Sicherheitsanalyse von Bluetooth Low Energy Geräten in der Heimautomatisierung

Conference Paper · May 2019

CITATIONS
0

READS
117

3 authors, including:

Michael Rademacher
Hochschule Bonn-Rhein-Sieg
21 PUBLICATIONS **66** CITATIONS
SEE PROFILE

Karl Jonas
Hochschule Bonn-Rhein-Sieg
74 PUBLICATIONS **196** CITATIONS
SEE PROFILE

Sicherheitsanalyse von Bluetooth Low Energy Geräten in der Heimautomatisierung

Kevin Fröhlich, Michael Rademacher, Karl Jonas
Hochschule Bonn-Rhein-Sieg, Sankt Augustin, Deutschland
kevin.froehlich@h-brs.de, michael.rademacher@h-brs.de, karl.jonas@h-brs.de

Kurzfassung

Verschiedene intelligente Heimautomatisierungsgeräte wie Lampen, Schlösser und Thermostate verbreiten sich rasant im privaten Umfeld. Ein typisches Kommunikationsprotokoll für diese Geräteklasse ist Bluetooth Low Energy (BLE). In dieser Arbeit wird eine strukturierte Sicherheitsanalyse für BLE vorgestellt. Die beschriebene Vorgehensweise kategorisiert bekannte Angriffsvektoren und beschreibt einen möglichen Aufbau für eine Analyse. Im Zuge dieser Arbeit wurden einige sicherheitsrelevante Probleme aufgedeckt, die es Angreifern ermöglichen die Geräte vollständig zu übernehmen. Es zeigte sich, dass im Standard vorgesehene Sicherheitsfunktionen wie Verschlüsselung und Integritätsprüfungen häufig gar nicht oder fehlerhaft implementiert sind.

1 Einleitung

Das Internet der Dinge verbreitet sich im privaten Umfeld in Form von Heimautomatisierung immer rasanter [1]. Viele Geräte werden dem Endkunden unter der Prämisse Vereinfachung und Komfort angeboten. Typische Anwendungen sind die Steuerung und Automatisierung von Lampen, Türschlössern oder Steckdosen. Ein weit verbreitetes Funkprotokoll für Heimautomatisierungsgeräte ist BLE.

Obgleich des immensen Potentials dieser Technologien ist es in den vergangenen Jahren vermehrt zu Angriffen gekommen, bei denen signifikante Sicherheitslücken bewusst ausgenutzt wurden. Zu nennen sind beispielsweise der Angriff mit einem Wurm, der Glühbirnen befällt oder Angriffe auf smarte Schlösser, durch die Unberechtigte physikalischen Zugang zu geschützten Gegenständen, Räumen oder ganzen Gebäuden erhalten konnten [2, 3].

Es existiert der Bedarf für eine strukturierte Vorgehensweise, um die Sicherheitsfunktionen von Heimautomatisierungsgeräten bewerten zu können. Diese Vorgehensweise ermöglicht es Sicherheitslücken oder -probleme systematisch zu erkennen bzw. spezielle Implementierungen zu untersuchen. Diese Arbeit stellt einen ersten Entwurf für eine solche Vorgehensweise am Beispiel von BLE vor.

Vorherige Arbeiten befassen sich mit Angriffen und beschreiben erste Analysetechniken. Angriffsmöglichkeiten auf Heimautomatisierungsprotokolle stellt [4] zusammen. Speziell für BLE gibt [5] eine Übersicht über mögliche Angriffe und stellt ein Framework zum Testen von Geräten vor. [6] beschreibt verschiedene Phasen zur Analyse smarter Armbänder. Zunächst wird die Kommunikation aufgenommen, dann analysiert und anschließend verifiziert. Ein ähnlicher Ansatz findet sich in [7]. In [8] werden Bluetooth-Funktionalitäten, Android Apps und die Netzwerkkommunikation der Geräte ebenfalls mit dem Fokus auf smarte Armbänder untersucht. Aus der Arbeit von [9] geht ein Profiler hervor, der das niedrigste Sicherheitslevel eines Bluetooth-Gerätes identifiziert. In [10] werden smarte Schlösser auf die zwei Angriffskategorien Zustandskonsistenz und unerwünschtes Öffnen untersucht.

Im Vergleich zu vorherigen Arbeiten werden in der in dieser Arbeit vorgestellten Vorgehensweise nicht nur einzelne Teilbereiche von Angriffsvektoren betrachtet. Vielmehr können viele unterschiedliche Kategorien von Angriffen strukturiert nach den jeweiligen Schutzzielen mit einer durchgängigen Analyse untersucht werden.

Im folgenden Kapitel wird BLE zusammengefasst. In Kapitel 3 wird die erarbeitete Vorgehensweise für die Sicherheitsanalyse beschrieben. Drei beispielhafte Heimautomatisierungsgeräte werden in Kapitel 4 detailliert analysiert. Im Kapitel 5 wird ein Fazit gezogen und die Ergebnisse für weitere untersuchte Geräte aufgelistet.

