.2018.09.003. URL: https://www.sciencedirect.com/science/article/pii/S1570870518306425 (besucht am 23. 02. 2023).

[Kul17] Selman Kulaç. „Security Belt for Wireless Implantable Medical Devices“. In: *Journal of medical systems* 41.11 (1. Jan. 2017), S. 172. DOI: 10.1007/s10916-017-0813-5. pmid: 28929373.

[Kul19] Selman Kulac. „A New Externally Worn Proxy-Based Protector for Non-Secure Wireless Implantable Medical Devices: Security Jacket“. In: *IEEE Access* 7 (1. Jan. 2019), S. 55358–55366. DOI: 10.1109/ACCESS.2019.2910029.

[Kwo+12] Sungjun Kwon u. a. „Validation of Heart Rate Extraction Using Video Imaging on a Built-in Camera System of a Smartphone“. In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 2012 (12. Nov. 2012), S. 2174–2177. DOI: 10.1109/embc.2012.6346392. pmid: 23366353.

[LAK10] Jing-Wei LIU u. a. „Secure Wake-Up Scheme for WBANs“. In: *IEICE Transactions on Communications* E93-B.4 (1. Jan. 2010), S. 854–857. ISSN: 0916-8516. DOI: 10.1587/transcom.E93.B.854.

[Lar+03] Berit Larsson u. a. „Lessons from the First Patient with an Implanted Pacemaker: 1958-2001“. In: *Pacing and clinical electrophysiology : PACE* 26 (1 Pt 1 1. Jan. 2003), S. 114–24. ISSN: 0147-8389. DOI: 10.1046/j.1460-9592.2003.00162.x. pmid: 12685152.

[Lin+19] Qi Lin u. a. „H2B: Heartbeat-Based Secret Key Generation Using Piezo Vibration Sensors“. In: *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*. Hrsg. von Rasit Eskicioglu u. a. New York, NY, USA: ACM, 1. Jan. 2019, S. 265–276. ISBN: 978-1-4503-6284-9. DOI: 10.1145/3302506.3310406.

[LRJ11] Chunxiao Li u. a. „Hijacking an Insulin Pump: Security Attacks and Defenses for a Diabetes Therapy System“. In: *2011 IEEE 13th International Conference on E-Health Networking, Applications and Services*. IEEE, 1. Jan. 2011, S. 150–156. ISBN: 978-1-61284-695-8. DOI: 10.1109/HEALTH.2011.6026732.

[Mar+16a] Eduard Marin u. a. „A Privacy-Preserving Remote Healthcare System Offering End-to-End Security“. In: *Ad-Hoc, Mobile, and Wireless Networks*. Hrsg. von Nathalie Mitton u. a. Lecture Notes in Computer Science. Cham: Springer International Publishing, 1. Jan. 2016, S. 237–250. ISBN: 978-3-319-40508-7. DOI: 10.1007/978-3-319-40509-4_17.

[Mar+16b] Eduard Marin u. a. *A Survey on Physiological-Signal-Based Security for Medical Devices*. 1. Jan. 2016. URL: https://eprint.iacr.org/2016/867.

[Mar+16c] Eduard Marin u. a. „On the (in)Security of the Latest Generation Implantable Cardiac Defibrillators and How to Secure Them“. In: *Proceedings of the 32nd Annual Conference on Computer Security Applications*. Hrsg. von Stephen Schwab u. a. New York, NY, USA: ACM, 1. Jan. 2016, S. 226–236. ISBN: 978-1-4503-4771-6. DOI: 10.1145/2991079.2991094.

[Mar+16d] Eduard Marin u. a. „On the Feasibility of Cryptography for a Wireless Insulin Pump System“. In: *Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy*. Hrsg. von Elisa Bertino u. a. New York, NY, USA: ACM, 1. Jan. 2016, S. 113–120. ISBN: 978-1-4503-3935-3. DOI: 10.1145/2857705.2857746.

[Mar+18] Eduard Marin u. a. „Securing Wireless Neurostimulators“. In: *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy*. Hrsg. von Ziming Zhao u. a. New York, NY, USA: ACM, 1. Jan. 2018, S. 287–298. ISBN: 978-1-4503-5632-9. DOI: 10.1145/3176258.3176310.

