Zigator: Analyzing the Security of Zigbee-Enabled Smart Homes

Dimitrios-Georgios Akestoridis, Madhumitha Harishankar, Michael Weber, and Patrick Tague

Carnegie Mellon University

ACM WiSec 2020

Motivation

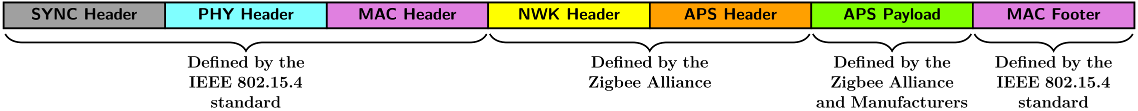
- Smart home network security affects the physical security of residents

Motivation

- Smart home network security affects the physical security of residents
- Zigbee supports two security models:
    - **Distributed** $\Rightarrow$ recommended for ease of use
    - **Centralized** $\Rightarrow$ recommended for higher security

Motivation

- Smart home network security affects the physical security of residents
- Zigbee supports two security models:
  - **Distributed** ⇒ recommended for ease of use
  - **Centralized** ⇒ recommended for higher security
- High-level view of a Zigbee packet without any security features:



SYNC Header | PHY Header | MAC Header | NWK Header | APS Header | APS Payload | MAC Footer

Defined by the IEEE 802.15.4 standard

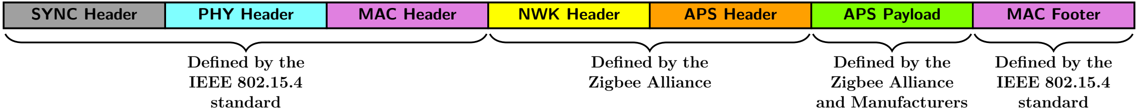
Defined by the Zigbee Alliance

Defined by the Zigbee Alliance and Manufacturers

Defined by the IEEE 802.15.4 standard

Motivation

- Smart home network security affects the physical security of residents
- Zigbee supports two security models:
  - **Distributed** ⇒ recommended for ease of use
  - **Centralized** ⇒ recommended for higher security
- High-level view of a Zigbee packet without any security features:



SYNC Header | PHY Header | MAC Header | NWK Header | APS Header | APS Payload | MAC Footer

Defined by the IEEE 802.15.4 standard

Defined by the Zigbee Alliance

Defined by the Zigbee Alliance and Manufacturers

Defined by the IEEE 802.15.4 standard

We study the security consequences of the design choice to disable **MAC-layer security** in centralized Zigbee networks

Threat Model and Assumptions

- **Security objectives:**
  - Authenticity, Integrity, Confidentiality, and Availability

Threat Model and Assumptions

- **Security objectives:**
  - Authenticity, Integrity, Confidentiality, and Availability
- **Assumptions:**
  - The end user and their devices are trusted

Threat Model and Assumptions

- **Security objectives:**
  - Authenticity, Integrity, Confidentiality, and Availability
- **Assumptions:**
  - The end user and their devices are trusted
  - The attacker is an outsider with potentially more powerful hardware
  - The attacker has no prior knowledge of any network key
  - The attacker is aware of the default Trust Center link key
  - The attacker may have access to a subset of install codes

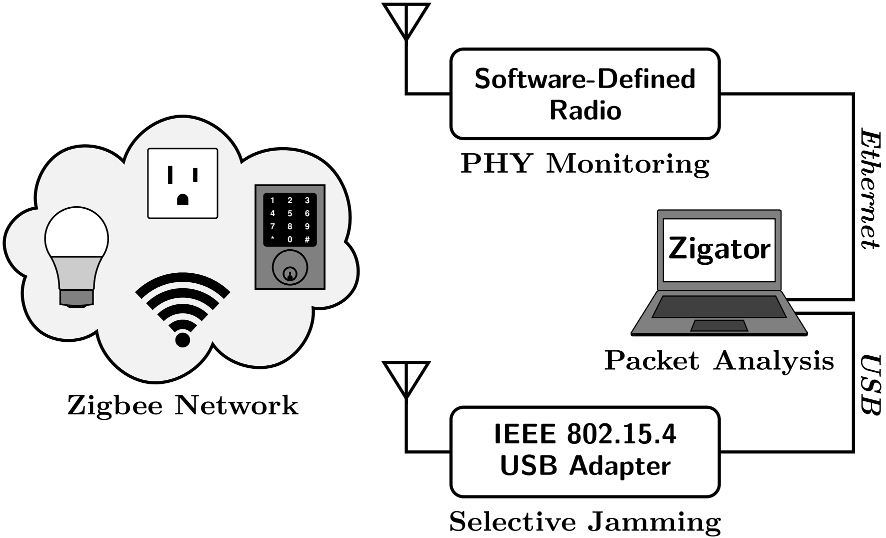
Threat Model and Assumptions

- **Security objectives:**
  - Authenticity, Integrity, Confidentiality, and Availability
- **Assumptions:**
  - The end user and their devices are trusted
  - The attacker is an outsider with potentially more powerful hardware
  - The attacker has no prior knowledge of any network key
  - The attacker is aware of the default Trust Center link key
  - The attacker may have access to a subset of install codes
  - We do not consider uncommon device configurations like low-power routers

Threat Model and Assumptions

- **Security objectives:**
  - Authenticity, Integrity, Confidentiality, and Availability
- **Assumptions:**
  - The end user and their devices are trusted
  - The attacker is an outsider with potentially more powerful hardware
  - The attacker has no prior knowledge of any network key
  - The attacker is aware of the default Trust Center link key
  - The attacker may have access to a subset of install codes
  - We do not consider uncommon device configurations like low-power routers
- **Attacker’s goal:**
  - Obtaining the network key from an already formed Zigbee network

Security Analysis with Zigator



Zigbee Network

Software-Defined Radio

PHY Monitoring

Ethernet

Zigator

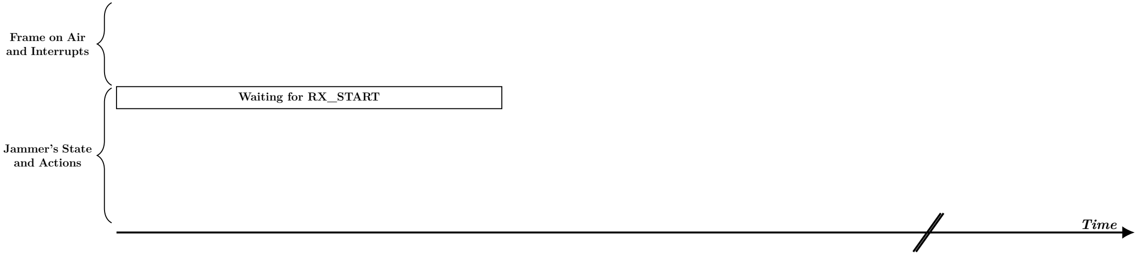
Packet Analysis

USB

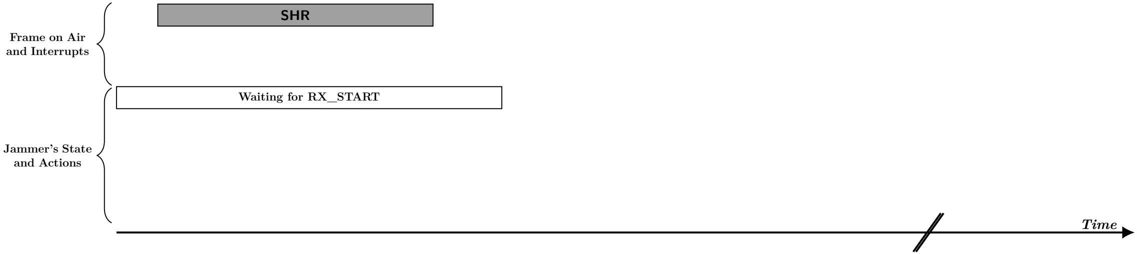
IEEE 802.15.4 USB Adapter

Selective Jamming

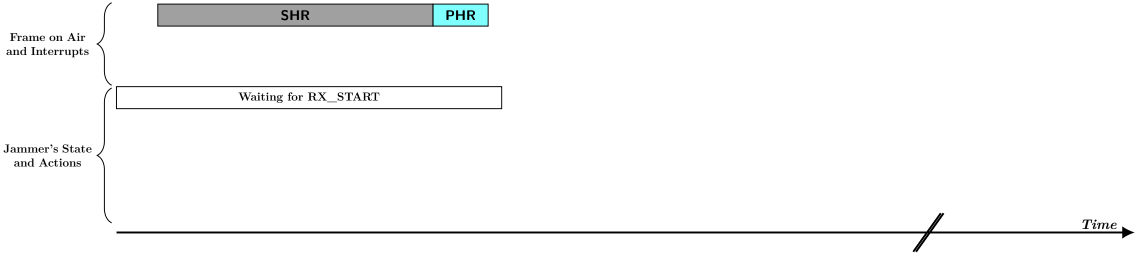
Our implementation of a selective jammer



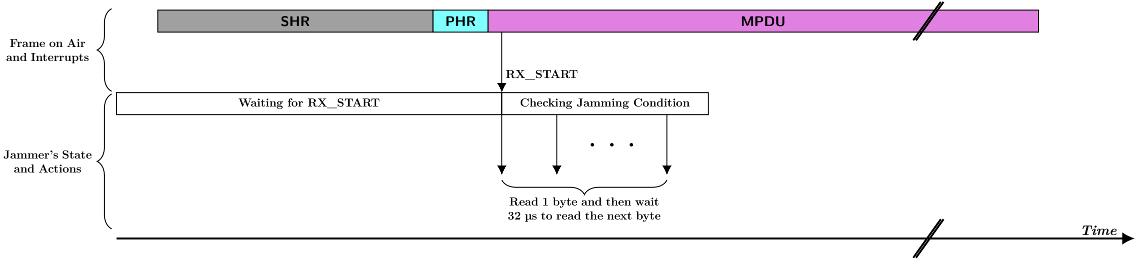
Our implementation of a selective jammer