2 Bluetooth Low Energy

Eines der meist genutzten Funkprotokolle in der Heimautomatisierung ist BLE. Dieses erfüllt viele der Anforderungen, die in diesem Kontext wichtig sind, wie Zuverlässigkeit, Sicherheit und Energieeffizienz [11]. In [12] werden die Konzepte von BLE erklärt.

BLE folgt einem Client/Server-Konzept. Die Kommunikation findet direkt zwischen der App auf einem Bluetooth-fähigen Endgerät und dem smarten Heimautomatisierungsgerät statt. Die Verbindung ist flüchtig, sie wird nur kurzzeitig aufgebaut, wenn der Client auf Daten zugreifen möchte. Der Client kann auf diese über *Handles* lesend oder schreibend zugreifen, sofern er die nötigen Rechte besitzt. Damit ein BLE-Gerät gefunden werden kann, sendet es in bestimmten Intervallen Advertising-Pakete aus.

BLE besitzt verschiedene Sicherheitsfunktionen um bestimmte Schutzziele zu erreichen. Diese werden im Bluetooth Standard spezifiziert [13]. In [12] werden diese anschaulich aufgearbeitet. Um Vertraulichkeit zu erreichen, sieht der Standard eine Verschlüsselung mit AES-CCM und einer Schlüssellänge von 128 Bit vor. Wird diese

eingesetzt, wird außerdem ein Message Integrity Check (MIC) an die verschlüsselten Nutzdaten angehängt um eine Manipulationen der Daten erkennen zu können. Ebenfalls existiert eine CRC-Prüfsumme, die jedoch nicht vor aktiver Manipulation schützt, sondern vor Übertragungsfehlern.
Für den Schlüsselaustausch zu Beginn der Kommunikation gibt es verschiedene Pairing-Verfahren. Die ausgehandelten Schlüssel können gespeichert und bei zukünftigen Verbindungen direkt verwendet werden (Bonding). Außerdem existiert die Möglichkeit die MAC-Adresse eines Gerätes periodisch ändern zu lassen, sodass ein Gerät nur mit einem gültigen, zuvor ausgehandelten Schlüssel (Identity Resolving Key, IRK) verfolgt werden kann.

3 Sicherheitsanalyse

Im Folgenden wird eine Vorgehensweise vorgestellt, um Geräte der Heimautomatisierung strukturiert und systematisch auf Schwachstellen untersuchen zu können. Ziel der Vorgehensweise ist es, die Funktionalitäten eines Geräts unbefugt zu übernehmen, also z.B. Schlösser zu öffnen oder Lampen an- und auszuschalten. Die Struktur orientiert sich an Schutzzielen, deren Erreichen im Bereich der Heimautomatisierung sinnvoll erscheint. Sie ist in **Abbildung 1** dargestellt und in sechs Hauptkategorien unterteilt. Die Mehrheit der Unterpunkte ist aus bekannten Angriffen zusammengestellt und verallgemeinert. Aus Gründen der Übersichtlichkeit sind diese stark verkürzt formuliert.
Neben den bekannten Schutzzielen Vertraulichkeit, Integrität und Verfügbarkeit sind in diesem Kontext die Schutzziele Authentizität und Nicht-Vermehrbarkeit von Bedeutung, wie im Folgenden beschrieben wird.

I Zu Beginn der Analyse werden zunächst **allgemeine Informationen** über das Gerät oder die gesamte Infrastruktur gesammelt.

II **Vertraulichkeit** bedeutet, dass Informationen nur Befugten zugänglich sein sollen. Erreicht werden kann dies z.B. durch den Einsatz von Verschlüsselung [14].

III Mit **Integrität** soll erreicht werden, dass Informationen vollständig und korrekt sind und nicht unbemerkt manipuliert werden können. Dies kann z.B. mit Hashfunktionen erreicht werden [14].

IV Die **Verfügbarkeit** bezeichnet die ständige Zugriffsmöglichkeit auf Informationen durch autorisierte Nutzer. Erreichbar ist dieses Schutzziel z.B. durch den Einsatz von Redundanzen [14].

V Mit **Authentizität** ist gemeint, dass Daten von einem bestimmten Absender stammen und die angegebene Identität dieses Absenders korrekt ist. Dieses Ziel kann mit dem Besitz bestimmter Informationen (z.B. Passwörter, Schlüssel) erreicht werden [14].

VI Mit **Nicht-Vermehrbarkeit** wird die Eigenschaft beschrieben, dass Informationen von Angreifern nicht kopiert und später wieder eingespielt werden können (Replay-Angriff). Das Schutzziel kann z.B. durch den Einsatz von Nonces (einmalig gültige Zeichenketten) erreicht werden [14].