[Mar18] Eduard Marin. „Security and Privacy of Implantable Medical Devices“. In: (1. Jan. 2018).

[Mia+09] Fen Miao u. a. „Biometrics Based Novel Key Distribution Solution for Body Sensor Networks“. In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 2009 (13. Nov. 2009), S. 2458–2461. DOI: 10.1109/iembs.2009.5334698. pmid: 19964960.

[Mil08] Elinor Mills. *Attack on Epilepsy Web Site Prompts Migraines, near Seizures*. CNET. 2008. URL: https://www.cnet.com/culture/attack-on-epilepsy-web-site-prompts-migraines-near-seizures/ (besucht am 08. 02. 2023).

[MO00] Mitsuru Matsui und Hidenori Ohta. *A Description of the MISTY1 Encryption Algorithm*. Request for Comments RFC 2994. Internet Engineering Task Force, Nov. 2000. 10 S. DOI: 10.17487/RFC2994. URL: https://datatracker.ietf.org/doc/rfc2994 (besucht am 05. 03. 2023).

[Nuñ18] Mª Carmen Camara Nuñez. „Cybersecurity in Implantable Medical Devices“. In: (1. Jan. 2018).

[OPP20] Lara Ortiz-Martin u. a. „Are the Interpulse Intervals of an ECG Signal a Good Source of Entropy? An in-Depth Entropy Analysis Based on NIST 800-90B Recommendation“. In: *Future Generation Computer Systems* 105 (1. Jan. 2020), S. 346–360. ISSN: 0167739X. DOI: 10.1016/j.future.2019.12.002. pmid: null.

[Ort+19] Lara Ortiz-Martin u. a. „Feasibility Analysis of Inter-Pulse Intervals Based Solutions for Cryptographic Token Generation by Two Electrocardiogram Sensors“. In: *Future Generation Computer Systems* 96 (Juli 2019), S. 283–296. ISSN: 0167739X. DOI: 10.1016/j.future.2019.02.021. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167739X18330784 (besucht am 27. 02. 2023).

[Par+94] B. Park u. a. „Predicting Intramuscular Fat in Beef Longissimus Muscle from Speed of Sound“. In: *Journal of Animal Science* 72.1 (1. Jan. 1994), S. 109–116. ISSN: 0021-8812, 1525-3163. DOI: 10.2527/1994.721109x. URL: https://academic.oup.com/jas/article/72/1/109-116/4632483 (besucht am 10. 03. 2023).

[Par12] Arundhati Parmar. *Hacking Wireless Insulin Pumps*. MedCity News. 1. März 2012. URL: https://medcitynews.com/2012/03/hacker-shows-off-vulnerabilities-of-wireless-insulin-pumps/ (besucht am 13. 03. 2023).

[PMP11] Ming-Zher Poh u. a. „Advancements in Noncontact, Multiparameter Physiological Measurements Using a Webcam“. In: *IEEE Transactions on Biomedical Engineering* 58.1 (1. Jan. 2011), S. 7–11. DOI: 10.1109/tbme.2010.2086456. pmid: 20952328.

[PZB06] C.C.Y. Poon u. a. „A Novel Biometrics Method to Secure Wireless Body Area Sensor Networks for Telemedicine and M-Health“. In: *IEEE Communications Magazine* 44.4 (1. Jan. 2006), S. 73–81. ISSN: 0163-6804. DOI: 10.1109/MCOM.2006.1632652. pmid: null.

[Qui86] J. R. Quinlan. „Induction of Decision Trees“. In: *Machine Learning* 1.1 (1. März 1986), S. 81–106. ISSN: 1573-0565. DOI: 10.1007/BF00116251. URL: https://doi.org/10.1007/BF00116251 (besucht am 10. 03. 2023).

[Raj+12] R. Thalapathi Rajasekaran u. a. „An Efficient and Secure Key Agreement Scheme Using Physiological Signals in Body Area Networks“. In: *International Conference on Advances in Computing, Communications and Informatics* (3. Aug. 2012), S. 1143–1147. DOI: 10.1145/2345396.2345579.

