Evaluating the Security Vulnerabilities of ZigBee Motion Detectors in Wireless Sensor Networks

A Study on Jamming, Interception, and Behavioral Manipulation

SNE/OS3.nl - Research Presentation

Diogo Marques
diogo.marque@os3.nl
University of Amsterdam

Henk Netten
henk.netten@os3.nl
University of Amsterdam

Muhammad Mansour
muhammad.mansour@os3.nl
University of Amsterdam

Introduction

The rise of IoT devices, especially those using the ZigBee protocol, has brought about greater convenience in automation but also heightened security risks.

Many sellers have been advertising their human presence sensors as effective tools for intruder detection, often selling these devices in bundles with alarms and security hubs for global markets.

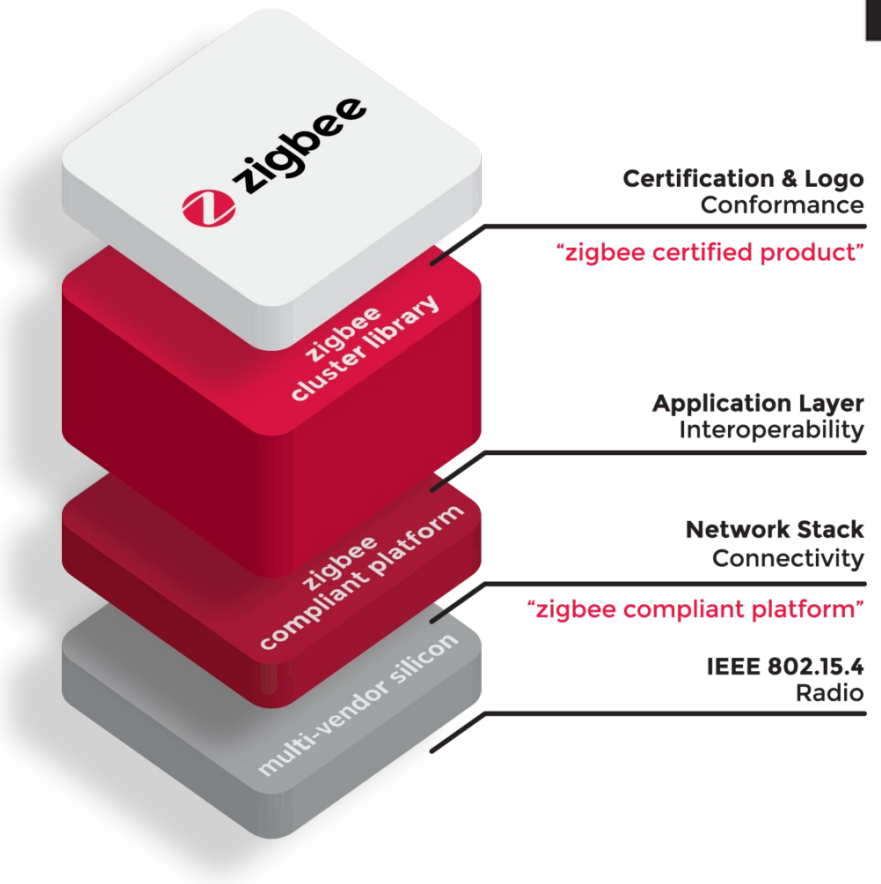
Research Questions

How can ZigBee-based motion detectors be utilized safely in high-security applications, such as break-in detection?
What are the vulnerabilities in the ZigBee-based IoT network?
How can we defend against attacks that exploit these vulnerabilities?

What is Zigbee?

A wireless communication protocol for low-power, low-data-rate applications.

Ideal for Internet of Things (IoT) devices, smart homes, and industrial automation.



What are its features?

Low Power Consumption: Designed for long battery life in devices like sensors.

Short Range: Operates at distances of 10 to 100 meters.

Low Data Rate: Supports rates from 20 kbps to 250 kbps.

Mesh Networking: Enhances reliability and extends communication range.