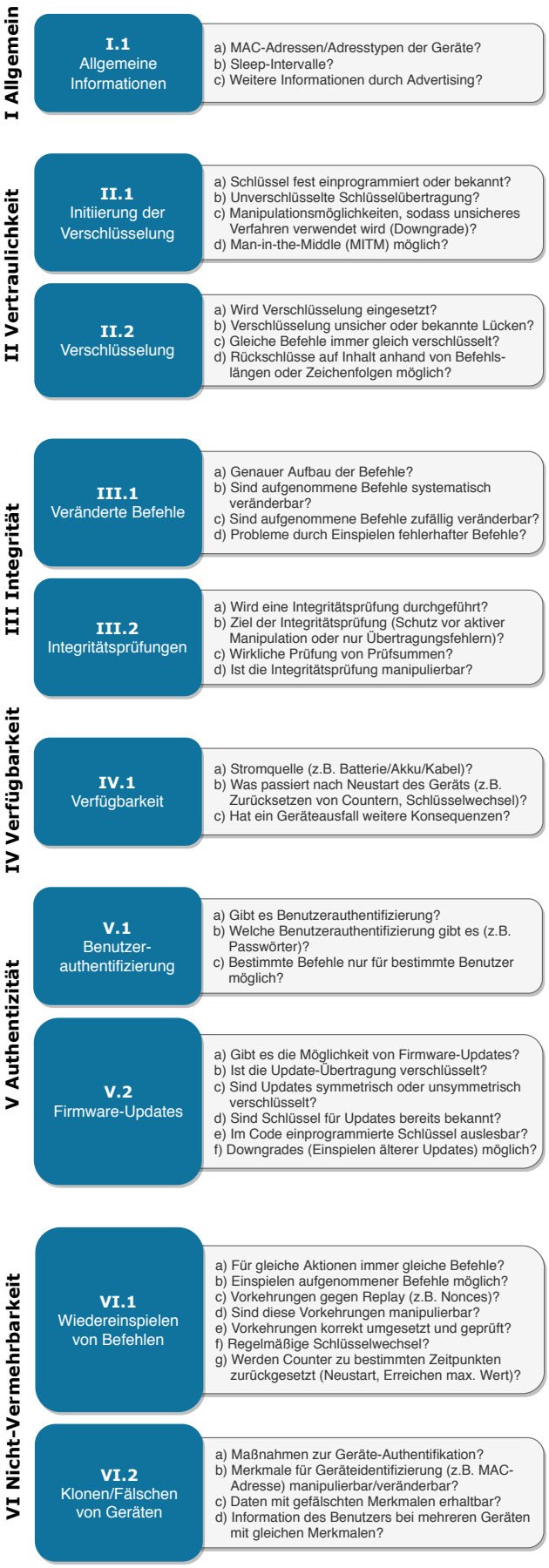

Abbildung 1 Vorgehensweise zur Sicherheitsanalyse

Um sämtliche BLE-Kommunikation mitlesen zu können, wird ein Texas Instruments CC2540 USB-Dongle mit entsprechender Software des Herstellers verwendet. Um eigene Befehle zu versenden, wird Python (2.7.15) und das Framework *bluepy* (1.2.0) verwendet.

4 Ergebnisse

Insgesamt wurden sieben Geräte analysiert. Diese wurden zufällig aus einem breiten Anwendungsspektrum im Bereich der Heimautomatisierung ausgewählt. Drei Sicherheitsanalysen unter Verwendung der vorgeschlagenen Vorgehensweise werden im Folgenden detaillierter vorgestellt. Dabei handelt es sich um Geräte, die mit BLE und einer entsprechenden App des Herstellers kommunizieren: Eine Glühbirne, ein Vorhängeschloss und ein Heizkörperthermostat.