[Ras+09] Kasper Bonne Rasmussen u. a. „Proximity-Based Access Control for Implantable Medical Devices“. In: *Proceedings of the 16th ACM Conference on Computer and Communications Security - CCS '09*. Hrsg. von Ehab Al-Shaer u. a. New York, New York, USA: ACM Press, 1. Jan. 2009, S. 410. ISBN: 978-1-60558-894-0. DOI: 10.1145/1653662.1653712.

[Rat+17] Heena Rathore u. a. „DLRT: Deep Learning Approach for Reliable Diabetic Treatment“. In: *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*. GLOBECOM 2017 - 2017 IEEE Global Communications Conference. Dez. 2017, S. 1–6. DOI: 10.1109/GLOCOM.2017.8255028.

[Rat+18] Heena Rathore u. a. „Multi-Layer Perceptron Model on Chip for Secure Diabetic Treatment“. In: *IEEE Access* 6 (1. Jan. 2018), S. 44718–44730. DOI: 10.1109/ACCESS.2018.2854822.

[Rio17] Billy Rios. *Security Evaluation of the Implantable Cardiac Device Ecosystem Architecture and Implementation Interdependencies*. 2017.

[Ros+13a] Masoud Rostami u. a. „Balancing Security and Utility in Medical Devices?“ In: *Proceedings of the 50th Annual Design Automation Conference on - DAC '13*. Hrsg. von Unknown. New York, New York, USA: ACM Press, 1. Jan. 2013, S. 1. ISBN: 978-1-4503-2071-9. DOI: 10.1145/2463209.2488750.

[Ros+13b] Masoud Rostami u. a. „Heart-to-Heart (H2H): Authentication for Implanted Medical Devices“. In: (4. Nov. 2013), S. 1099–1112. DOI: 10.1145/2508859.2516658.

[Rus+14] Michael Rushanan u. a. „SoK: Security and Privacy in Implantable Medical Devices and Body Area Networks“. In: *2014 IEEE Symposium on Security and Privacy*. IEEE, 1. Jan. 2014, S. 524–539. ISBN: 978-1-4799-4686-0. DOI: 10.1109/SP.2014.40.

[SA00] Frank Stajano und Ross Anderson. „The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks“. In: *Security Protocols*. Hrsg. von Gerhard Goos u. a. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1. Jan. 2000, S. 172–182. ISBN: 978-3-540-67381-1. DOI: 10.1007/10720107_24.

[Sch10] Stuart Schechter. *Security That Is Meant to Be Skin Deep: Using Ultraviolet Micropigmentation to Store Emergency-Access Keys for Implantable Medical Devices*. 1. Jan. 2010. URL: https://www.microsoft.com/en-us/research/publication/security-that-is-meant-to-be-skin-deep-using-ultraviolet-micropigmentation-to-store-emergency-access-keys-for-implantable-medical-devices/.

[SE09] Richard K. Shepard und Kenneth A. Ellenbogen. „Leads and Longevity: How Long Will Your Pacemaker Last?“ In: *EP Europace* 11.2 (1. Feb. 2009), S. 142–143. ISSN: 1099-5129. DOI: 10.1093/europace/eun359. URL: https://academic.oup.com/europace/article/11/2/142/520275 (besucht am 19. 01. 2023).

[Sey19] Beste Seymen. „On the Establishment of PSEUDO Random Keys for Body Area Network Security Using Physiological Signals“. In: (4. Jan. 2019).

[Shi+09] Scott A. Shikora u. a. „Implantable Gastric Stimulation for the Treatment of Clinically Severe Obesity: Results of the SHAPE Trial.“ In: *Surgery for Obesity and Related Diseases* 5.1 (1. Jan. 2009), S. 31–37. DOI: 10.1016/j.soard.2008.09.012. pmid: 19071066.

[Shi+13] Lu Shi u. a. „BANA: Body Area Network Authentication Exploiting Channel Characteristics“. In: *IEEE Journal on Selected Areas in Communications* 31.9 (26. Aug. 2013), S. 1803–1816. DOI: 10.1109/jsac.2013.130913.