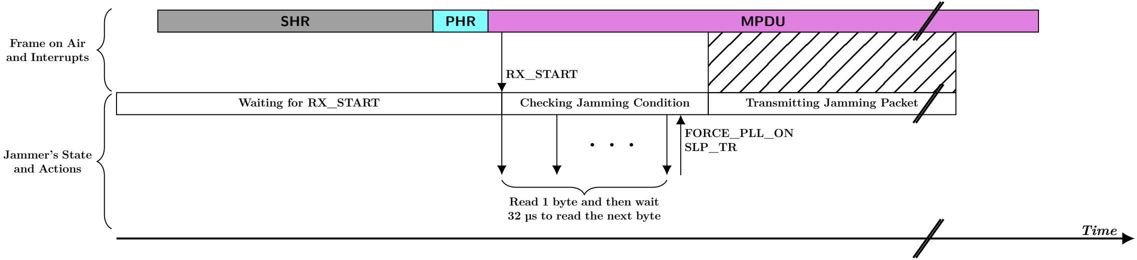
Our implementation of a selective jammer



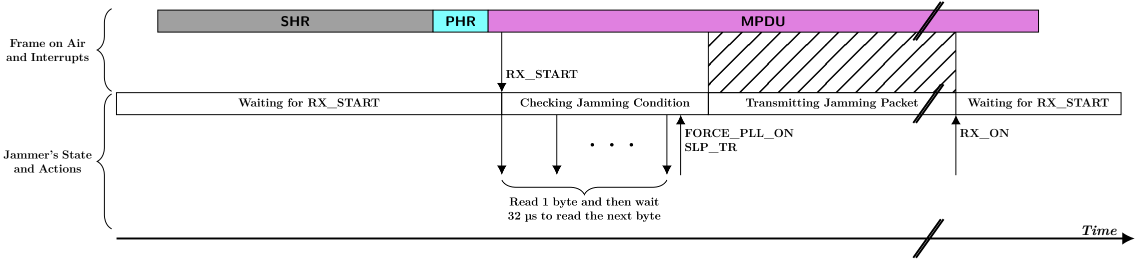
Our implementation of a selective jammer


SHR
PHR
MPDU
Frame on Air and Interrupts
RX_START
Waiting for RX_START
Checking Jamming Condition
Jammer's State and Actions
. . .
Read 1 byte and then wait 32 µs to read the next byte
Time

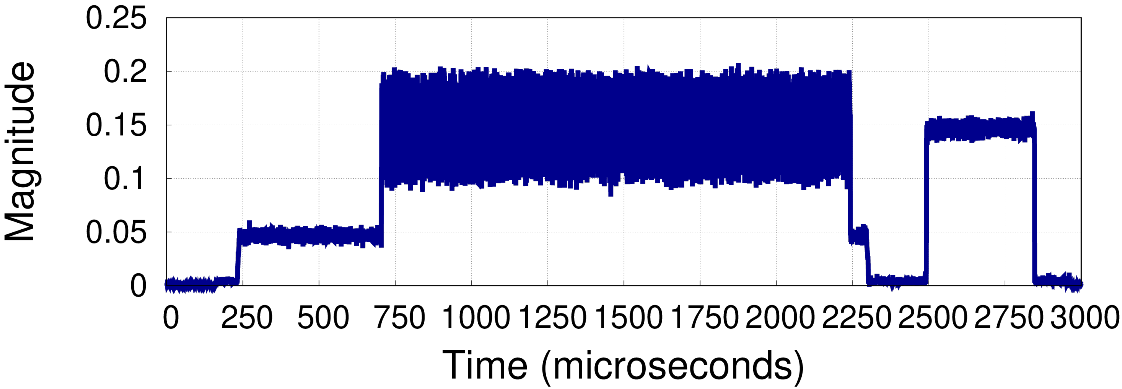
Our implementation of a selective jammer


Frame on Air and Interrupts
SHR
PHR
MPDU
RX_START
Waiting for RX_START
Checking Jamming Condition
Transmitting Jamming Packet
FORCE_PLL_ON
SLP_TR
Jammer's State and Actions
Read 1 byte and then wait 32 µs to read the next byte
Time

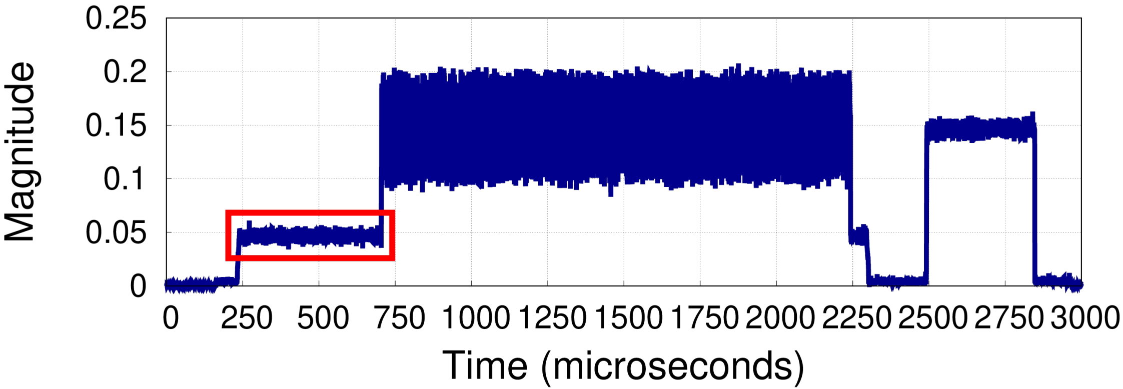
Our implementation of a selective jammer


SHR
PHR
MPDU
Frame on Air and Interrupts
RX_START
Waiting for RX_START
Checking Jamming Condition
Transmitting Jamming Packet
Waiting for RX_START
FORCE_PLL_ON
SLP_TR
RX_ON
Jammer's State and Actions
Read 1 byte and then wait 32 µs to read the next byte
Time

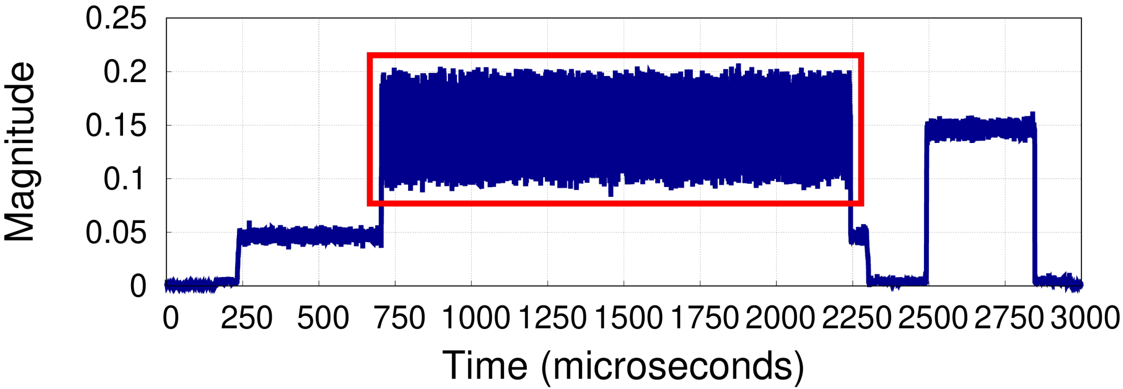
Combining Core Functionalities



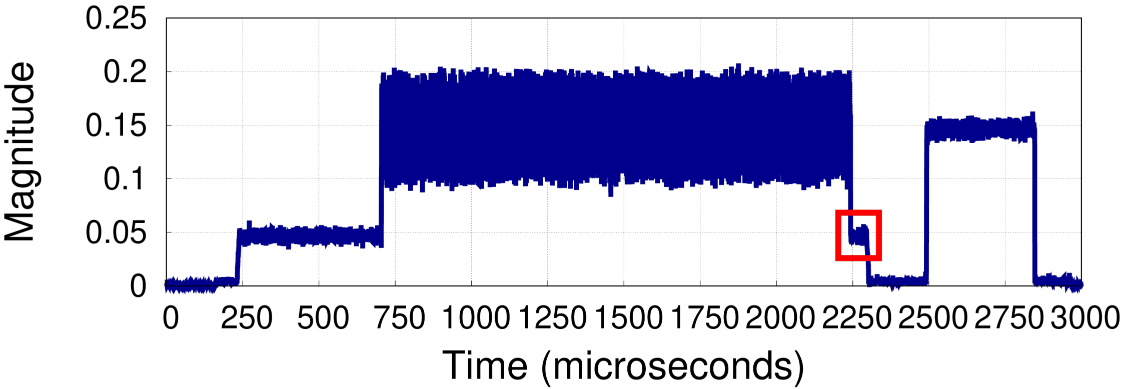
Combining Core Functionalities



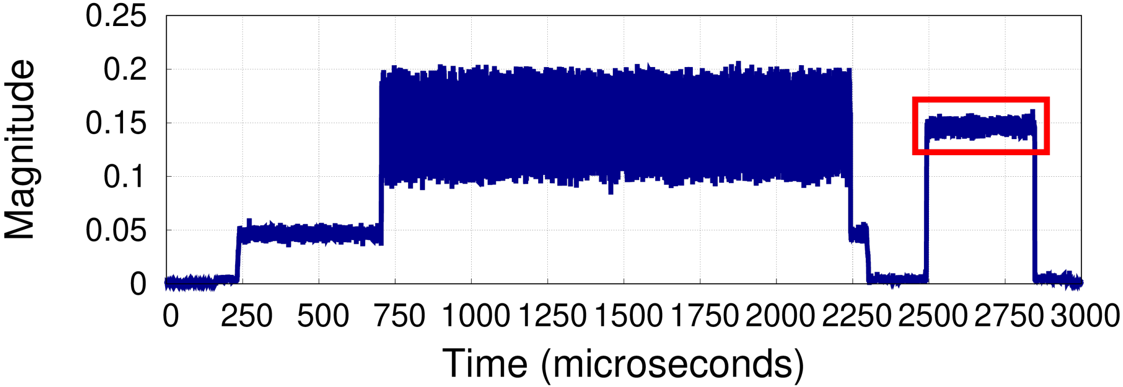
Combining Core Functionalities



Combining Core Functionalities

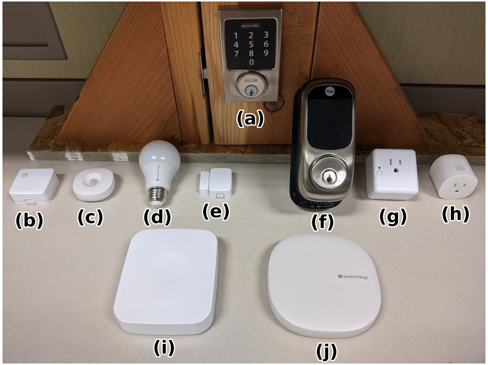


Combining Core Functionalities



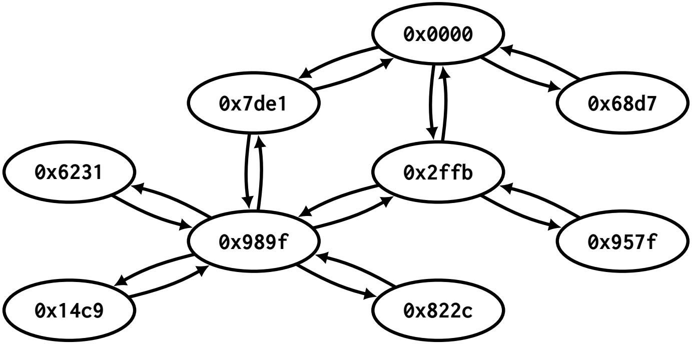
Experimental Setup

- We captured packets that were generated from **ten commercial Zigbee devices**
- We conducted **eight experiments** that differed in the smart hub that was used and the physical topology of the devices
- Our experiments lasted about 34.644 hours in total and resulted in a dataset of **571,509 valid packets**