Robust Security: Implements encryption and authentication for data protection.

- **Encryption**: Uses AES-128 encryption to secure data transmission, preventing unauthorized access and ensuring data confidentiality.
- **Authentication**: Employs mechanisms such as key establishment protocols to authenticate devices before joining the network, protecting against unauthorized device access.
- **Integrity Checks**: Includes message integrity checks to ensure data has not been tampered with during transmission.

Interoperability: Based on IEEE 802.15.4 standard for compatibility with various devices.

Environment - Hardware

Wireless router

Silvercrest Controller

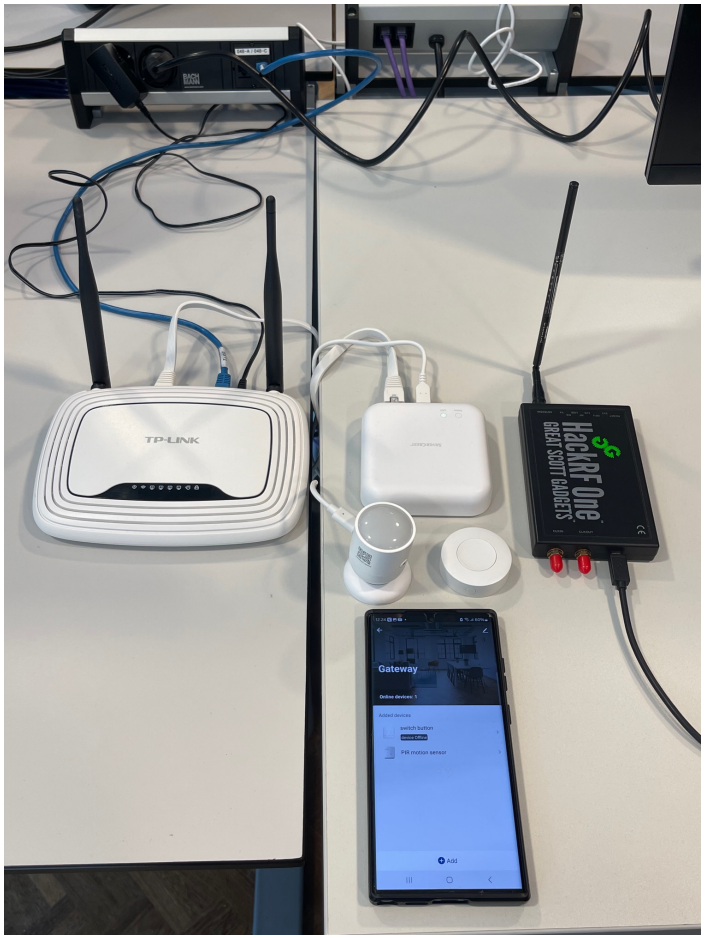
Sonoff Zigbee Human Presence Sensor

Sonoff Zigbee Wireless Switch

HackRF One

Smartphone

Linux laptop



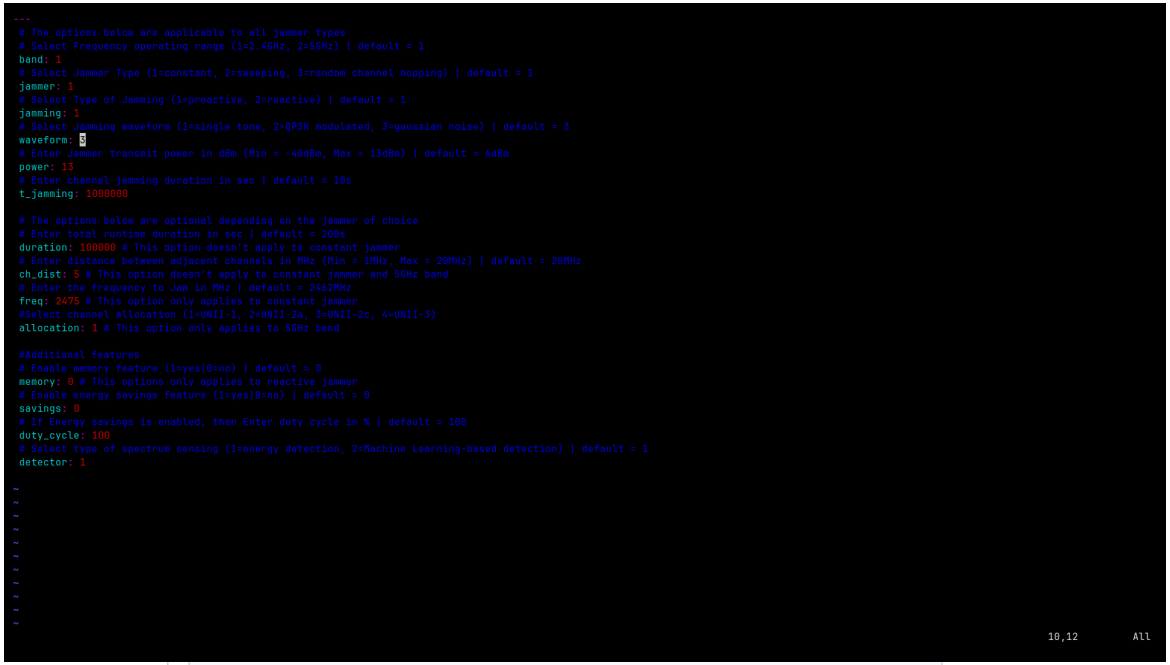
Environment - Software

Zigbee App (Tuya Smart)

GnuRadio Companion

Wireshark

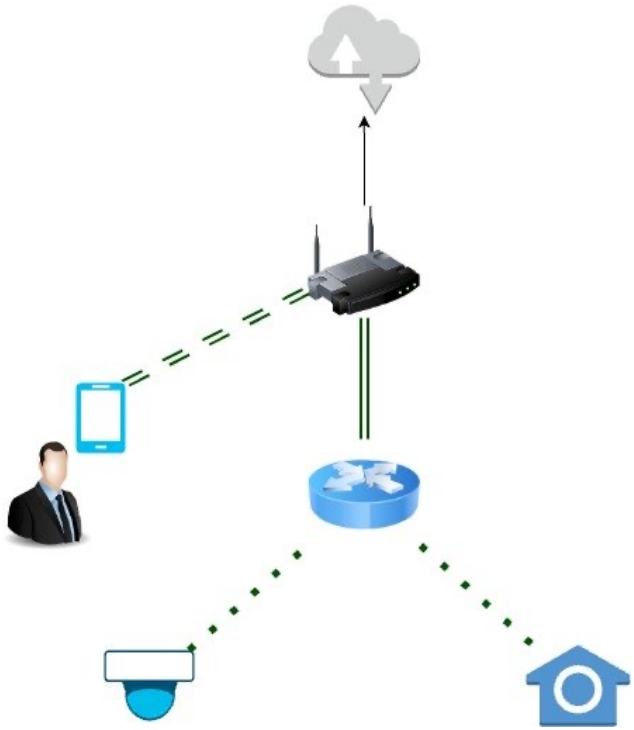
JamRF



Environment - Zigbee

We connected the zigbee controller to the internet router.