4.1 Glühbirne

Abbildung 2 Lampe „Magic Blue Bulb“ mit App

Bei der „Magic Blue Bulb“ handelt es sich um eine smarte LED-Glühbirne, die via BLE über eine App gesteuert werden kann (vgl. **Abbildung 2**).
Die Glühbirne ist unmittelbar nach der Installation der App mit dieser steuerbar. Es wird lediglich in der App eine Suche gestartet, die alle Glühbirnen dieser Art in Empfangsreichweite auflistet. Die gewünschte Glühbirne kann anschließend vom Benutzer ausgewählt und gesteuert werden. Die Glühbirne sendet automatisch in einem Intervall von 40 ms bis 50 ms Advertising-Pakete aus, die Informationen über das Gerät enthalten, worin auch die MAC-Adresse angegeben ist. Der Benutzer der App benötigt keinen PIN oder ein Passwort um sich mit der Glühbirne zu verbinden. **(I)**
BLE bietet im Standard die Möglichkeit Verschlüsselung einzusetzen, nachdem ein Pairing-Prozess vorausgegangen ist. In der „Magic Blue Bulb“ wird darauf jedoch verzichtet. Entsprechend ist die Kommunikation nicht vor Mitlesen geschützt und es sind Rückschlüsse auf die Befehle und Inhalte der Nachrichten möglich. **(II)**
Für das Ein- bzw. das Ausschalten der Glühbirne gibt es jeweils einen statischen Schreibbefehl. Dieser besteht aus drei Bytes. Beim Ändern der Farbe und/oder Helligkeit werden immer ähnlich lange und gleich aussehende Schreibbefehle abgesendet. Diese sind ebenfalls fast vollständig statisch, bis auf genau drei Byte, die nach Einstellen bestimmter Farben in der App darauf schließen lassen, dass es sich dabei um einen RGB-Farbwert handelt, der je nach Benutzereingabe des gewünschten Farbtons/Helligkeit variiert.
Da es keine Mechanismen gibt, um die Befehle zu schützen, können diese nachgestellt und gesendet werden. Eine Integritätsprüfung existiert ebenfalls nicht. Alle Befehle bestehen zwar aus weiteren Daten, die zu dem Befehl an sich (z.B. Farbwert einstellen) unterschiedlich sind, jedoch sind diese nur für die jeweilige Art von Befehl unterschiedlich, nicht für jeden einzelnen Befehl.
Die selbst erstellten Befehle (ohne App) wurden von der Glühbirne verarbeitet und ausgeführt. Es ist möglich, zuvor aufgenommene Nachrichten wieder einzuspielen. Die Glühbirne reagiert entsprechend auf diese Befehle, als wären sie von einem legitimen Absender gesendet worden. Auch ohne die zugehörige App zu verwenden, ist die Glühbirne vollständig durch jede Person in Bluetooth-Reichweite steuerbar. **(III)**
Nach dem Neustart des Geräts bzw. nach erneutem Einschalten der Stromversorgung wird die zuletzt eingestellte Kombination aus Farbe und Helligkeit eingestellt. Weitere Konsequenzen gibt es nicht. **(IV)**
Der Benutzer wird nicht nach Benutzerdaten wie Passwörtern gefragt, was bedeutet, dass die Glühbirne keine Abhängigkeiten zu Benutzern in den Befehlen beinhaltet und die Befehle für alle Absender gleich sind. Authentizität wird nicht überprüft, jeder Befehl jedes Absenders wird angenommen und ausgeführt. Ein Zurücksetzen des Geräts auf den Werkszustand ist nicht möglich.
In der App konnte keine Möglichkeit gefunden werden die Firmware der Glühbirne zu aktualisieren. Es existiert damit keine Möglichkeit die nicht vorhandenen Sicherheitsmechanismen nachträglich zu korrigieren. **(V)**
Aus jeder Aktion in der App resultiert der jeweils gleiche BLE-Befehl. Es ist ohne Probleme möglich zuvor aufgenommene Nachrichten wieder einzuspielen. Die Lampe reagiert entsprechend auf diese Befehle, als wären sie von einem legitimen Absender gesendet worden. Schutzmechanismen sind nicht implementiert. **(VI)**
Zusammenfassend sind die im BLE-Standard vorgesehenen Sicherheitsfunktionen wie Verschlüsselung oder Integritätsprüfung in der „Magic Blue Bulb“ nicht umgesetzt. Es existieren keine Maßnahmen, um einen Angreifer davon abzuhalten, die Glühbirne vollständig zu steuern.

4.2 Vorhängeschloss

Das in **Abbildung 3** dargestellte „Tapplock One“ ist ein smartes Vorhängeschloss. Optisch wirkt das Schloss auf den ersten Blick wie ein handelsübliches Vorhängeschloss, es besitzt jedoch keine Öffnung für einen Schlüssel. In der Mitte des Schlosses befindet sich ein Fingerabdrucksensor, darüber eine Kontrollleuchte und an der Unterseite ein kleiner Knopf sowie ein Anschluss für ein Ladekabel.
Der Nutzer des Schlosses hat drei Möglichkeiten das