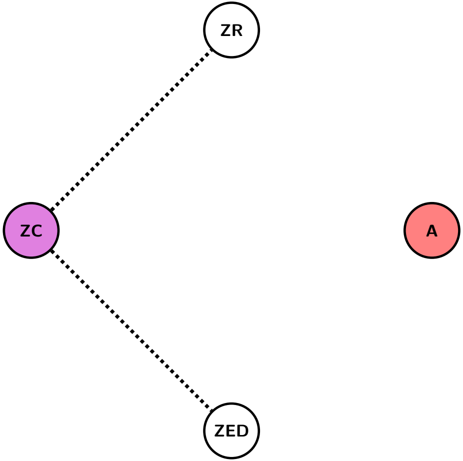


Inferring the Topology of a Zigbee Network

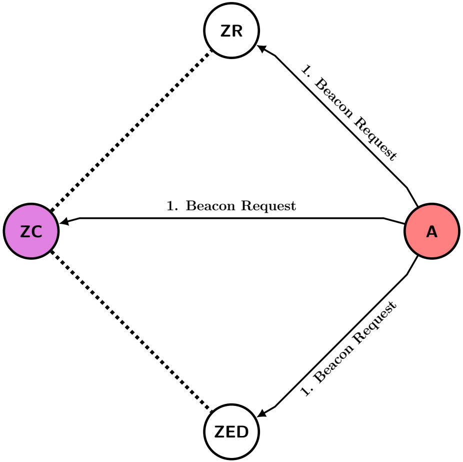
- Log distinct pairs of source and destination addresses
- Trivial identification of the **Zigbee Coordinator** ⇒ always `0x0000`



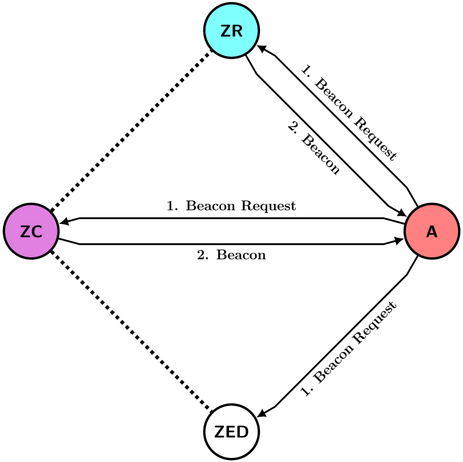
Identifying Logical Device Types



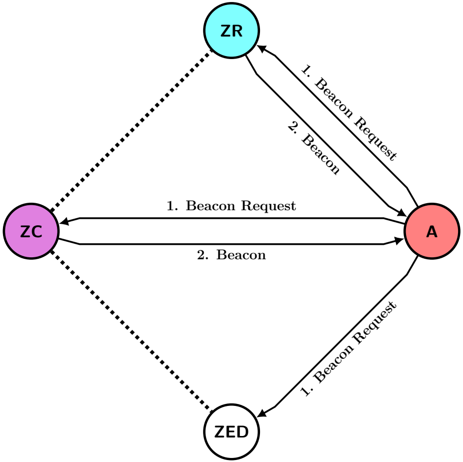
Identifying Logical Device Types

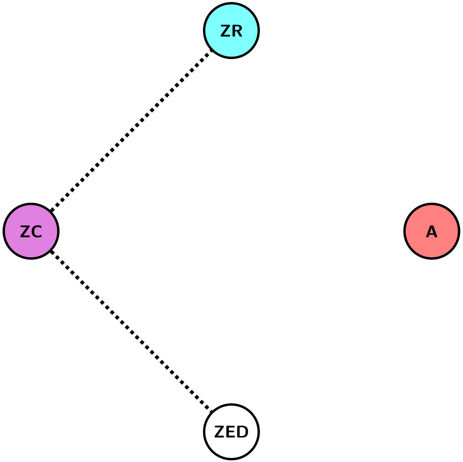


Identifying Logical Device Types

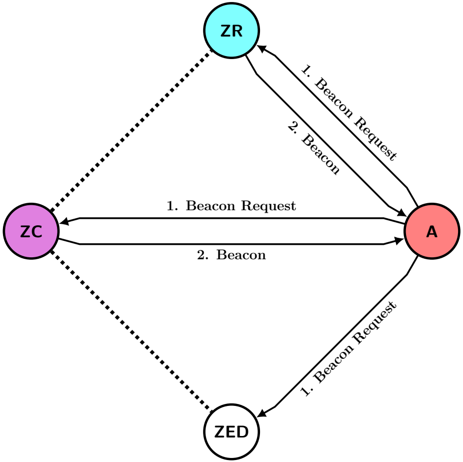


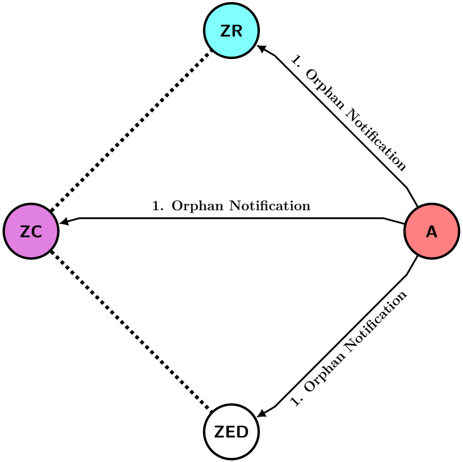
Identifying Logical Device Types



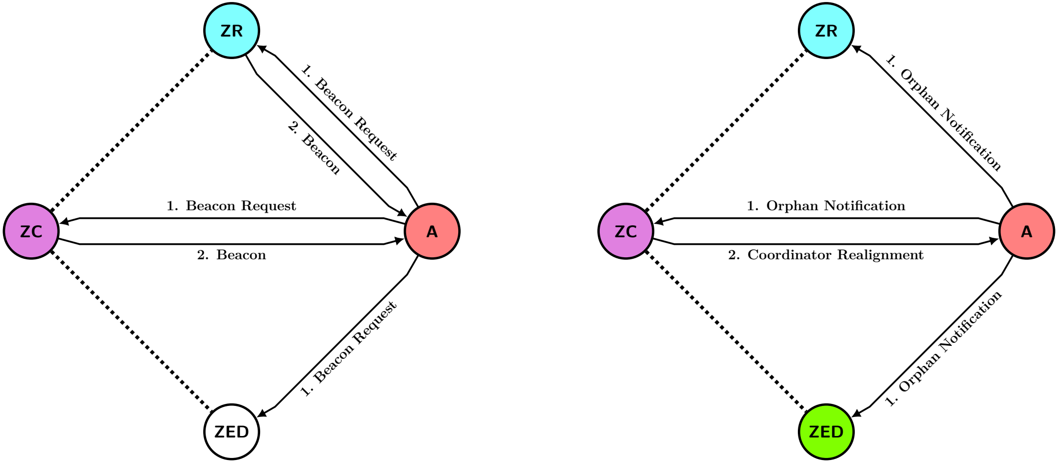


Identifying Logical Device Types

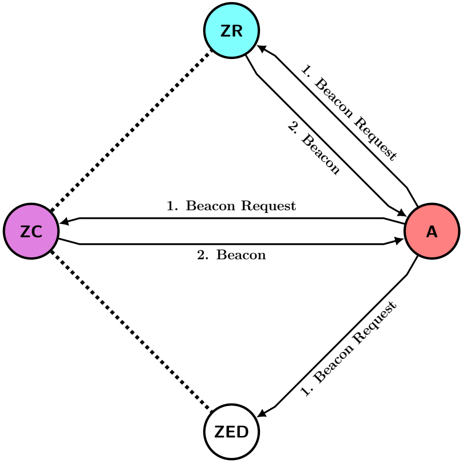


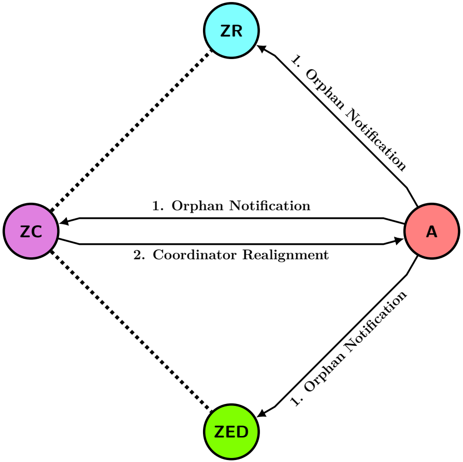


Identifying Logical Device Types



Identifying Logical Device Types





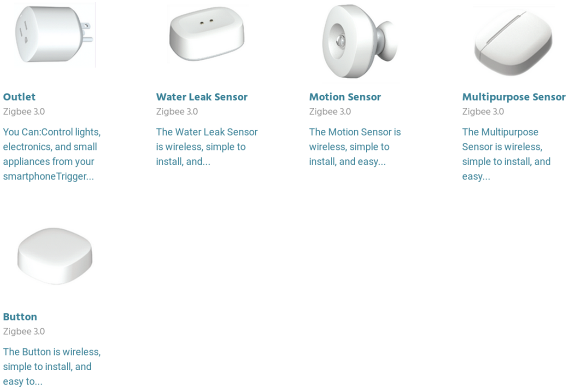
Passive identification based on **Data Request** and **Link Status** commands

Examining Short and Extended Addresses

- **NWK commands** contain both the extended and the short address of their source
- The extended address in the **auxiliary header of the NWK layer** matches with the short address of the source in the MAC header

Examining Short and Extended Addresses

- **NWK commands** contain both the extended and the short address of their source
- The extended address in the **auxiliary header of the NWK layer** matches with the short address of the source in the MAC header
- 28:6d:97:00:01:09:4b:c8
  ⇒ 0x286d97
  ⇒ **SAMJIN Co., Ltd.**



Outlet
Zigbee 3.0
You Can:Control lights, electronics, and small appliances from your smartphoneTrigger...