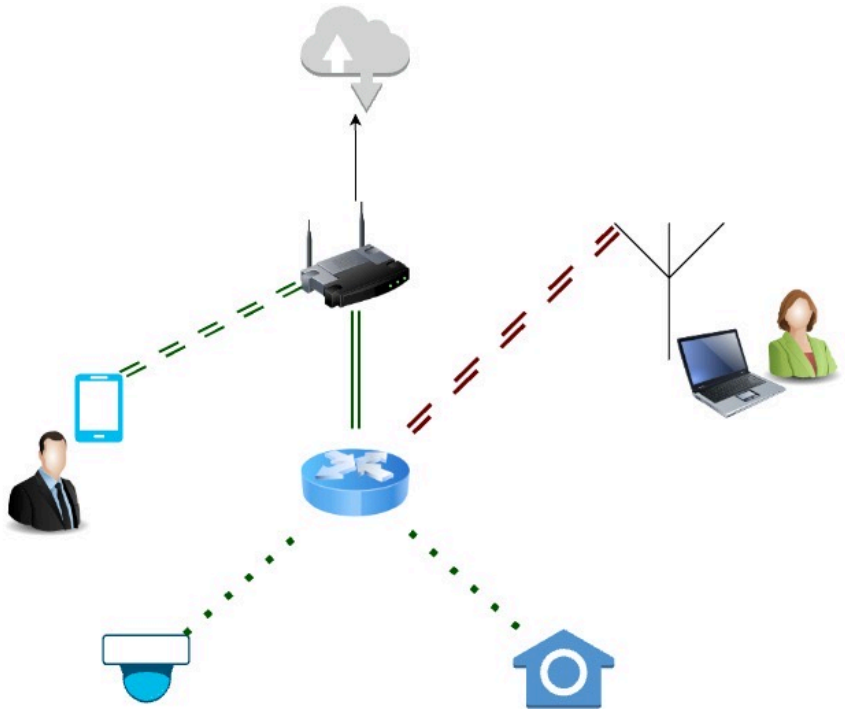
Using the smartphone application Tuya we paired the zigbee controller and both sensors.



Intruder

Scans the 2.4GHz network to retrieve information using HackRF and Gnuradio (RFTap).

Wireshark can the decrypt the packets with the Global Key but gives us limited information.

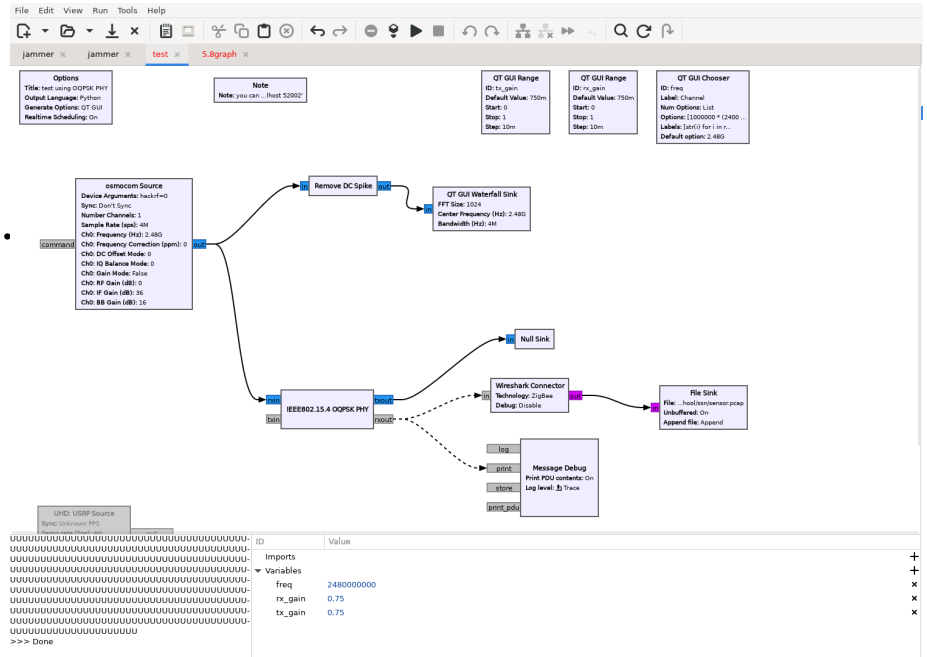


Gnu Radio

Required to control the HackRF hardware.

Prerequisites were to build 3 projects (.grc) from gr-foo and gr-ieee802-15-4.

Input source is connected to an Remove DC Spike and OQPSK Phy block to transform the signals and using the wireshark connector its saved to a .pcap file.