[Sid+21] Muhammad Ali Siddiqi u. a. „Securing Implantable Medical Devices Using Ultrasound Waves“. In: *IEEE access : practical innovations, open solutions* 9 (1. Jan. 2021), S. 80170–80182. DOI: 10.1109/ACCESS.2021.3083576. pmid: null.

[Smi+06] Joshua R. Smith u. a. „A Wirelessly-Powered Platform for Sensing and Computation“. In: *UbiComp 2006: Ubiquitous Computing*. Hrsg. von Paul Dourish und Adrian Friday. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2006, S. 495–506. ISBN: 978-3-540-39635-2. DOI: 10.1007/11853565_29.

[Str+13] Christos Strydis u. a. „A System Architecture, Processor, and Communication Protocol for Secure Implants“. In: *ACM Transactions on Architecture and Code Optimization* 10.4 (1. Jan. 2013), S. 1–23. ISSN: 15443566. DOI: 10.1145/2555289.2555313.

[Tan+17] A. Tantin u. a. „Implantable MICS-based Wireless Solution for Bladder Pressure Monitoring“. In: *2017 IEEE Biomedical Circuits and Systems Conference (BioCAS)* (1. Okt. 2017), S. 1–4. DOI: 10.1109/biocas.2017.8325205.

[Tar+14] Lionel Tarassenko u. a. „Non-Contact Video-Based Vital Sign Monitoring Using Ambient Light and Auto-Regressive Models“. In: *Physiological Measurement* (2014). DOI: 10.1088/0967-3334/35/5/807. pmid: 24681430.

[VBG08] Krishna Kumar Venkatasubramanian u. a. „EKG-based Key Agreement in Body Sensor Networks“. In: (1. Jan. 2008). Unter Mitarb. von Institute of Electrical and Electronics Engineers.

[Ven+08] Krishna K. Venkatasubramanian u. a. „Plethysmogram-Based Secure Inter-Sensor Communication in Body Area Networks“. In: (1. Nov. 2008), S. 1–7. DOI: 10.1109/milcom.2008.4753199.

[Ven+10] Krishna K. Venkatasubramanian u. a. „PSKA: Usable and Secure Key Agreement Scheme for Body Area Networks“. In: *null* 14.1 (1. Jan. 2010), S. 60–68. DOI: 10.1109/titb.2009.2037617. pmid: 20007032.

[WMD16] Christoph Wegener u. a. *Informationssicherheits-Management: Leitfaden Für Praktiker Und Begleitbuch Zur CISM-Zertifizierung*. 1. Aufl. 2016. Xpert.Press. Berlin, Heidelberg: Springer Berlin Heidelberg, 1. Jan. 2016. -. ISBN: 978-3-662-49167-6.

[WSL06] Haodong Wang u. a. „Elliptic Curve Cryptography-Based Access Control in Sensor Networks“. In: *Int. J. Security and Networks* (Vol. 1 1. Jan. 2006).

[Xu+11] Fengyuan Xu u. a. „IMDGuard: Securing Implantable Medical Devices with the External Wearable Guardian“. In: *2011 Proceedings IEEE INFOCOM*. IEEE, 1. Jan. 2011, S. 1862–1870. ISBN: 978-1-4244-9919-9. DOI: 10.1109/INFCOM.2011.5934987.

[Yan+14a] Qing Yang u. a. „An On-Chip Security Guard Based on Zero-Power Authentication for Implantable Medical Devices“. In: *2014 IEEE 57th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 1. Jan. 2014, S. 531–534. ISBN: 978-1-4799-4132-2. DOI: 10.1109/MWSCAS.2014.6908469.

[Yan+14b] Qing Yang u. a. „An On-Chip Security Guard Based on Zero-Power Authentication for Implantable Medical Devices“. In: *Midwest Symposium on Circuits and Systems* (25. Sep. 2014), S. 531–534. DOI: 10.1109/mwscas.2014.6908469.