Water Leak Sensor
Zigbee 3.0
The Water Leak Sensor is wireless, simple to install, and...

Motion Sensor
Zigbee 3.0
The Motion Sensor is wireless, simple to install, and easy...

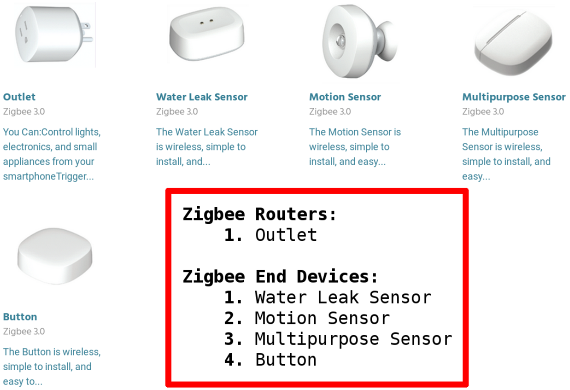
Multipurpose Sensor
Zigbee 3.0
The Multipurpose Sensor is wireless, simple to install, and easy...

Button
Zigbee 3.0
The Button is wireless, simple to install, and easy to...

Source: https://zigbeealliance.org/product_type/certified_product/

Examining Short and Extended Addresses

- **NWK commands** contain both the extended and the short address of their source
- The extended address in the **auxiliary header of the NWK layer** matches with the short address of the source in the MAC header
- 28:6d:97:00:01:09:4b:c8
  ⇒ 0x286d97
  ⇒ **SAMJIN Co., Ltd.**



Outlet
Zigbee 3.0
You Can:Control lights, electronics, and small appliances from your smartphoneTrigger...

Water Leak Sensor
Zigbee 3.0
The Water Leak Sensor is wireless, simple to install, and...

Motion Sensor
Zigbee 3.0
The Motion Sensor is wireless, simple to install, and easy...

Multipurpose Sensor
Zigbee 3.0
The Multipurpose Sensor is wireless, simple to install, and easy...

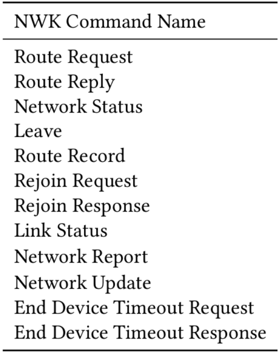
Button
Zigbee 3.0
The Button is wireless, simple to install, and easy to...

```
Zigbee Routers:
    1. Outlet

Zigbee End Devices:
    1. Water Leak Sensor
    2. Motion Sensor
    3. Multipurpose Sensor
    4. Button
```

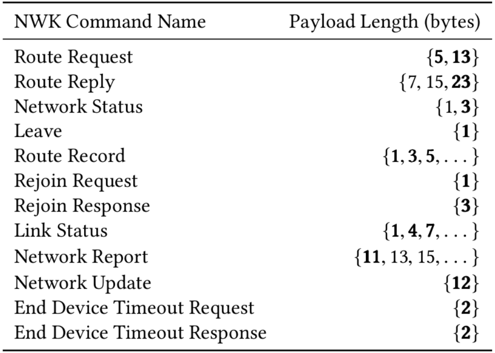
Source: https://zigbeealliance.org/product_type/certified_product/

Identifying Encrypted NWK Commands



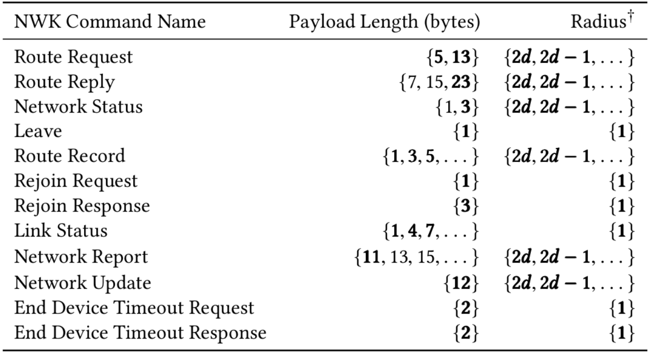
NWK Command Name
Route Request
Route Reply
Network Status
Leave
Route Record
Rejoin Request
Rejoin Response
Link Status
Network Report
Network Update
End Device Timeout Request
End Device Timeout Response

Identifying Encrypted NWK Commands



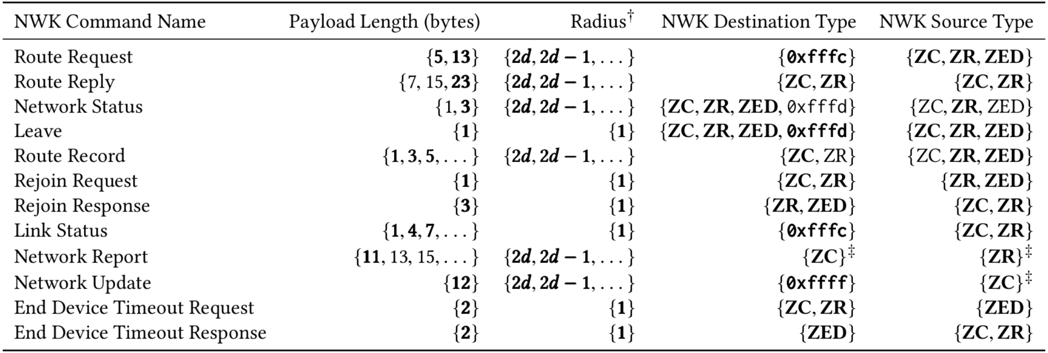
NWK Command Name	Payload Length (bytes)
Route Request	{**5**, **13**}
Route Reply	{7, 15, **23**}
Network Status	{1, **3**}
Leave	{**1**}
Route Record	{**1**, **3**, **5**, . . . }
Rejoin Request	{**1**}
Rejoin Response	{**3**}
Link Status	{**1**, **4**, **7**, . . . }
Network Report	{**11**, 13, 15, . . . }
Network Update	{**12**}
End Device Timeout Request	{**2**}
End Device Timeout Response	{**2**}

Identifying Encrypted NWK Commands



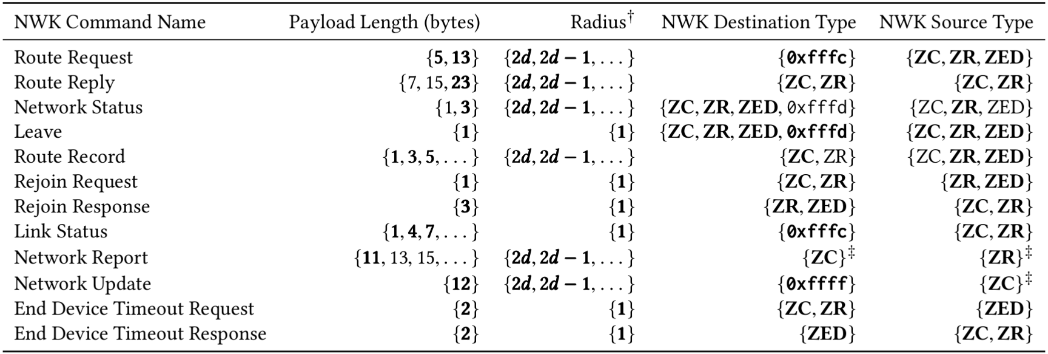
NWK Command Name	Payload Length (bytes)	Radius†
Route Request	$\{\mathbf{5}, \mathbf{13}\}$	$\{\boldsymbol{2d}, \boldsymbol{2d-1}, \ldots\}$
Route Reply	$\{7, 15, \mathbf{23}\}$	$\{\boldsymbol{2d}, \boldsymbol{2d-1}, \ldots\}$
Network Status	$\{1, \mathbf{3}\}$	$\{\boldsymbol{2d}, \boldsymbol{2d-1}, \ldots\}$
Leave	$\{\mathbf{1}\}$	$\{\mathbf{1}\}$
Route Record	$\{\mathbf{1}, \mathbf{3}, \mathbf{5}, \ldots\}$	$\{\boldsymbol{2d}, \boldsymbol{2d-1}, \ldots\}$
Rejoin Request	$\{\mathbf{1}\}$	$\{\mathbf{1}\}$
Rejoin Response	$\{\mathbf{3}\}$	$\{\mathbf{1}\}$
Link Status	$\{\mathbf{1}, \mathbf{4}, \mathbf{7}, \ldots\}$	$\{\mathbf{1}\}$
Network Report	$\{\mathbf{11}, 13, 15, \ldots\}$	$\{\boldsymbol{2d}, \boldsymbol{2d-1}, \ldots\}$
Network Update	$\{\mathbf{12}\}$	$\{\boldsymbol{2d}, \boldsymbol{2d-1}, \ldots\}$
End Device Timeout Request	$\{\mathbf{2}\}$	$\{\mathbf{1}\}$
End Device Timeout Response	$\{\mathbf{2}\}$	$\{\mathbf{1}\}$

