ZIGBEE

Wahlfachprojekt SS 2018

Daniel Tod, Luca Strobl, Dominik Mayer, Jean Castillo

Betreuung durch Silvia Schmidt, MSc BSc

Inhalt

Einleitung				
ZigBee Allgem Techni A I Z G T K	eine Informationen sche Informationen Allgemeines implementierung igBee-Profile Gerätetypen Topologie Communikation	7		
Angriff a Voraus H S	auf ZigBee setzungen lardware software	12		
Setup W R P K K Sniffi R K K	Vindows-PC Caspberry-Pi Philips Hue Cali Linux CillerBee Dependencies KillerBee Atmel RZ Raven USB-Stick Firmware-Sicherung Firmware-Flash Verifizierung Ong Caspberry Pi I Cali-PC I Cali-PC II			

Quellen

26

EINLEITUNG

Einleitung

Dieses Projekt gliedert sich in zwei Teile: in einen theoretischen und einen praktischen Teil.

Im theoretischen Teil geben wir einen allgemeinen Überblick über den Netzwerkstandard gefolgt von den Erläuterungen zur technischen Umsetzung des Protokolls.

Im praktischen Teil demonstrieren wir einen Angriff auf das ZigBee-Netzwerk, indem wir Befehle mitschneiden. Das Sniffing erfolgt mit spezieller Hardware und Software und wurde in einer Netzwerklaborähnlichen Umgebung durchgeführt. Die technischen Voraussetzungen sowie alle Schritte wurden genau dokumentiert und mit zahlreichen Grafiken hinterlegt, um dem Leser die Möglichkeit zu bieten, das Setup nachzubauen und selbst einen Angriff zu starten.

ZIGBEE

Allgemeine Informationen

ZigBee ist ein offener Niedrigenergie-Netzwerkstandard für geringe Datenaufkommen. Er findet unter anderem Anwendung in der Hausautomation, in Sensornetzwerken und in der Lichttechnik.

Der Standard wurde von der ZigBee-Allianz entwickelt, welche 2002 gegründet wurde und mittlerweile mehr als 400 Unternehmen umfasst.

Das Ziel von ZigBee ist, verschiedene Geräte miteinander zu verbinden. So soll zum Beispiel möglich sein, alle Glühbirnen eines Haushalts mit einer einzigen Fernbedienung zu steuern.

Aufgrund der Tatsache, dass es viele verschiedene Hersteller von ZigBee-fähigen Produkten gibt, stellt die Interoperabilität zwischen den unterschiedlichen Geräten ein zentrales Designkriterium dar. Und da viele der Produkte der Unterhaltungselektronik zuzuordnen sind, ist auch eine einfache Handhabung, wie das Paaren von Geräten, von großer Bedeutung. Dadurch ergeben sich zahlreiche Schwachstellen. [1][2][3]

Technische Informationen

Allgemeines

ZigBee durchdringt Wände und Einrichtungsgegenstände und wird nicht von Personen beeinflusst, die sich durch den Kommunikationskanal bewegen. Die Übertragung ist robust gegen Interferenz und zeichnet sich vor allem durch ultra low power consumption aus. Die Kommunikation erfolgt bi-direktional, reicht bis zu 100 Meter und erlaubt eine Übertragungsrate von 250 kbps.[1][2]

Implementierung

ZigBee erweitert die PHY-Schicht und MAC-Schicht des IEEE 802.15.4 Standards um eine eigene NWK-Schicht und APL-Schicht (Grafik 1). Die verwendete Frequenz ist abhängig von der jeweiligen Region: in Amerika erfolgt die Kommunikation im 915 MHz Bereich, in England liegt sie bei 868 MHz, in China bei 784 MHz und in Europa bei 2,4 GHz.[1][2][5][6]



Grafik 1: ZigBee

ZigBee-Profile

In ZigBee-Profilen werden Systemvoraussetzungen definiert. Um Kommunikation zwischen Geräten zu ermöglichen, müssen bestimmte Übereinkünfte getroffen werden. Das heißt, um den Anwendungen zu erlauben, Befehle zu senden, Daten abzufragen und Befehle und Anfragen zu verarbeiten, muss unter anderem die Übereinkunft von Nachrichten, das Nachrichtenformat und die Verarbeitung von Befehlen definiert werden.