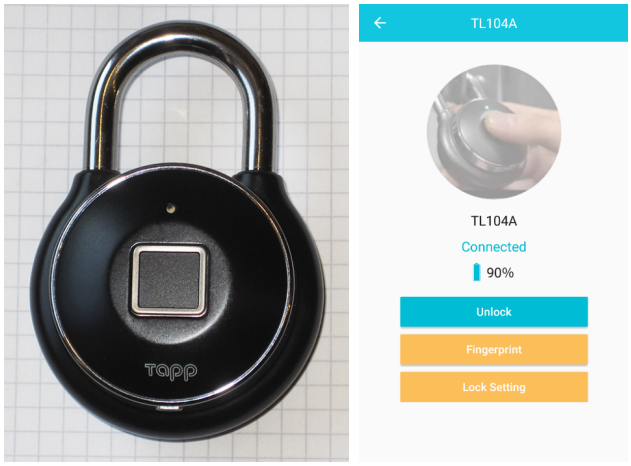

Abbildung 3 Vorhängeschloss „Tapplock One“ mit App

Schloss zu Öffnen: App, Fingerabdrucksensor und Morse-Code mittels Knopf. In der zugehörigen App „Tapplock“ existiert die Möglichkeit sich mit dem Schloss per BLE zu verbinden. Dort kann der Benutzer alle „Tapplock“-Schlösser in seiner Reichweite auflisten lassen und auswählen. Bei Auswahl eines Schlosses wird die Verbindung hergestellt, der Benutzer bekommt den aktuellen Ladezustand des Akkus angezeigt und kann das Schloss öffnen. Dafür ist lediglich ein Klick auf den entsprechenden Button in der App notwendig.

Durch einen Blog-Artikel [15] ist bereits im Vorfeld bekannt, dass das Schloss sicherheitskritische Probleme aufweist und sich durch Angreifer mit Kenntnis der MAC-Adresse öffnen lässt.

Das Schloss sendet keine Advertising-Pakete dauerhaft aus, sondern erst nach zweimaligem Betätigen des Knopfes an der Unterseite des Schlosses. Dies geschieht im Intervall von etwa 40 ms bis 50 ms. Dieser Modus hält für 20 s an, danach geht das Schloss in einen Sleep-Modus, um Energie zu sparen. Befindet sich das Schloss im Sleep-Modus, werden keine Befehle von diesem angenommen. **(I)**

Beim „Tapplock One“ wird nach dem Verbinden mit dem Schloss kein Schlüsselaustausch initiiert und auch keine Verschlüsselung nach dem BLE-Standard eingesetzt. Die Befehle werden im Klartext an das Schloss gesendet, was ein Mitlesen möglich macht. Es ist möglich Rückschlüsse auf die Inhalte der Befehle zu ziehen, da diese unterschiedliche Längen aufweisen und mit verschiedenen Zeichenfolgen beginnen. **(II)**

Eine Integritätsprüfung nach BLE existiert nicht. **(III)**

Ein statischer Befehl initiiert die Verbindung mit dem Schloss, erst danach ist das Absenden weiterer Befehle möglich. Durch das Dekompilieren der Android-App ist es möglich eine Übersicht über alle möglichen Befehle zu erhalten. Der Befehl „Regular Pair“ besteht aus einem festen Wert, gefolgt von Bestandteilen eines MD5-Hashwertes der MAC-Adresse des Schlosses, gefolgt von einer zwei Byte langen Prüfsumme. Die MAC-Adresse kann aus den Advertising-Paketen des Schlosses gewonnen werden, während die Prüfsummen-Berechnung aus dem dekompilierten App-Code entnommen werden kann. Somit kann dieser Befehl nachgestellt werden. Der Befehl zum Öffnen kann ebenfalls nachgestellt werden, da er lediglich ein fester Bytewert ist. Sofern der Befehl „Regular Pair“ zuvor gesendet wurde, ist es möglich den Befehl zum Öffnen an das Schloss zu senden. **(IV)**

Es existiert die Möglichkeit von Firmware-Updates, wodurch die Sicherheitslücke nach Herstellerangaben inzwischen behoben sein soll [16]. Allerdings werden Updates nicht automatisch eingespielt und Schlösser im Werkszustand sind weiterhin anfällig für diese Sicherheitsproblematik. **(V)**

Durch die statischen Befehle müssen diese nicht zwingend selbst berechnet werden, sondern können auch aufgenommen und erneut eingespielt werden. **(VI)**

Zusammenfassend ist es durch die fehlende Verschlüsselung und statischen Befehle möglich, gesendete Befehle aufzunehmen und wiedereinzuspielen, um das Schloss zu öffnen. Mit Kenntnis über den Aufbau der Befehle durch Dekompilieren der App ist es zudem möglich, jedes beliebige „Tapplock One“ zu öffnen, zumindest mit einer anfälligen Firmware-Version. Die Prüfsumme am Ende der Befehle schützt nicht vor Manipulationen der Nachrichten, da diese ebenfalls manipuliert und selbst berechnet werden kann. Lediglich die MAC-Adresse des Schlosses muss bekannt sein. Diese ist durch die Advertising-Pakete des Schlosses ermittelbar.