Wireshark

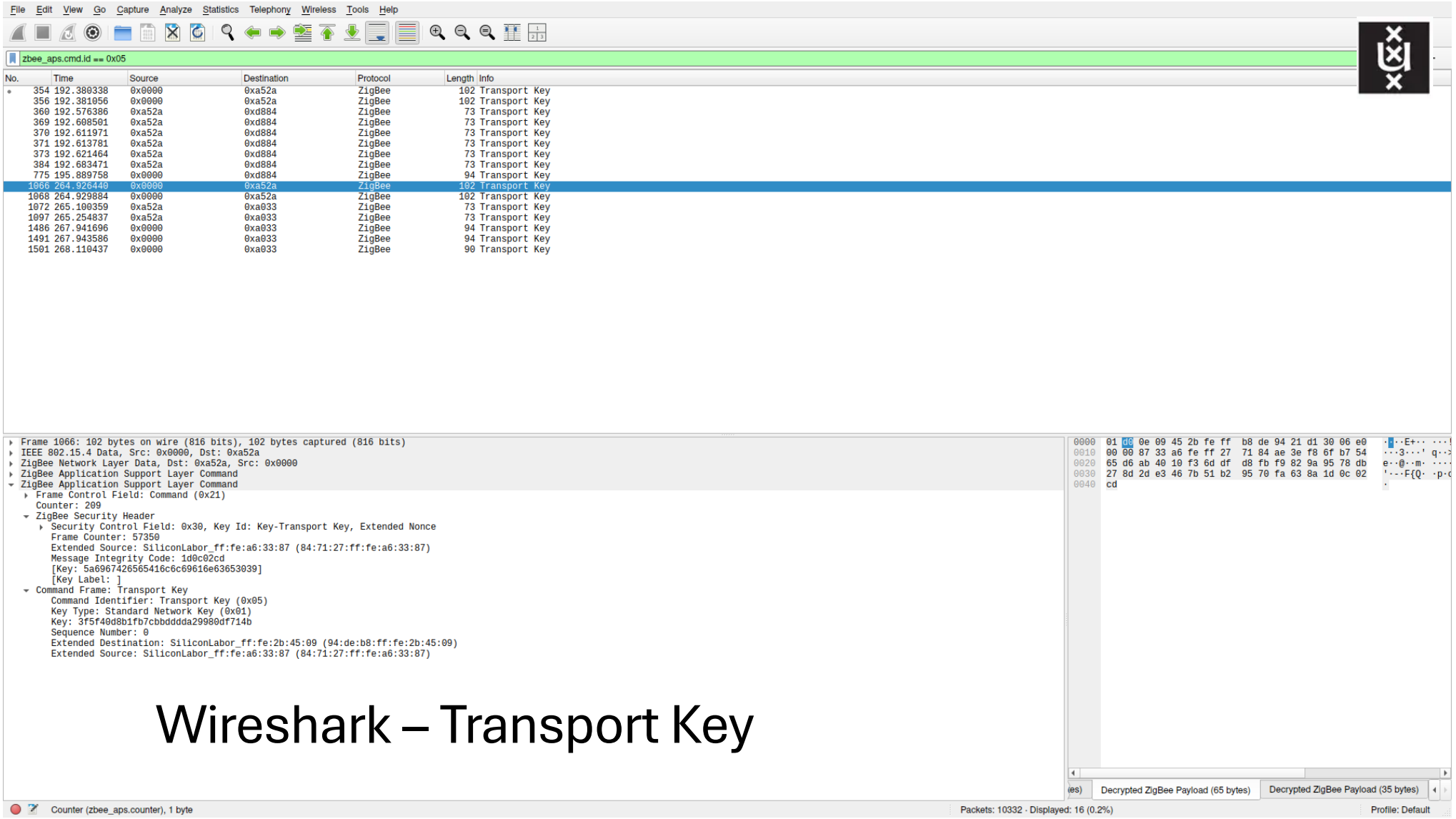
Required to read the HackRF output.

Using the Zigbee Global Key is possible to decrypt some of the messages.