Beispiele sind das ZigBee Light Link und das Home Automation Profil.[1][2][3]

Gerätetypen

Der ZigBee-Standard definiert drei verschiedene Gerätetypen: den ZigBee-Coordinator, den ZigBee-Router und das ZigBee-End Device. Der Coordinator startet das Netzwerk mit festgelegten Parametern und arbeitet danach als Gateway, das immer erreichbar ist. Zu den Parametern zählen unter anderem der Übertragungskanal und die verwendeten Verschlüsselungscodes.

Der Router erweitert den Sendebereich und ist ebenfalls immer erreichbar. Beim Routing unterscheidet man zwischen der Star-, Meshund Cluster Tree-Topologie.

End Devices melden sich bei einem Router an, um dem ZigBee-Netzwerk beizutreten. Sie verfügen über einen Standby-Modus, um die Lebenszeit zu erhöhen und zeichnen sich durch geringe Produktionskosten
aus.[2][4][5]

Topologie

In der Star-Topologie verbinden sich alle End Devices mit einem zentralen Coordinator. Es gibt keine Router.

In der Mesh-Topologie bauen Router eine Verbindung zum Coordinator auf, wobei ein Router nicht zwingend mit dem Coordinator verbunden sein muss. Es reicht, wenn eine Verbindung zu mindestens einem anderen Router besteht. End Devices stellen keine direkte Verbindung mit dem Coordinator her, sie binden sich lediglich an Router.

Die Cluster Tree-Topologie zeichnet sich dadurch aus, dass sich sowohl Router als auch End Devices mit dem Coordinator verbinden können. An die Router können sich weitere Router oder End Devices hängen. Grafik 2 verbildlicht die verschiedenen Topologie-Konzepte.[2][3]



Grafik 2: ZigBee-Netzwerk Topologien

Kommunikation

Der Coordinator startet ein Wireless Personal Area Network (WPAN) mit einer 64-Bit erweiterten PAN-ID zur Identifizierung. ZigBee-fähige Geräte können diesem Netzwerk nun beitreten, indem sie die verfügbaren Netze scannen und einen Pairing-Prozess anstoßen. Der Pairing-Prozess mit dem Coordinator unterscheidet sich zwischen den einzelnen Produkten. So sendet eine Philips Hue Glühbirne beispielsweise eine Anfrage an den stärksten Sender, um eine Verbindung aufzubauen. Dabei ist zu beachten, dass sich aufgrund der Sicherheit die Glühbirne innerhalb eines bestimmten Radius um den Coordinator befinden muss, um dem Netzwerk beitreten zu können. Eine andere Methode des Pairings ist die Push-Button-Funktion, die unter anderem bei Fernbedieungen zum Einsatz kommt.

Jedes ZigBee-Funkmodul besitzt eine eindeutige 64-Bit IEEE Adresse, wobei den Modulen bei Eintritt ins Netzwerk noch eine 16-Bit Short Address zugewiesen wird.[2][3][4]

Für die Kommunikation innerhalb des Netzwerks werden zwei verschiedene Schlüssel verwendet: der Network Key und der Link Key. Der Network Key wird für die Verschlüsselung des Broadcast Traffics benutzt und muss jedem Gerät im Netzwerk bekannt sein. Der Link Key hingegen dient zur Verschlüsselung von Unicast Traffic. Für jede einzelne bidirektionale Verbindung zwischen zwei Nodes ist ein Link Key notwendig, das heißt, wenn ein Sensor mit dem Coordinator und einem anderen Sensor verbunden ist, so verwendet er zwei verschiedene Link Keys, einen für den Kommunikationskanal mit dem Coordinator und einen für den Kanal mit dem anderen Sensor. Die Schlüssel haben eine Länge von 128 Bit und werden mittels AES-CCM* generiert.[2][3][4][5][6][7]

Die Verteilung des Network Keys auf die Geräte erfolgt durch den Transport über das Netzwerk oder durch manuelles Vorinstallieren des Schlüssels auf den einzelnen Geräten, das aber einen erheblichen zusätzlichen Aufwand bedeutet und dadurch die Usability stark einschränkt.