4.3 Heizkörperthermostat

Abbildung 4 „eQ-3 Heizkörperthermostat“ mit App

Bei dem „eQ-3 BLUETOOTH Smart Heizkörperthermostat“ handelt es sich um ein Gerät zur Steuerung eines Heizkörpers (vgl. **Abbildung 4**). Das Gerät wird anstelle des Thermostats an der Heizung installiert und die gewünschte Temperatur kann anschließend eingestellt werden. Möglich ist dabei eine Bedienung des Geräts entweder über die App, die via BLE mit dem Thermostat kommuniziert, oder manuell am Thermostat selbst.

Bei der ersten Benutzung der App wird das Thermostat durch Betätigen des Stellrades für einige Sekunden in einen Pairing-Modus versetzt, wodurch dem Benutzer auf dem Display eine Nummer bestehend aus vier Ziffern angezeigt wird, die er in der App eingeben muss. Danach sind Thermostat und App gekoppelt und der Benutzer ist in der

Lage, über einen Regler in der App die gewünschte Temperatur einzustellen und weitere Einstellungen vorzunehmen. Nachdem sich das Heizkörperthermostat im Normalzustand befindet, sendet es automatisch im Abstand von 1 s Advertising-Pakete aus. **(I)**

Das Mitschneiden der Kommunikation zwischen App und Thermostat ergibt, dass beide miteinander verschlüsselt kommunizieren. Dies ist daran zu erkennen, dass nach dem Verbinden der Geräte von der App der Befehl „Encryption Request“ gesendet wird, gefolgt von einem „Encryption Response“ des Thermostats sowie ein Befehl zum Starten der Verschlüsselung, wie es im Standard vorgesehen ist.

Die Initiierung der Verschlüsselung wurde bereits im Groben in der Beschreibung und Einrichtung des Geräts erläutert. Demnach beinhaltet das Thermostat zwar keine Möglichkeit zur Eingabe, sehr wohl aber über ein Display zur Ausgabe eines Nummernwertes. Das mobile Endgerät mit der App verfügt über die Möglichkeit sowohl von Eingaben als auch Ausgaben. Somit ist es laut BLE-Standard möglich — und diese Vorgehensweise findet auch statt —, dass auf dem Thermostat ein zufälliger Wert generiert wird, den der Benutzer nach dem Ablesen auf dem Display des Thermostats in seiner App eingibt. Mit diesem Wert kann das Bluetooth Pairing-Verfahren „Passkey Entry“ durchgeführt werden. Durch dieses Verfahren wird dafür gesorgt, dass dieser zur Generierung des Schlüssels verwendete Wert nicht per drahtloser Übertragung ausgetauscht wird und somit nicht mitgeschnitten werden kann.

Jedoch ist ein Downgrade bei der Initiierung der Verschlüsselung möglich. Das Thermostat akzeptiert ebenfalls unverschlüsselte Verbindungen, auch nachdem es zuvor bereits einen Schlüsselaustausch gab. **(II)**

Alle verschlüsselten Daten enthalten zum Schutz vor Manipulation der verschlüsselten Nutzdaten zusätzlich einen MIC, wie es im BLE-Standard vorgesehen ist.

Durch den Einsatz der Verschlüsselung ist es nicht möglich den Aufbau und die Struktur der Nachrichten zu erfahren, die die App zur Steuerung an das Thermostat sendet. Mithilfe der dekompilierten App können jedoch weitere Informationen über die Befehle gewonnen werden. Der Befehl zum Einstellen der Temperatur besteht lediglich aus einem festen Byte-Wert, gefolgt vom gewünschten Temperaturwert (ebenfalls 1 Byte). Es handelt sich also um statische Befehle, die nur durch den Einsatz von Verschlüsselung bei jedem Absenden unterschiedlich aussehen.

Da das Thermostat von sich aus keine Anfrage zur Verschlüsselung aussendet, sondern nur die App, wird eine eigene, unverschlüsselte Verbindung mit dem Gerät aufgebaut. Die statischen Befehle zum Einstellen der Temperatur werden gesendet und ausgeführt. Somit ist das Gerät vollständig steuerbar. In diesem Fall war es möglich die eingestellte Temperatur auf beliebige Werte zu setzen. Es können außerdem am Thermostat und in der App nicht einstellbare Werte (z.B. 80,5 °C) an das Thermostat gesendet werden. Diese werden angenommen und ausgeführt. Somit war die vollständige Kontrolle des Thermostats möglich, ohne im Besitz des gültigen Schlüssels zu sein oder ein Pairing-Verfahren durchgeführt zu haben. **(III)**

Bei einem Neustart sind bis auf den Schlüssel alle Eigenschaften wie die eingestellte Temperatur zurückgesetzt und müssen neu eingegeben bzw. eingestellt werden, bis das Gerät wieder wie vor dem Neustart einsatzbereit ist. **(IV)**

In der App ist es möglich Firmware-Updates manuell durchzuführen. Somit besteht die Möglichkeit nachträglich Sicherheitslücken zu schließen. **(V)**

Die implementierten Sicherheitsfunktionen folgen zusammenfassend dem Standard von BLE. Allerdings ist es durch das Dekompilieren und Analysieren des Programmcodes der App möglich, Informationen über die genauen gesendeten Befehle zu erhalten. Da die Verschlüsselung durch die App initiiert wird und das Thermostat von sich aus keine verschlüsselte Verbindung erfordert, ist es möglich eine ungesicherte Verbindung aufzubauen und Befehle unverschlüsselt zu versenden, die vom Gerät als gültig angesehen und ausgeführt werden.