[Yao+10] Lin Yao u. a. „An ECG-Based Signal Key Establishment Protocol in Body Area Networks“. In: *2010 7th International Conference on Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing* (26. Okt. 2010), S. 233–238. DOI: 10.1109/uic-atc.2010.7.

[Yao+11] Lin Yao u. a. „A Biometric Key Establishment Protocol for Body Area Networks“. In: *International Journal of Distributed Sensor Networks* 7.1 (1. Jan. 2011), S. 282986. ISSN: 1550-1477. DOI: 10.1155/2011/282986. pmid: null.

[Zag+15] Emna Kalai Zaghouani u. a. „ELPA: A New Key Agreement Scheme Based on Linear Prediction of ECG Features for WBAN“. In: *2015 23rd European Signal Processing Conference (EUSIPCO)*. 2015 23rd European Signal Processing Conference (EUSIPCO). Aug. 2015, S. 81–85. DOI: 10.1109/EUSIPCO.2015.7362349.

[Zha+12] Zhaoyang Zhang u. a. „ECG-Cryptography and Authentication in Body Area Networks“. In: *null* 16.6 (1. Nov. 2012), S. 1070–1078. DOI: 10.1109/titb.2012.2206115. pmid: 22752143.

[Zha+21a] Junqing Zhang u. a. „H2K: A Heartbeat-Based Key Generation Framework for ECG and PPG Signals“. In: *IEEE Transactions on Mobile Computing* 1 (1. Jan. 2021), S. 1–1. DOI: 10.1109/tmc.2021.3096384. pmid: null.

[Zha+21b] Guangrong Zhao u. a. „LeaD: Learn to Decode Vibration-based Communication for Intelligent Internet of Things“. In: *ACM Transactions on Sensor Networks* 17.3 (1. Jan. 2021), S. 1–25. ISSN: 1550-4859. DOI: 10.1145/3440250.

[Zhe+14a] Guanglou Zheng u. a. „An ECG-based Secret Data Sharing Scheme Supporting Emergency Treatment of Implantable Medical Devices“. In: *2014 International Symposium on Wireless Personal Multimedia Communications (WPMC)*. IEEE, 1. Jan. 2014, S. 624–628. ISBN: 978-986-03-3407-4. DOI: 10.1109/WPMC.2014.7014892.

[Zhe+14b] Guanglou Zheng u. a. „Securing Wireless Medical Implants Using an ECG-based Secret Data Sharing Scheme“. In: *International Symposium on Communications and Information Technologies* (1. Sep. 2014), S. 373–377. DOI: 10.1109/iscit.2014.7011935.

[Zhe+15] Guanglou Zheng u. a. „Encryption for Implantable Medical Devices Using Modified One-Time Pads“. In: *IEEE Access* 3 (2015), S. 825–836. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2015.2445336. URL: http://ieeexplore.ieee.org/document/7123574/ (besucht am 26. 02. 2023).

[Zhe+19] Guanglou Zheng u. a. „Finger-to-Heart (F2H): Authentication for Wireless Implantable Medical Devices“. In: *IEEE Journal of Biomedical and Health Informatics* 23.4 (1. Juli 2019), S. 1546–1557. DOI: 10.1109/jbhi.2018.2864796. pmid: 30106744.

[Zhu+22] Ying Zhu u. a. „Research on the Non-Contact Physiological Parameter Measurement Technology Based on Imaging Photoplethysmography“. In: *2022 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)* (2022). DOI: 10.1109/cisp-bmei56279.2022.9980189.

[ZRJ13] Meng Zhang u. a. „MedMon: Securing Medical Devices through Wireless Monitoring and Anomaly Detection“. In: *IEEE transactions on biomedical circuits and systems* 7.6 (1. Jan. 2013), S. 871–881. DOI: 10.1109/TBCAS.2013.2245664. pmid: 24473551.

[ZRJ14] Meng Zhang u. a. „Trustworthiness of Medical Devices and Body Area Networks“. In: *Proceedings of the IEEE* 102.8 (Aug. 2014), S. 1174–1188. ISSN: 1558-2256. DOI: 10.1109/JPROC.2014.2322103.