Beim Transport wird der Network Key vom Trust Center auf alle Geräte verteilt. Dabei wird der Schlüssel mit dem Default Trust Center Link Key verschlüsselt. Das Trust Center ist verantwortlich für das Sicherheitsmanagement in einem ZigBee-Netzwerk. Es kümmert sich um die Authentifizierung von Geräten, die dem Netzwerk beitreten möchten sowie um die Verteilung der Schlüssel.[2][3][5][6][7]

Die Generierung des Link Keys kann durch drei verschiedene Methoden erfolgen: er wird mithilfe des Master Keys erzeugt, der bei der Produktion auf dem Gerät vorinstalliert wird. Die zweite Möglichkeit ist das Hinzufügen des Link Keys durch den Endbenutzer über eine outof-band Methode. Zuletzt kann das Trust Center den Link Key an die einzelnen Nodes senden.[3][5][6][7]

ANGRIFF AUF ZIGBEE

Angriff auf ZigBee

Voraussetzungen

Hardware

In diesem Projekt wurde die folgende Hardware verwendet:

- Raspberry Pi 3 Model B+ [18] (Grafik 3)
- SD Karte mit mindestens 8 GB Speicher
- RaspBee: ZigBee Aufsatzmodul für Raspberry Pi [19] (Grafik 4)
- Philips Hue Glühbirne [29] (Grafik 5)
- Atmel RZ Raven USB-Stick [21] (Grafik 6)
- AVR Dragon Programmier-Board [22] (Grafik 7)
- IDC-Kabel (Grafik 8)
- USB 2.0 Kabel, A Stecker auf B Stecker (Grafik 9)
- Windows-PC
- USB-Stick mit mindestens 4GB Speicher
- USB Maus und Tastatur
- Externer Monitor
- o HDMI-Kabel





Grafik 3: Raspberry Pi 3 Model B+ [12] Grafik 4: RaspBee Modul [13]





Grafik 5: Philips Hue Glühbirne [14] Grafik 6: Atmel RZ Raven USB-Stick [15]



Grafik 7: AVR Dragon Programmier-Board [16]



Grafik 8: IDC-Kabel



Grafik 8: USB 2.0 Kabel, A Stecker auf B [17]

Software

In diesem Projekt wurde die folgende Software verwendet:

- RaspBee Gateway SD-Karten Image Raspbian Jessi RaspBee (Stable)
 Version 01-2017 [23]
 - Verwandelt den Raspberry Pi in ein ZigBee-Gateway
- SD Card Formatter 5.0 for SD/SDHC/SDXC [24]
- Win32 Disk Imager [25]
- KillerBee [26]
 - Framework und Tools f
 ür Angriffe auf ZigBee und IEEE 802.15.4 Netzwerke
- Kali Linux 64 Bit Version 2018.2 [27]

Setup

Windows-PC

- Downloade und entpacke das RaspBee Gateway SD Karten-Image
- $\circ~$ Downloade und installiere den SD Card Formatter
- Downloade und installiere den Win32 Disk Imager
- $\circ~$ Stecke die SD Karte in das Notebook und formatiere sie mit dem SD Card Formatter
- $\circ~$ Installiere das RaspBee Gateway SD-Karten Image mit Win32 Disk Imager
 - Wähle das Image aus
 - Überprüfe das angegebene Laufwerk
 - Klicke auf "Write"
- Entferne die SD Karte sicher aus dem Notebook

Raspberry Pi

- Stecke die SD Karte in den dafür vorgesehenen Slot am Raspberry Pi
- Stecke das RaspBee Modul auf die rechten äußeren GPIO Pins (Grafik 10)
- Verbinde den Monitor, die Maus und die Tastatur mit dem Raspberry
 Pi und stecke ihn an die Stromversorgung an



Grafik 10: Raspberry Pi mit aufgestecktem RaspBee-Modul

Philips Hue

 Schraube die Philips Hue Glühbirne in eine passende Fassung und versorge sie mit Strom