Somit ist das Gerät von beliebigen Angreifern vollständig steuerbar, obwohl die Sicherheitsfunktionen von BLE in der App umgesetzt wurden. Der Hersteller wurde über die oben genannten Erkenntnisse informiert.

5 Fazit

Aus verschiedenen Angriffen unterschiedlicher Funkprotokolle aus der Vergangenheit wurde eine verallgemeinerte Vorgehensweise erarbeitet, die eine strukturierte Sicherheitsanalyse von Geräten in der Heimautomatisierung ermöglicht.

Im Standard von BLE sind Maßnahmen vorgesehen, die eingesetzt werden sollten, um die Geräte bzw. ganze Infrastrukturen vor Angriffen zu schützen. Der Einsatz dieser Sicherheitsfunktionen ist jedoch häufig den Herstellern überlassen. Von sieben analysierten Geräten hatte lediglich eines die Sicherheitsfunktionen so implementiert wie sie im BLE-Standard vorgesehen sind (Pairing-Prozess, Verschlüsselung, Integritätsprüfung). Zwei weitere Geräte haben Sicherheitsmechanismen implementiert, verwenden BLE jedoch nur als Transportebene für ihre Nachrichten.

Von den sieben analysierten Geräten gab es nur zwei Geräte, die so gegen Angriffe auf das Funkprotokoll abgesichert waren, dass es mit der erarbeiteten Vorgehensweise nicht möglich war, die Geräte vollständig unter Kontrolle zu bringen und zu steuern, sofern der verwendete Schlüssel nicht bekannt ist. Alle anderen fünf Geräte — drei von ihnen wurden genauer vorgestellt — konnten beliebig gesteuert werden. Dies war entweder aufgrund von Implementierungsfehlern möglich, in den meisten Fällen jedoch wegen fehlender Verschlüsselung, unsicherem Schlüsselaustausch oder Verwendung von statischen Befehlen. **Tabelle 1** fasst alle Ergebnisse zusammen. Zur Übersichtlichkeit sind die allgemeinen Informationen in der Tabelle nicht aufgeführt. Die Update-Funktionalitäten aus IV.2 wurden — sofern diese existierten — nicht untersucht, da entweder keine Firmware-Updates zur Verfügung standen oder die Geräte in ihrer Firmware zu Schulungszwecken nicht verändert werden sollten. Aus diesem Grund sind die Punkte IV.2 b)–f) in der Tabelle nicht aufgeführt. Kritische Punkte sind orange, sehr kritische Punkte, die zur vollständigen Kontrolle durch Angreifer führen, rot markiert.

Insgesamt lässt sich zum einen zusammenfassen, dass es

	II								III								IV			V				VI										X
	1				2				1			2					1			1			2	1							2				
	a)	b)	c)	d)	a)	b)	c)	d)	b)	c)	d)	a)	b)	c)	d)	a)	b)	c)	a)	b)	c)	a)	a)	b)	c)	d)	e)	f)	g)	a)	b)	c)	d)	
1	NR	NR	NR	NR	N	NR	NR	J	J	NR	N	N	NR	NR	NR	K	LZ	N	N	NR	NR	N	J	J	N	NR	NR	NR	NR	N	NR	N	NB	J
2	NR	NR	NR	NR	N	NR	NR	J	J	NR	N	J	M	J	J	A	LZ	N	N	NR	NR	J	J	J	N	NR	NR	NR	NR	N	NR	N	NB	J
3	N	N	N	N	J	N	N	J	N	N	N	J	M	J	N	B	LZ	N	J	ID, S	N	J	N	N	J	N	J	NB	NB	J	N	N	NB	N
4	NR	NR	NR	NR	N	NR	NR	J	J	NR	J	N	NR	NR	NR	B	LZ	N	N	NR	NR	N	N	J	N	NR	NR	NR	NR	N	NR	N	NB	J
5	N	N	J	N	J	N	N	N	N	N	N	J	M	J	N	B	SZ	N	J	S	N	J	N	N	J	N	J	N	NB	J	N	N	NB	J
6	J	NR	NR	NR	J	N	J	J	J	NR	N	J	M	J	J	A	LZ	N	J	S	N	N	J	N	J	N	J	N	J	J	N	N	NB	J
7	N	N	N	N	J	N	N	N	N	N	N	NB	NB	NB	NB	A	NB	NB	J	S	N	NB	N	N	NB	NB	NB	NB	NB	J	N	N	NB	N