Identifying Encrypted NWK Commands



NWK Command Name	Payload Length (bytes)	Radius†	NWK Destination Type	NWK Source Type
Route Request	{**5**, **13**}	$\{\mathbf{2}\boldsymbol{d}, \mathbf{2}\boldsymbol{d}-\mathbf{1}, \ldots\}$	{**0xfffc**}	{**ZC**, **ZR**, **ZED**}
Route Reply	{7, 15, **23**}	$\{\mathbf{2}\boldsymbol{d}, \mathbf{2}\boldsymbol{d}-\mathbf{1}, \ldots\}$	{**ZC**, **ZR**}	{**ZC**, **ZR**}
Network Status	{1, **3**}	$\{\mathbf{2}\boldsymbol{d}, \mathbf{2}\boldsymbol{d}-\mathbf{1}, \ldots\}$	{**ZC**, **ZR**, **ZED**, 0xfffd}	{ZC, **ZR**, ZED}
Leave	{**1**}	{**1**}	{**ZC**, **ZR**, **ZED**, **0xfffd**}	{**ZC**, **ZR**, **ZED**}
Route Record	{**1**, **3**, **5**, . . . }	$\{\mathbf{2}\boldsymbol{d}, \mathbf{2}\boldsymbol{d}-\mathbf{1}, \ldots\}$	{**ZC**, ZR}	{ZC, **ZR**, **ZED**}
Rejoin Request	{**1**}	{**1**}	{**ZC**, **ZR**}	{**ZR**, **ZED**}
Rejoin Response	{**3**}	{**1**}	{**ZR**, **ZED**}	{**ZC**, **ZR**}
Link Status	{**1**, **4**, **7**, . . . }	{**1**}	{**0xfffc**}	{**ZC**, **ZR**}
Network Report	{**11**, 13, 15, . . . }	$\{\mathbf{2}\boldsymbol{d}, \mathbf{2}\boldsymbol{d}-\mathbf{1}, \ldots\}$	{ZC}‡	{**ZR**}‡
Network Update	{**12**}	$\{\mathbf{2}\boldsymbol{d}, \mathbf{2}\boldsymbol{d}-\mathbf{1}, \ldots\}$	{**0xffff**}	{**ZC**}‡
End Device Timeout Request	{**2**}	{**1**}	{**ZC**, **ZR**}	{**ZED**}
End Device Timeout Response	{**2**}	{**1**}	{**ZED**}	{**ZC**, **ZR**}

Identifying Encrypted NWK Commands

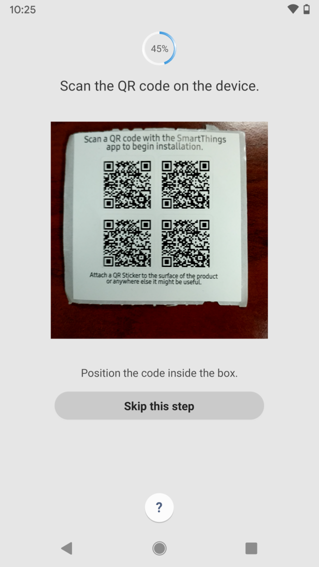


NWK Command Name	Payload Length (bytes)	Radius†	NWK Destination Type	NWK Source Type
Route Request	{**5**, **13**}	{***2d***, ***2d* − 1**, . . . }	{**`0xfffc`**}	{**ZC**, **ZR**, **ZED**}
Route Reply	{7, 15, **23**}	{***2d***, ***2d* − 1**, . . . }	{**ZC**, **ZR**}	{**ZC**, **ZR**}
Network Status	{1, **3**}	{***2d***, ***2d* − 1**, . . . }	{**ZC**, **ZR**, **ZED**, `0xfffd`}	{ZC, **ZR**, ZED}
Leave	{**1**}	{**1**}	{**ZC**, **ZR**, **ZED**, **`0xfffd`**}	{**ZC**, **ZR**, **ZED**}
Route Record	{**1**, **3**, **5**, . . . }	{***2d***, ***2d* − 1**, . . . }	{**ZC**, ZR}	{ZC, **ZR**, **ZED**}
Rejoin Request	{**1**}	{**1**}	{**ZC**, **ZR**}	{**ZR**, **ZED**}
Rejoin Response	{**3**}	{**1**}	{**ZR**, **ZED**}	{**ZC**, **ZR**}
Link Status	{**1**, **4**, **7**, . . . }	{**1**}	{**`0xfffc`**}	{**ZC**, **ZR**}
Network Report	{**11**, 13, 15, . . . }	{***2d***, ***2d* − 1**, . . . }	{**ZC**}‡	{**ZR**}‡
Network Update	{**12**}	{***2d***, ***2d* − 1**, . . . }	{**`0xffff`**}	{**ZC**}‡
End Device Timeout Request	{**2**}	{**1**}	{**ZC**, **ZR**}	{**ZED**}
End Device Timeout Response	{**2**}	{**1**}	{**ZED**}	{**ZC**, **ZR**}

The **decision tree** that we developed is included in our paper

Commissioning of Zigbee Devices

- Legacy Zigbee devices use the **default Trust Center link key** to join a network
- A Zigbee 3.0 device can join a Zigbee 3.0 network using an **install code**



Scan the QR code on the device.

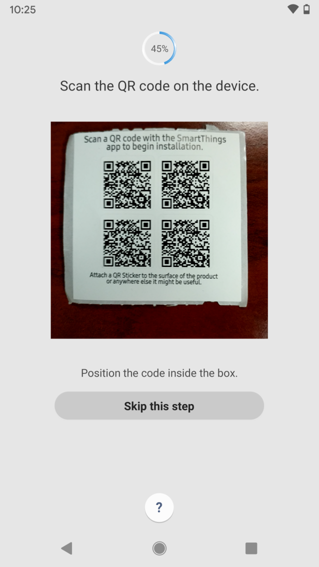
Scan a QR code with the SmartThings app to begin installation.

Attach a QR Sticker to the surface of the product or anywhere else it might be useful.

Position the code inside the box.

Commissioning of Zigbee Devices

- Legacy Zigbee devices use the **default Trust Center link key** to join a network
- A Zigbee 3.0 device can join a Zigbee 3.0 network using an **install code**
- The attacker's main strategy is to launch a **denial-of-service attack** that would force the end user to **factory reset** a device that uses a known Trust Center link key

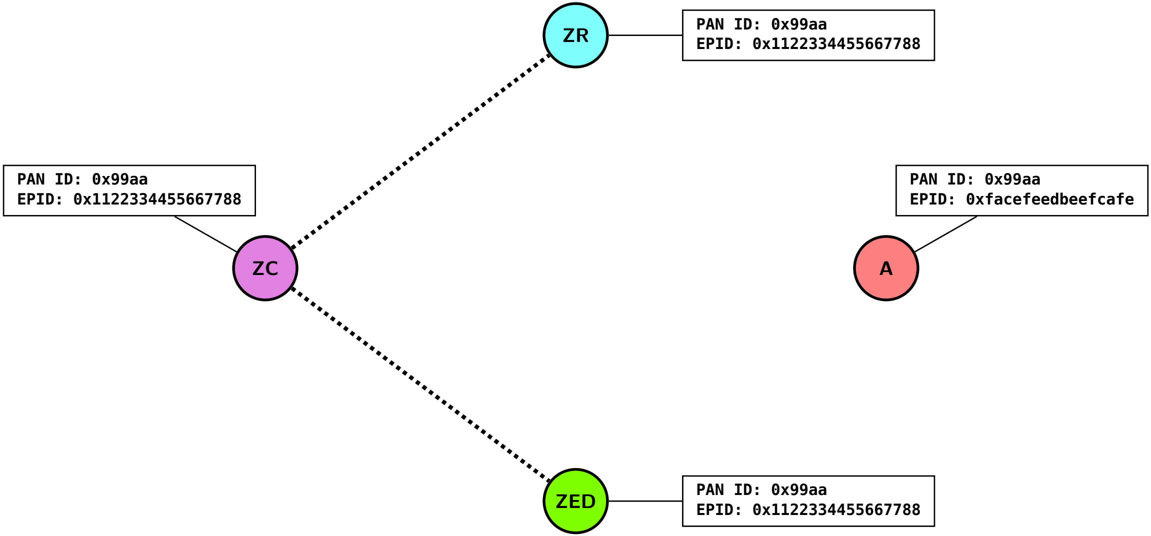


Scan the QR code on the device.

Scan a QR code with the SmartThings app to begin installation.
Attach a QR Sticker to the surface of the product or anywhere else it might be useful.

Position the code inside the box.

Disconnecting Zigbee Devices



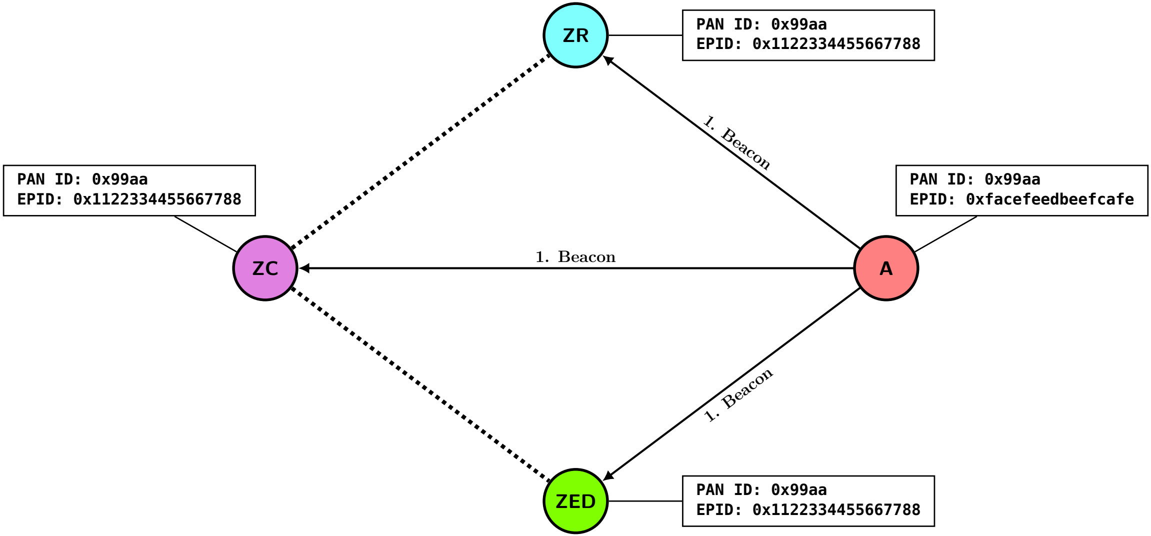
PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