Kali Linux

- Downloade Kali Linux
- Stecke den USB-Stick in den Windows-PC
- \circ Starte Win 32 Disk Imager und installiere das Image
 - Wähle das Image aus
 - Überprüfe das angegebene Laufwerk
 - Klicke auf "Write"
- Lass den USB-Stick stecken und starte den PC neu
- Unterbreche den Neustart mit der Hersteller-spezifischen
 Tastenkombination (wird normalerweise am Bildschirm angezeigt)
- $\circ~$ Bei der Frage, von welchem Laufwerk gebootet werden soll, wähle den USB-Stick aus
- \circ Nachdem das Image geladen worden ist, erscheint das Boot-Menü
- $\circ~$ Wähle die erste Option (Live amd
64) aus und warte, bis Kali geladen ist

KillerBee

Öffne in Kali ein Terminal (Strg+Alt+T) und gib die folgenden Befehle ein:

Dependencies

- \$ sudo apt-get install python-gtk2 python-cairo python-usb python-crypto python-serial python-dev libgcrypt-dev
 - installiert die notwendigen Dependencies
- o \$ sudo apt install mercurial
 - o installiert mercurial
- \$ sudo hg clone https://bitbucket.org/secdev/scapy-com
 - klont das Repository in das aktuelle Verzeichnis
- o \$ cd scapy-com
 - $\circ~$ wechselt in das Verzeichnis "scapy-com"

- o \$ sudo python setup.py install
 - führt die Installation aus

KillerBee

- o \$ sudo apt-get install git
 - installiert das git-Kommando
- o \$ mkdir Killerbee
 - erstellt ein neues Verzeichnis "Killerbee"
- \$ cd Killerbee
 - $\circ~$ wechselt in das zuvor erstellte Verzeichnis
- o \$ git clone https://github.com/riverloopsec/killerbee.git
 - klont das Repository in das aktuelle Verzeichnis
- o \$ cd /Killerbee/killerbee/
 - wechselt in das Verzeichnis "killerbee"
- o \$ sudo python setup.py install
 - $\circ~$ führt die Installation aus

Atmel RZ Raven USB-Stick

- Verbinde den AVR Dragon mit dem Atmel RZ Raven USB über das IDC-Kabel (Grafik 11)
 - Achte dabei auf die korrekte Verbindung der Pins (Grafik 12 und 13) [28]:

2	4	6	8	10	9		7	5	3	1
1	3	5	7	9	10	0	8	6	4	2

Grafik 12: AVR Dragon Pin Out [28] Grafik 13: RZ Raven Pin Out [28]

• Verbinde den PC, auf dem Kali

Linux läuft, mit dem AVR Dragon über das USB-Kabel A Stecker auf B Stecker

Firmware-Sicherung

- o \$ sudo apt-get install avrdude
 - o installiert avrdude
- $\circ~$ \$ sudo avrdude -P usb -c dragon_jtag -p usb1287 -U
 - flash:r:Desktop/backup.hex:i
 - $\circ\,$ -P port: identifiziert die Schnittstelle, über die der AVR Dragon verbunden ist
 - \circ -c programmer-id: verwendet das angegebene Programmier-Board
 - -p partno: gibt den Typ des Microcontrollers an, der mit dem Programmier-Board verbunden ist
 - -U memtype:op:filename:filefmt: führt eine Speicher-Operation aus
 - memtype: gibt den Speichertyp an
 - op: gibt die Art der Operation an (r: read, w: write)
 - filename: gibt die Datei an, in die geschrieben werden soll
 - filefmt: beinhaltet das Format der Datei (i: Intel Hex)
- Terminal-Output: avrdude done. Thank you.



Grafik 11: AVR Dragon Board und Atmel RZ Raven Stick

Firmware-Flash

- o \$ sudo avrdude -P usb -c dragon_jtag -p usb1287 -B 10 -U
 flash:w:/Killerbee/killerbee/firmware/kb-rzusbstick-003.hex:a
 - -B bitclock: spezifiziert die bit clock Periode f
 ür das JTAG-Interface oder die ISP-Clock; der Wert wird in Mikrosekunden angegeben
- Terminal-Output: avrdude done. Thank you. [28]

Verifizierung

- Die LED des Atmel USB-Sticks leuchtet nun bernsteinfarben (Grafik
 14)
- Bei Eingabe des Befehls \$ sudo zbid erscheint folgender Output:
 [9]

lstrobl@ubuntu:~\$ sudo zbid					
Dev Product String	Serial Number				
1:4 KILLERB001	FFFFFFFFFFF				