J: Ja, N: Nein, NB: Nicht betrachtet, NR: Nicht relevant, K: Kabel, B: Batterie, A: Akku, LZ: Letzter Zustand, SZ: Standardzustand, ID: Benutzer-ID, S: Schüssel, Spalte X: Vollständig übernehmbar

Tabelle 1 Ergebnisse der analysierten Geräte (1: Magic Blue Bulb, 2: Tapplock One, 3: Equiva Türschlossantrieb, 4: Basetech Raumthermostat, 5: eQ-3 Heizkörperthermostat, 6: iHealth Blutdruckmessgerät, 7: ABUS Fahrradschloss)

mit der erarbeiteten Vorgehensweise möglich war, strukturiert vorzugehen und Sicherheitsprobleme bei der Kommunikation zu erkennen. Zum anderen lässt sich feststellen, dass es insbesondere bei den Implementierungen der Sicherheitsmechanismen noch Verbesserungsbedarf aufseiten der Hersteller von Geräten im Bereich der Heimautomatisierung gibt. Entweder existieren kaum Mechanismen die eine Übernahme des Geräts durch Angreifer verhindern oder diese sind nur unzureichend umgesetzt.
Ein Übertrag dieses Verfahrens für BLE auf weitere Funkprotokolle der Heimautomatisierung ist angedacht, womit es dann möglich wäre beliebige Smart Home Geräte auf ihre Sicherheit zu untersuchen.

6 Literatur

[1] Bitkom: Home Smart Home: Jeder Vierte ist auf dem Weg zum intelligenten Zuhause, 2018
https://www.bitkom.org/Presse/Presseinformation/Home-Smart-Home-Jeder-Vierte-ist-auf-dem-Weg-zum-intelligenten-Zuhause.html
(Letzter Zugriff 11.03.2019)

[2] Scherschel F.: Licht an, Licht aus: ZigBee-Wurm befät smarte Glühbirnen. Heise Newsticker Artikel vom 08.11.2016

[3] Rose A., Ramsey B.: Picking Bluetooth Low Energy Locks from a Quarter Mile Away. Merculite Security, Folien DEFCON Vortrag, 2016

[4] Celebucki D., Lin M., Graham S.: A security evaluation of popular Internet of Things protocols for manufacturers. IEEE Int. Conf. on Consumer Electron. (ICCE), S.1–6, 2018

[5] Ray A., Raj V., Oriol M., Monot A., Obermeier S.: Bluetooth Low Energy Devices Security Testing Framework. IEEE 11th Int. Conf. on Software Testing, Verification and Validation (ICST), S.384–393, 2018

[6] Zhang Q., Liang Z.: Security analysis of bluetooth low energy based smart wristbands. 2nd Int. Conf. on Frontiers of Sensors Tech. (ICFST), S.421–425, 2017

[7] Langone, M., Setola R., Lopez J.: Cybersecurity of Wearable Devices: An Experimental Analysis and a Vulnerability Assessment Method. IEEE 41st Annu. Computer Software and Applications Conf. (COMPSAC), S.304–309, 2017

[8] Goyal R., Dragoni N., Spognardi A.: Mind the tracker you wear: a security analysis of wearable health trackers. Proc. of the 31st Annu. ACM Symp. on Appl. Computing, S.131–136, 2016

[9] Sivakumaran P., Alis J.: A Low Energy Profile: Analysing Characteristic Security on BLE Peripherals. Proc. of the Eighth ACM Conf. on Data and Application Security and Privacy, S.152–154, 2018

[10] Ho G., Leung D., Mishra P., Hosseini A., Song D., Wagner D.: Smart Locks: Lessons for Securing Commodity Internet of Things Devices. Proc. of the 11th ACM on Asia Conf. on Computer and Commun. Security, S.461–472, 2016

[11] Bertko C., Weber T.: Home, Smart Home. Carl Hanser Verlag München, 2017

[12] Heydon, R.: Bluetooth Low Energy — The Developer's Handbook. Pearson Education, Inc., 2013

[13] Bluetooth Core Specification v5.1. Bluetooth SIG, 2019

[14] Bedner M., Ackermann T.: Schutzziele der IT-Sicherheit. Datenschutz und Datensicherheit (Volume 34, Nr. 5), S.323–328, 2010

[15] Tierney A.: Totally Pwning the Tapplock Smart Lock. PenTestPartners Penetration testing and security services, 2018
https://www.pentestpartners.com/security-blog/totally-pwning-the-tapplock-smart-lock/
(Letzter Zugriff 19.02.2019)

[16] Tapplock Corp.: Stay Protected — Notiz vom 12.06.2018
https://tapplock.com/notice/20180612/
(Letzter Zugriff 19.02.2019)