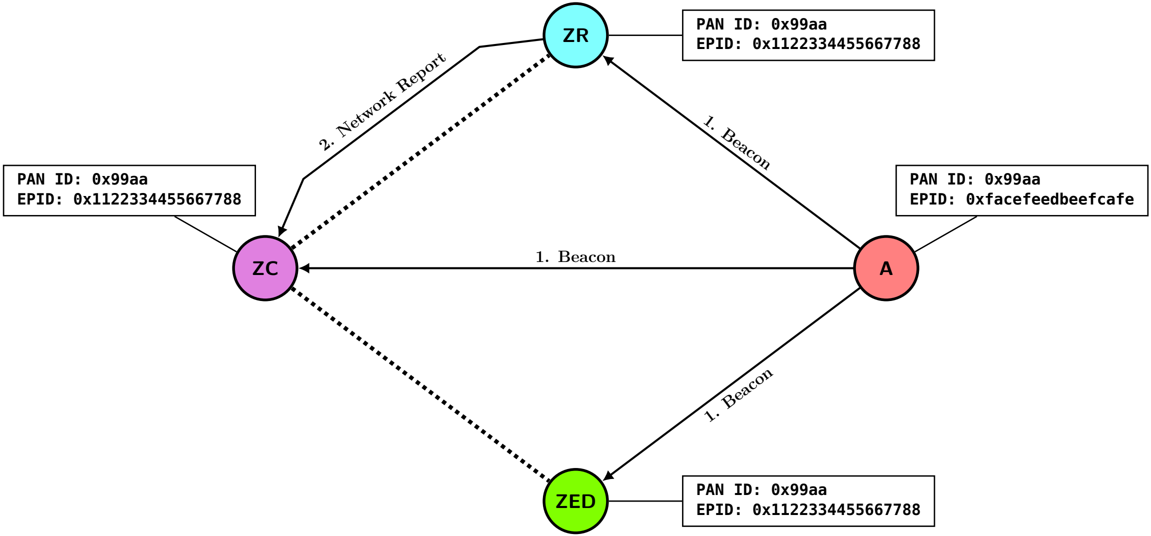
PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0x99aa
EPID: 0x1122334455667788

Disconnecting Zigbee Devices



Disconnecting Zigbee Devices



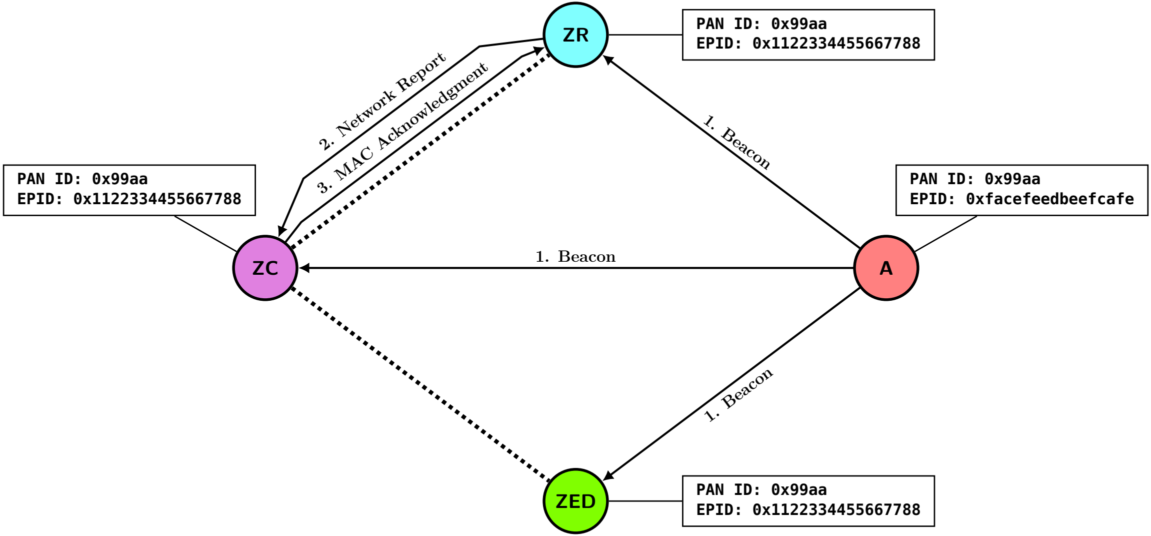
PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0x99aa
EPID: 0x1122334455667788

Disconnecting Zigbee Devices



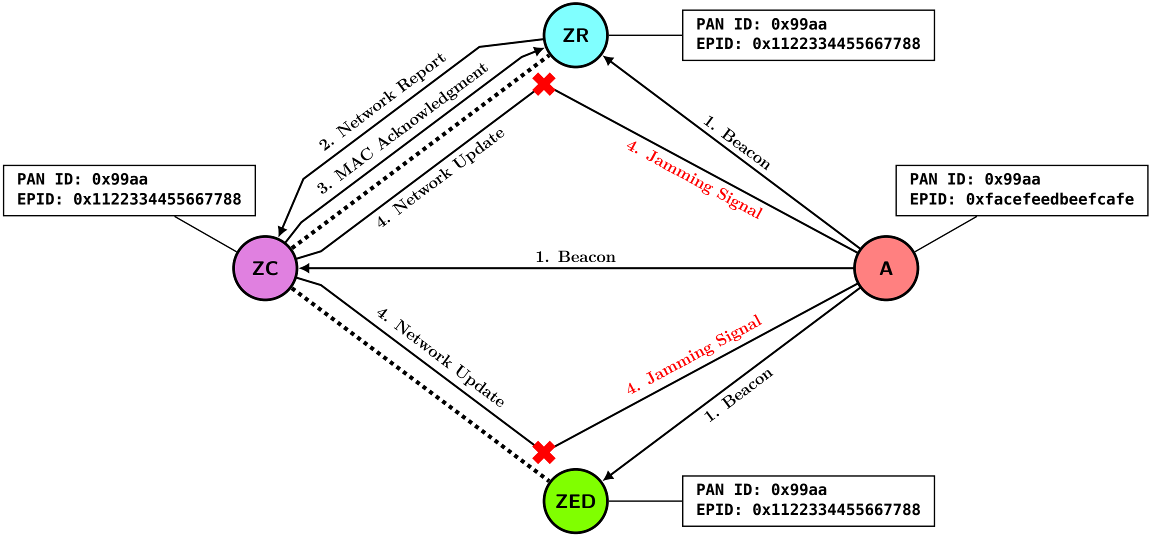
PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0x99aa
EPID: 0x1122334455667788

Disconnecting Zigbee Devices



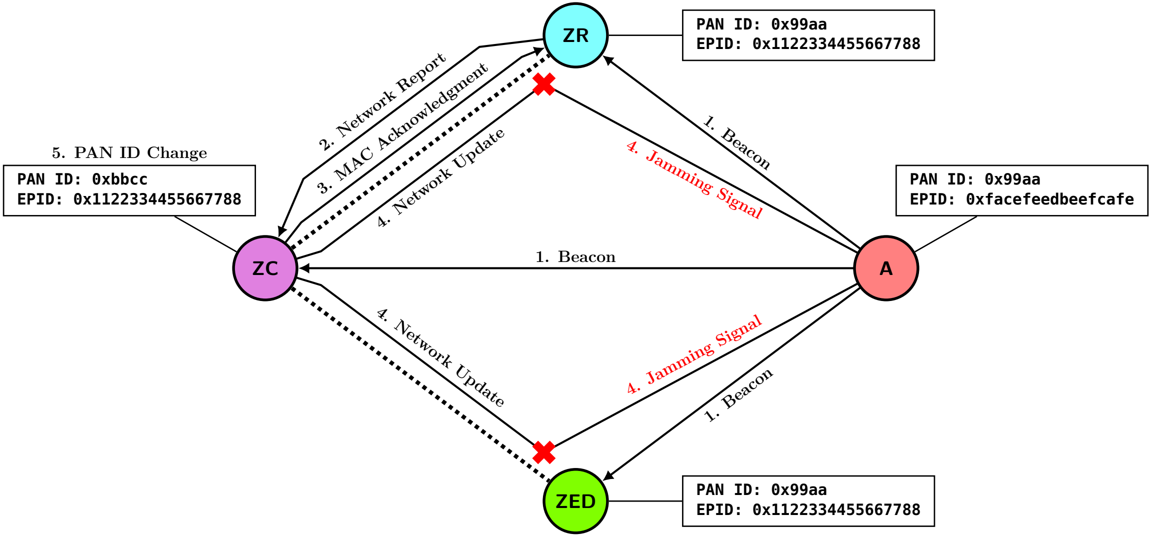
PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0x99aa
EPID: 0x1122334455667788

Disconnecting Zigbee Devices



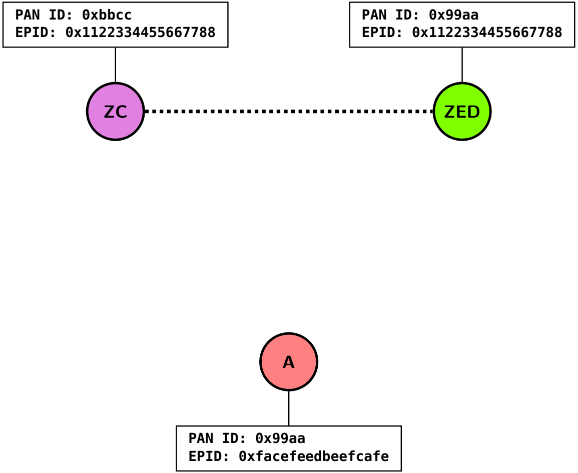
PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0x99aa
EPID: 0x1122334455667788

Keeping Zigbee Devices Disconnected (pt. 1)

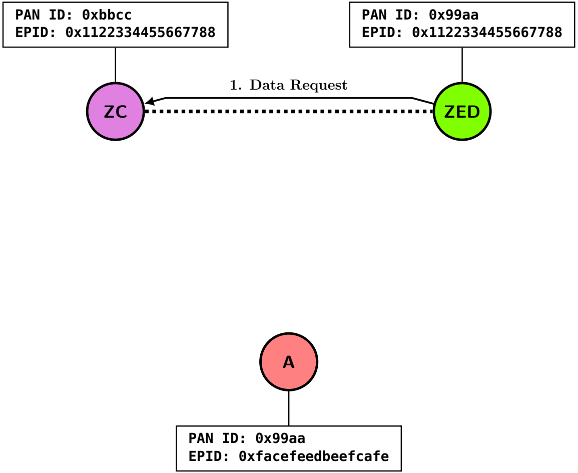


PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

Keeping Zigbee Devices Disconnected (pt. 1)