Grafik 14: Der Stick leuchtet jetzt bernsteinfarben

Sniffing

Raspberry Pi I

- $\circ\,$ Öffne den Browser am Raspberry Pi und gib die folgende Adresse in die Adresszeile ein:
 - o **127.0.0.1**
- Melde dich im Login-Fenster mit den folgenden Werten an (Grafik
 15)
 - Username: delight
 - Password: delight
- Es erscheint ein Überblicksfenster



Grafik 15: Browser-Login

Kali-PC I

```
○ Starte Kali Linux

    Stecke den Atmel RZ Raven USB-Stick an

    Öffne ein Terminal und gib die folgenden Befehle ein (Grafik 16):

o $ sudo zbstumbler

    Notiere dir den Channel und beende den Befehl mit Strg+C

o $ sudo wireshark -f <channel>

    ersetze <channel> mit der vorher notierten Zahl

    Wireshark öffnet sich und beginnt, den Netzwerkverkehr

   mitzuschneiden [9]
lstrobl@ubuntu:~$ sudo zbstumbler
[sudo] password for lstrobl:
Warning: You are using pyUSB 1.x, support is in beta.
zbstumbler: Transmitting and receiving on interface '1:4'

New Network: PANID 0x0B64 Source 0x8E85
         Ext PANID: 00:21:2e:ff:ff:01:40:30
                                                     Stack Profile: ZigBee Enterprise
         Stack Version: ZigBee 2006/2007
         Channel: 15
^C
7 packets transmitted, 1 responses.
lstrobl@ubuntu:~$ sudo zbwireshark -f 15
Warning: You are using pyUSB 1.x, support is in beta.
zbwireshark: listening on '1:4'
```

Grafik 16: zbstumbler und zbwireshark

Raspberry Pi II

- Klicke im Überblicksfenster im rechten oberen Eck auf "Menu" und dann auf Settings (Grafik 17)
- Klicke im Settings-Fenster "Scan for Devices" (Grafik 18)
- Die Glühbirne sollte nun angezeigt werden; falls ein roter Reset-Button angezeigt wird, so drücke diesen (Grafik 18)
- Scrolle etwas hinauf und klicke auf "Open Network" (Grafik 18)
- Klicke auf "Control" in der Leiste und klicke im neuen Fenster auf das Pluszeichen
- Klicke auf "Create Group", gib einen Namen ein und erstelle die Gruppe mit "Create" (Grafik 19)
- Klicke auf "Apply"
- Die Glühbirne wurde jetzt hinzugefügt
- $\circ\,$ Klicke "on" zum einschalten und "off" zum ausschalten

Wireless Light	Control Groups	Schedules Devices		Menu 🗸	
Note! A new software of	update is available, to ins	tall it please open the system setti	ings - View Change Log.	Groups Schedules	
Info New lights were found! Settings In order to control them click at the + button to add them to a group. Or Logout Logout					
Name Light 1	Model LWB010	Group +			



Unlock Gateway	Unlock the gateway for 60 seconds to allow 3rd party apps to register themself to the gateway. After the registration they can access the gateway even if it's locked.						
		Apply Done					
Network Settings							
Open Network	Open the network for 1 minute and allow other ZigBee devices to join the network.						
Reset Devices via To	uchlink						
Scan for Devices	Scan for Devices • The devices need to be close (50 cm) to the gateway • Use reset button to reset a device to factory new state • After reset devices can rejoin any network						
Your gateway is on channe	Your gateway is on channel 15 and has the network Id 0x0b64 .						
Lights Name Ado	dress Channel Network Id RSSI						
blink Light 1 0x0	0017880102a200d2 15 0x0b64 -51 reset						
OK! Found 1 light.		Sat, 26 Nov 2016 03:57:07 GMT					

Grafik 18: Scan for Devices, reset, Open Network

Add to Group ×					
Light name Group	Light 1 test	×			
Create Group					
		Apply	Cancel		

Grafik 19: Create Group

Kali-PC II

- Stoppe Wireshark durch Drücken des roten Knopfes in der linken oberen Ecke
- Gehe zu Edit -> Preferences -> Protocols -> ZigBee
- Klicke auf das Plus-Zeichen und gib folgenden Schlüssel ein (Grafik 20):
 - o 5a:69:67:42:65:65:41:6c:6c:69:61:6e:63:65:30:39
 - Dies ist der Default Trust Center Link Key; mit ihm wird der Network Key verschlüsselt [5][6][8][10]
- Sieh dir den mitgeschnittenen Netzwerkverkehr an und suche nach "Transport Key" im Info-Feld (Grafik 21)
- Füge den Schlüssel wie bereits oben beschrieben in Wireshark hinzu (Grafik 22)
- Wireshark entschlüsselt nun automatisch verschlüsselte Pakete mit dem neu hinzugefügten Transport Key
- Abhängig von der Anzahl, wie oft du die Glühbirne aus- und eingeschaltet hast, siehst du jetzt die entschlüsselten Kommandos "ZCL OnOff" (Grafik 23)

VCDU	ZigBee Network Layer				
VLAN	Security Level AES-128 Encrys	otion, 32-bit Integrity Protection			
VNC VP8	Pre-configured Keys Edit			Expressio	n +
VRRP	Laicht		ıfo		
WBXML		😣 🗊 Pre-configured Keys			
WebSocket WiMax (wmx)		Kow	Puto Order Label		
WIMAX ASN C		5a6967426565416c6c69616	e63653039 Normal DTCLK		
WINS-Replica				Ix8e85	- II
X.25				X0000	
X11 X2AP					
XMCP					
XMPP					
YAMI				100	
YMSG ZEP				100	
ZigBee ZigBee Green					-
ZVT					A
vanced 👻					
Þ				nimario	
		+ - 9	/home/lstrobl/.config/wireshark/zigi	bee pc keys	
	Sequence Number: o Extended Source: Philips				
	 ZigBee Security Header Security Control Field 		OK Cancel	Help	
	Frame Counter: 1251792				
	Extended Source: Philip Key Sequence Number: 0	osL_01:02:a2:00:d2 (00:17:88:0	1:02:a2:00:d2)		
	Message Integrity Code	: 5676b391			-
00	00 41 88 b7 64 0b ff ff 85	8e 09 12 fc ff 85 8e 01 A	d		

Grafik 20: Hinzufügen des Default Trust Center Link Keys

<u>F</u> ile	Edit View Go	Capture Analyze Statis	stics Telephon <u>y W</u> irele	ss <u>T</u> ools <u>H</u> el	p	
		🚞 🖹 🖹 🙆 🤇	♦ € 著 ♦	Ł		
Ap	oply a display filte	r <ctrl-></ctrl->				
No.	Time 13 18.170939 14 23.799714 15 24.250142 16 24.254574 17 25.218573 18 30.32321 20 31.473275 21 31.477376 23 31.970290 23 31.97894 24 31.978113 25 31.981847 26 31.998921 27 31.92015	Source 0x0000 0x0000 0x0000 0x0000 0x0000 00:17:88:01:02:a2:0 00:17:88:01:02:a2:0 00:21:2e:ff:ff:01:4 0x0000	Destination Broadcast Broadcast Broadcast Broadcast Broadcast 0x0000 0x0000 00:17:88:01:02:a2:0 0x573b	Protocol Lengt IEEE 8 1 ZigBee 4 ZigBee 2 ZigBee 2 IEEE 8 1 ZigBee 2 IEEE 8 2 IEEE 8 1 IEEE 8 1 IEEE 8 2 IEEE 8 2	<pre>Info 0 Beacon Request 7 Command, Dst: Broadcast, Src: 0x0000 0 Beacon Request 8 Beacon, Src: 0x0000, EPID: Dresdenff:ff:01:40:30 8 Data, Dst: Broadcast, Src: 0x0000 9 Deacon Request 8 Beacon, Src: 0x0000, EPID: Dresdenff:ff:01:40:30 1 Association Request 5 Ack 8 Data Request 5 Ack 9 Tassport Key 5 Ack 9 Transport Key 5 Ack 9 Data Output Differ Output Differ Output Differ 9 Ack 9 Data Differ Output Differ Outpu</pre>	
 Fr: IEI 74 	28 32.229755 29 32.244827 30 32.246584 31 32.248452 32 32.250637 ame 26: 73 byte EE 802.15.4 Dat	0x573b 0x573b 0x0000 s on wire (584 bits), 7 a, Dst: 0x573b, Src: 0x	0x0000 0x0000 0x573b 3 bytes captured (584 0000	ZigBee 7 ZigBee 7 IEEE 8 ZigBee 5 IEEE 8 bits) on inte	7 Route Reply, Dst: 0x573b, Src: 0x0000 7 Route Reply, Dst: 0x573b, Src: 0x0000 5 Ack 0 ZCL Groups: Remove a Group, Seq: 21 5 Ack erface 0	
 ZigBee Network Layer Data, Dst: 0x0730, SrC: 0x0000 ZigBee Application Support Layer Command Frame Control Field: Command (0x21) Counter: 127 ZigBee Security Header Command Frame: Transport Key Command Identifier: Transport Key (0x05) Key Type: Standard Network Key (0x01) Key: 000048006666474902cr6c6402cr4982c 						
	Sequence Number: 0 Extended Destination: PhilipsL_01:02:a2:00:d2 (00:17:88:01:02:a2:00:d2) Extended Source: Dresdenff:ff:01:40:30 (00:21:2e:ff:ff:01:40:30)					
0000 0010 0020	0000 05 01 00 00 48 00 6c 64 f4 90 2c fc fc 40 2c f4					