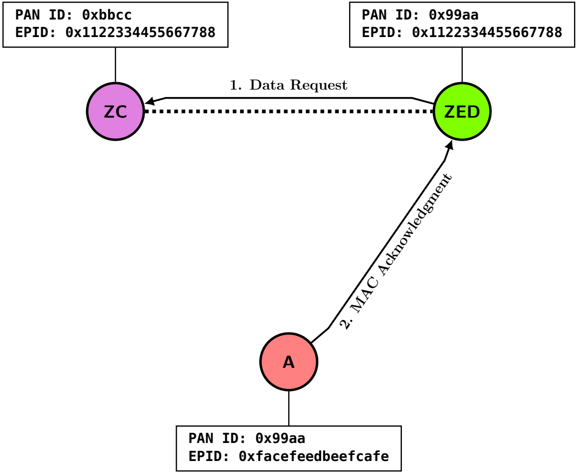


PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

Keeping Zigbee Devices Disconnected (pt. 1)

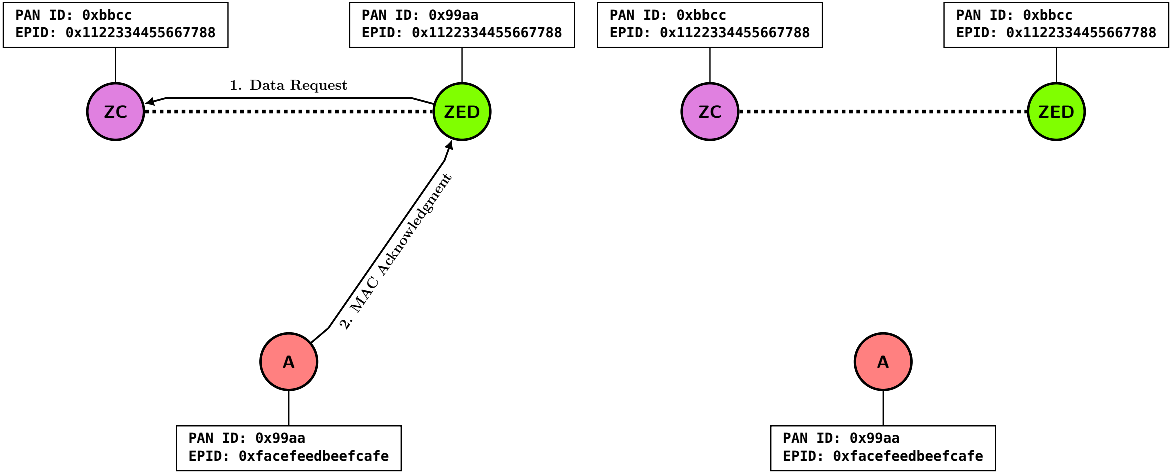


PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

Keeping Zigbee Devices Disconnected (pt. 1)



PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

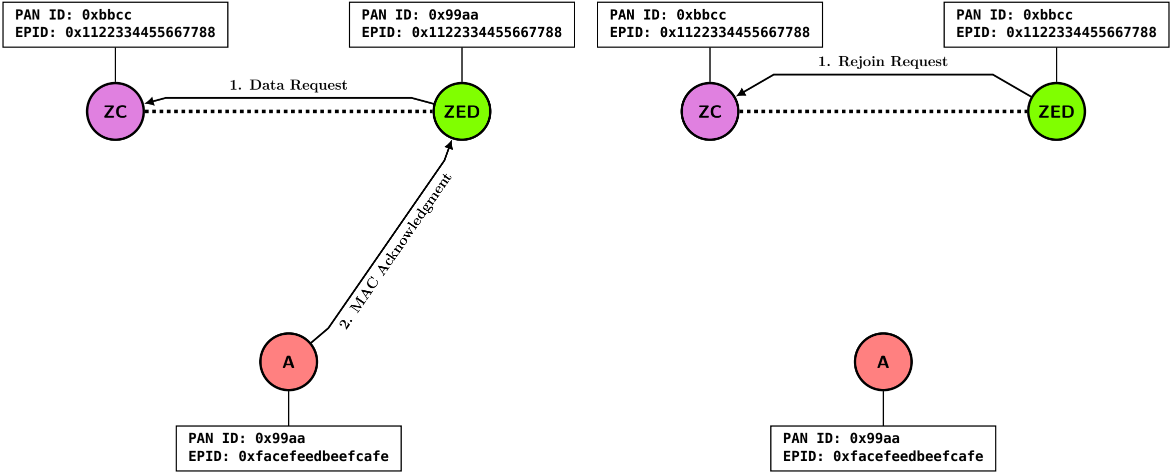
PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0xbbcc
EPID: 0x1122334455667788

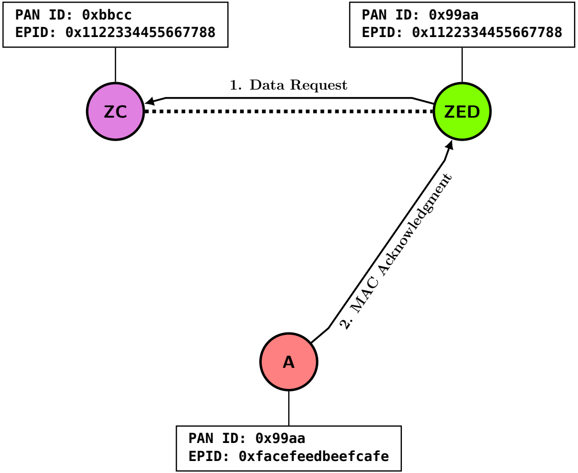
PAN ID: 0xbbcc
EPID: 0x1122334455667788

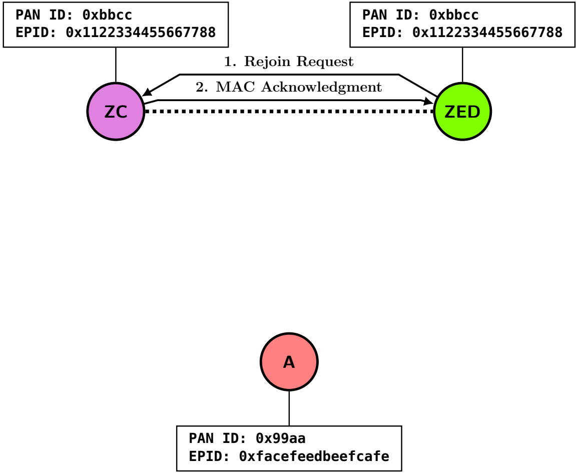
PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

Keeping Zigbee Devices Disconnected (pt. 1)

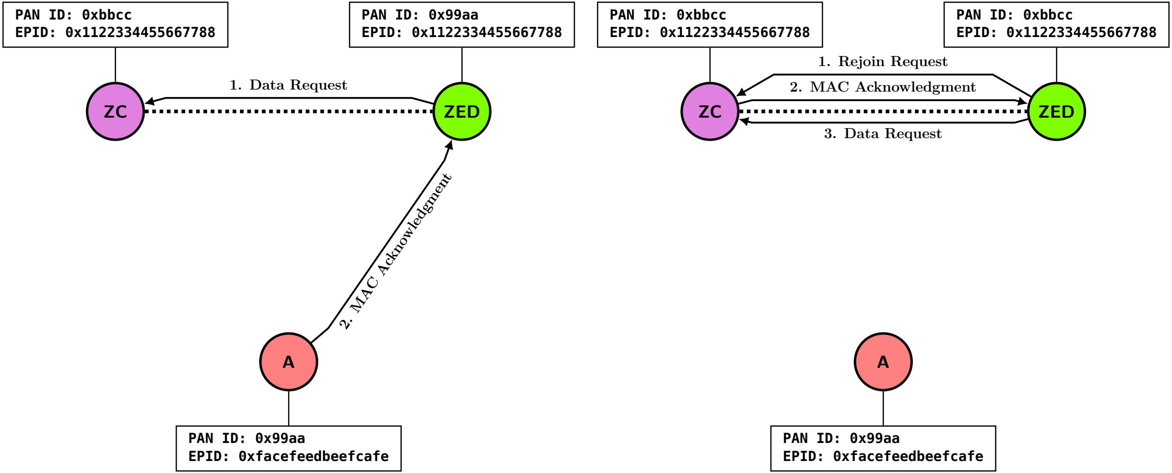


Keeping Zigbee Devices Disconnected (pt. 1)





Keeping Zigbee Devices Disconnected (pt. 1)



PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

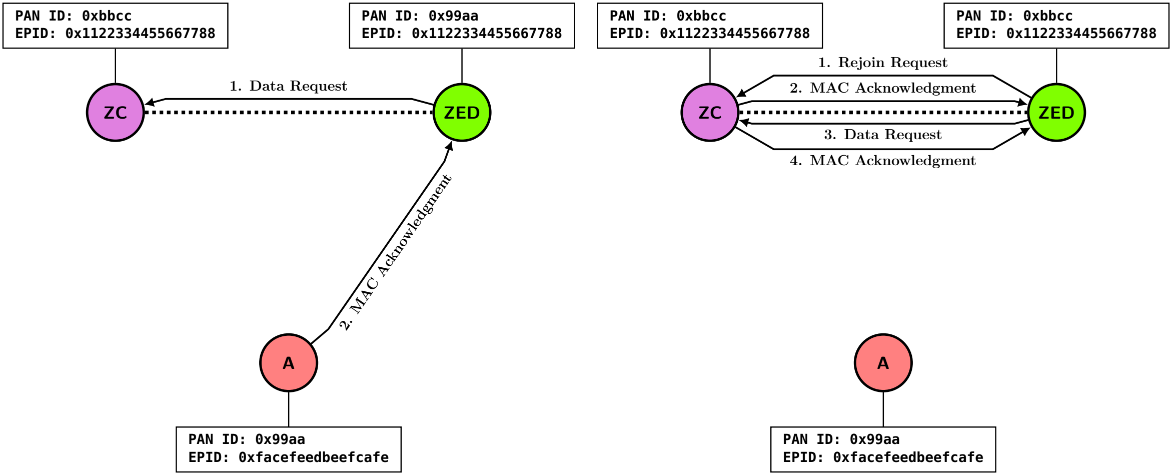
PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0xbbcc
EPID: 0x1122334455667788