🕲 🕒 Wireshark · Preferences	
USBCCID VCDU VTIA 49 VLAN VNC VP8 VRP Vuze-DHT WBSML WebSocket WiMAX ASN C WiMAX ASN C XACP XML XML XMP XACP XMI Y ASI ZIGBEE ZIGBEE ZIGBEE ASI X X	rk Layer AE5-128 Encryption, 32-bit Integrity Protection Keys Edit Reys Edit Reys Byte Order Label 5a:69:67:42:65:65:41:6c:66:63:65:30:39 Normal DTCLK 00:00:48:00:6c:64:f4:90:2c:fc:fc:40:2c:f4:98:2c Normal Network Key 30
 Command Frame: Transport Key Command Identifier: Transport Key Type: Standard Network Key: 000048006064f4902crcrd Sequence Number: 0 Extended Destination: Phil: Extended Source: Dresden1 	+ - Po /home/lstrobl/.config/wireshark/zigbee pc keys rrt Key (0x05) OK Key (0x01) OK 402cr4982c OK psL_01:02:a2:00:d2 (00:17:88:01:02:a2:00:d2) f:ff:01:40:30 (00:21:2e:ff:ff:01:40:30)
0000 05 01 00 00 48 00 6c 64 f4 9 0010 98 2c 00 d2 00 a2 02 01 88 3 0020 2e 21 00	0 2c fc fc 40 2c f4

Grafik 22: Hinzufügen des Transport Keys

File	<u>E</u> dit <u>V</u> iew <u>G</u> o	<u>Capture</u> Analyze	Statistics Telephony Wi	ireless <u>T</u> ools <u>H</u> elp
			। ९ 🗢 🔿 🖀 🗿	
A	pply a display filte	er <ctrl-></ctrl->		
No.	Time	Source	Destination	Protocol Length Info
	91 39.226317 92 39.228405	0x573b	0×0000	ZigBee… 72 Link Quality Response, Status: Succes IEEE 8… 5 Ack
	93 39.231956 94 39.232821	0×0000	0x573b	ZigBee 45 APS: Ack, Dst Endpt: 0, Src Endpt: 0 IEEE 8 5 Ack
	95 40.305303	0x573b	Broadcast	ZigBee 50 Link Status
	96 42.880350	0×0000	Broadcast	ZigBee… 49 ZCL OnOff: Off, Seq: 34
	97 42.964650	0x0000	Broadcast	ZigBee… 49 ZCL OnOff: Off, Seq: 34
	98 44.014899 99 44.018821	0×0000	0x573b	ZigBee… 47 Link Quality Request IEEE 8… 5 Ack
	100 44.029257 101 44.031159	0x573b	0×0000	ZigBee… 72 Link Quality Response, Status: Succes IEEE 8… 5 Ack
	102 44.035777 103 44.037313	0×0000	0x573b	ZigBee 45 APS: Ack, Dst Endpt: 0, Src Endpt: 0 IEEE 8 5 Ack
	104 45.219648	0×0000	Broadcast	ZigBee… 49 ZCL OnOff: On, Seq: 35
	105 45.265824	0x0000	Broadcast	ZigBee… 49 ZCL OnOff: On, Seq: 35
	106 48.815949 107 48.817392	0×0000	0x573b	ZigBee… 47 Link Quality Request IEEE 8… 5 Ack
	108 48.819960 109 48.821394	0×0000	0x573b	ZigBee… 47 Link Quality Request IEEE 8… 5 Ack
	110 48.829468	0x573b	0×0000	ZigBee… 72 Link Quality Response, Status: Succes
4				

Frame 96: 49 bytes on wire (392 bits), 49 bytes captured (392 bits) on interface 0
 IEEE 802.15.4 Data, Dst: Broadcast, Src: 0x0000
 ZigBee Network Layer Data, Dst: Broadcast, Src: 0x0000
 ZigBee Application Support Layer Data, Group: 0x0007, Src Endpt: 1
 Frame Control Field: Data (0x0c) Group: 0x0007
 Cluster: On/Off (0x0006) Profile: Home Automation (0x0104) Source Endpoint: 1
 ZigBee Cluster Library Frame

Grafik 23: Entschlüsseltes OnOff-Kommando

QUELLEN

Quellen

[1] ZigBee Remote Control 2.0: Updated Standard for Radio Frequency-Based Remote Controls, October 2014

[2] ZigBee RF4CE: A Quiet Revolution is Underway December 6, 2012

[3] https://research.kudelskisecurity.com/2017/11/01/zigbee-security-basicspart-1/

[4] https://de.wikipedia.org/wiki/ZigBee

[5] Niko Vidgren, Keijo Haataja, José Luis Patino-Andres, Juan José Ramírez-Sanchis and Pekka Toivanen, "Security Threats in ZigBee-Enabled Systems: Vulnerability Evaluation, Practical Experiments, Countermeasures, and Lessons Learned," 2013

[6] Philipp Morgner, Stephan Mattejat and Zinaida Benenson, "All Your Bulbs Are Belong to Us: Investigating the Current State of Security in Connected Lighting Systems," 2017

[7] https://research.kudelskisecurity.com/2017/11/08/zigbee-security-basicspart-2/

[8] https://research.kudelskisecurity.com/2017/11/21/zigbee-security-basicspart-3/

[9] http://securitysynapse.blogspot.com/2015/11/fun-with-zigbee-wirelesspart-iii.html

[10] Tobias Zillner, "ZigBee Exploited - The good, the bad and the ugly," Black Hat 2015 Power Point Präsentation

[11] https://www.elprocus.com/what-is-zigbee-technology-architecture-and-itsapplications/

[12] https://at.rs-online.com/web/p/entwicklungskits-prozessormikrocontroller/1373331/

[13] https://www.dresden-elektronik.de/raspbee/

[14] https://www2.meethue.com/de-de/p/hue-white-einzelne-lampee27/8718696449578

[15] http://at.farnell.com/atmel/atavrzusbstick/kit-2-4ghz-rzraven-usbstick/dp/1562234 [16] https://www.reichelt.de/Programmer-Entwicklungstools/AT-AVR-DRAGON/3/index.html?ACTION=3&GROUPID=5514&ARTICLE=97200

[17] https://www.startech.com/de/Kabel/USB-2.0/USB-2.0-Kabel/05-m-High-Speed-USB-20-Kabel~USB2HAB50CM

[18] https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/

[19] https://www.dresden-elektronik.de/raspbee/

[21] https://www.microchip.com/webdoc/rzraven/rzraven.RZ_Raven_module.html

[22]https://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ATA
VRDRAGON

[23] https://www.dresden-elektronik.de/funktechnik/solutions/wireless-lightcontrol/raspbee-gw-sd-card-image/

[24] https://www.sdcard.org/downloads/formatter_4/index.html

[25] https://sourceforge.net/projects/win32diskimager/

[26] https://github.com/riverloopsec/killerbee

[27] https://www.kali.org/downloads/

[28] https://medium.com/@netscylla/zigbee-killerbee-1a35af5ef4f4

[29] https://www2.meethue.com/de-at/p/hue-white-einzelne-lampee27/8718696449578