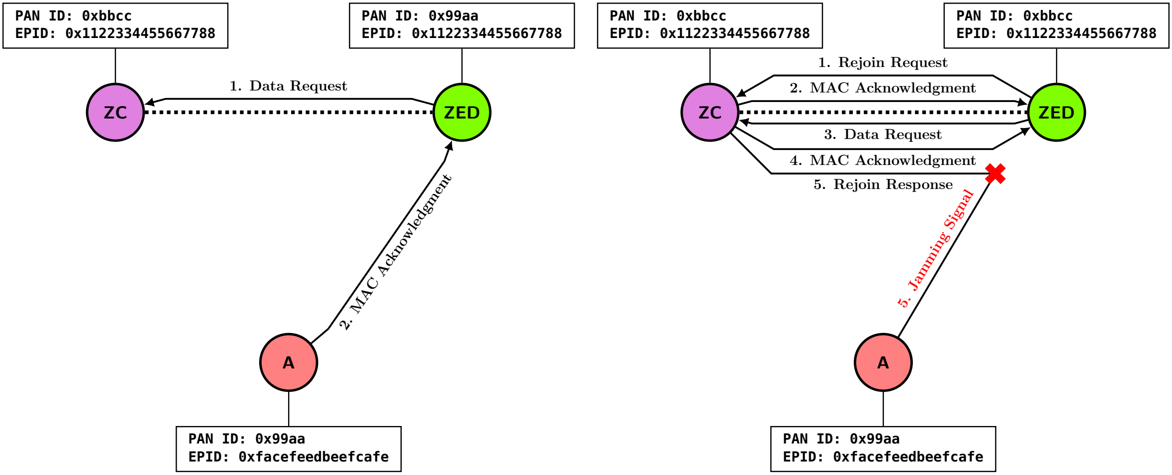
PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

Keeping Zigbee Devices Disconnected (pt. 1)



Keeping Zigbee Devices Disconnected (pt. 1)



PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

1. Data Request

2. MAC Acknowledgment

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0xbbcc
EPID: 0x1122334455667788

1. Rejoin Request

2. MAC Acknowledgment

3. Data Request

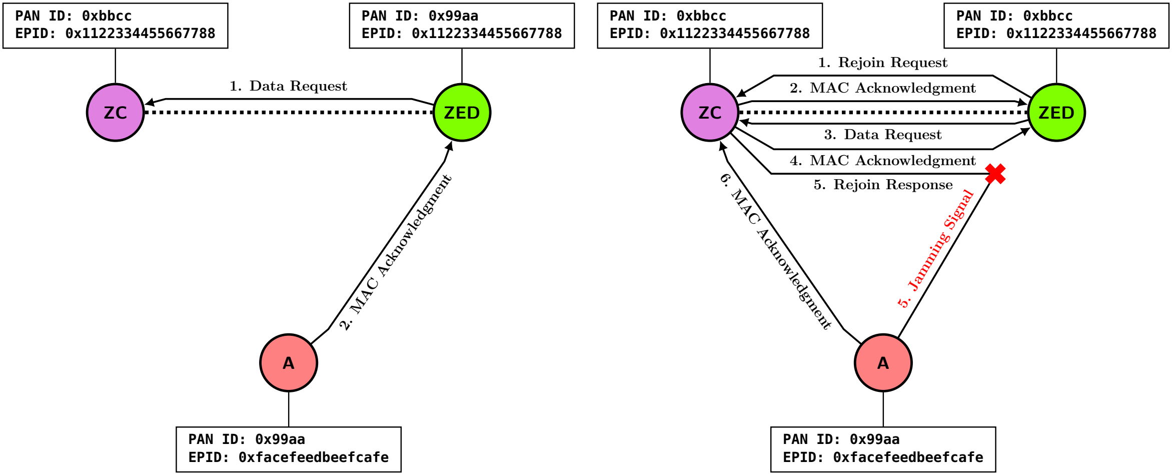
4. MAC Acknowledgment

5. Rejoin Response

5. Jamming Signal

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

Keeping Zigbee Devices Disconnected (pt. 1)



PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0x99aa
EPID: 0x1122334455667788

ZC

ZED

1. Data Request

2. MAC Acknowledgment

A

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

PAN ID: 0xbbcc
EPID: 0x1122334455667788

PAN ID: 0xbbcc
EPID: 0x1122334455667788

ZC

ZED

1. Rejoin Request

2. MAC Acknowledgment

3. Data Request

4. MAC Acknowledgment

5. Rejoin Response

5. Jamming Signal

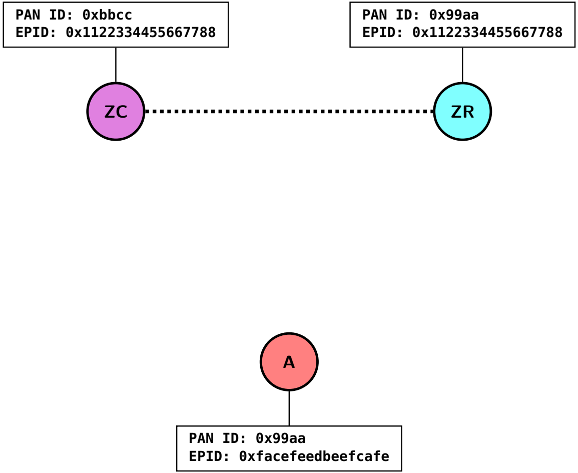
6. MAC Acknowledgment

A

PAN ID: 0x99aa
EPID: 0xfacefeedbeefcafe

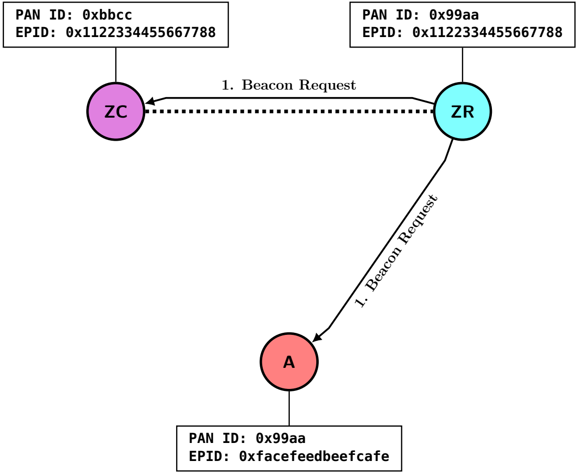
Keeping Zigbee Devices Disconnected (pt. 2)

- Some of our Zigbee devices were able to rejoin the network even if we jammed all **Rejoin Responses**
- By jamming the **beacons** with the updated PAN ID we could keep any Zigbee device disconnected



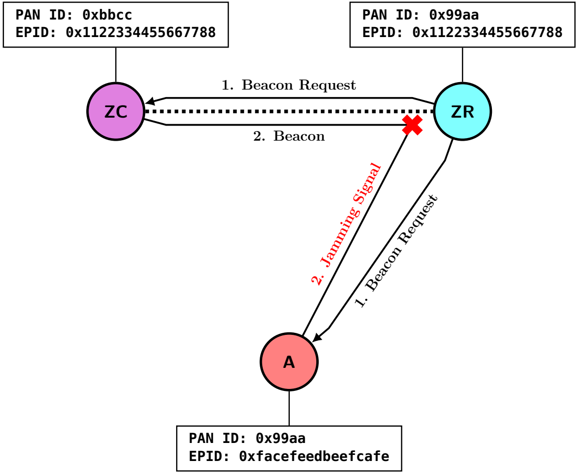
Keeping Zigbee Devices Disconnected (pt. 2)

- Some of our Zigbee devices were able to rejoin the network even if we jammed all **Rejoin Responses**
- By jamming the **beacons** with the updated PAN ID we could keep any Zigbee device disconnected



Keeping Zigbee Devices Disconnected (pt. 2)

- Some of our Zigbee devices were able to rejoin the network even if we jammed all **Rejoin Responses**
- By jamming the **beacons** with the updated PAN ID we could keep any Zigbee device disconnected



Responsible Disclosure

- Zigbee Routers may not initiate or significantly delay the **rejoin process** when they fail to receive the Network Update command:
  - Our SmartThings Smart Bulb did not initiate the rejoin process within 38 hours

Responsible Disclosure

- Zigbee Routers may not initiate or significantly delay the **rejoin process** when they fail to receive the Network Update command:
  - Our SmartThings Smart Bulb did not initiate the rejoin process within 38 hours
- We received the following comments from the **Zigbee Alliance**:
  - Specification changes will prevent malicious PAN ID changes
  - A more aggressive algorithm will be required to avoid missing PAN ID changes
  - It is difficult for the network key to be leaked from Zigbee 3.0 devices

Responsible Disclosure

- Zigbee Routers may not initiate or significantly delay the **rejoin process** when they fail to receive the Network Update command:
  - Our SmartThings Smart Bulb did not initiate the rejoin process within 38 hours
- We received the following comments from the **Zigbee Alliance**:
  - Specification changes will prevent malicious PAN ID changes
  - A more aggressive algorithm will be required to avoid missing PAN ID changes
  - It is difficult for the network key to be leaked from Zigbee 3.0 devices
- We recommend the following security enhancements:
  - The Trust Center link key should be **reconfigurable** over an out-of-band communication channel
  - The end users should be **made aware** of the security risks that the use of a legacy Zigbee device would introduce to their networks

Conclusion

- The lack of MAC-layer security exposes Zigbee networks to several passive and active attacks
- Developed software:
  - https://github.com/akestoridis/zigator
  - https://github.com/akestoridis/atusb-attacks
  - https://github.com/akestoridis/grc-ieee802154
  - https://github.com/akestoridis/wireshark-zigbee-profile
- CRAWDAD dataset cmu/zigbee-smarthome:
  - https://doi.org/10.15783/c7-nvc6-4q28
- Additional resources:
  - http://mews.sv.cmu.edu/research/zigator/