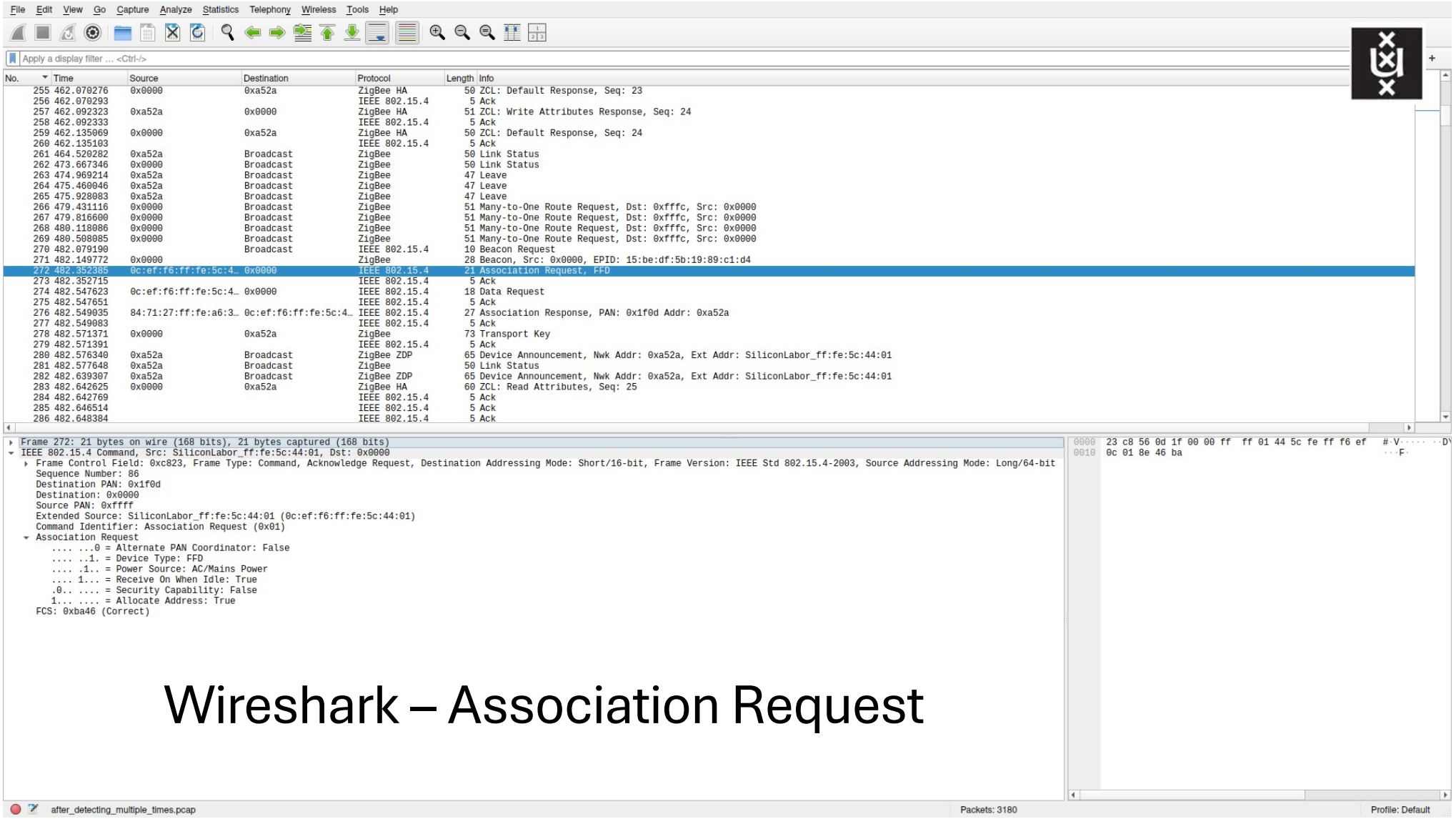
A second key is required to have the full information.

Relevant Zigbee Packages

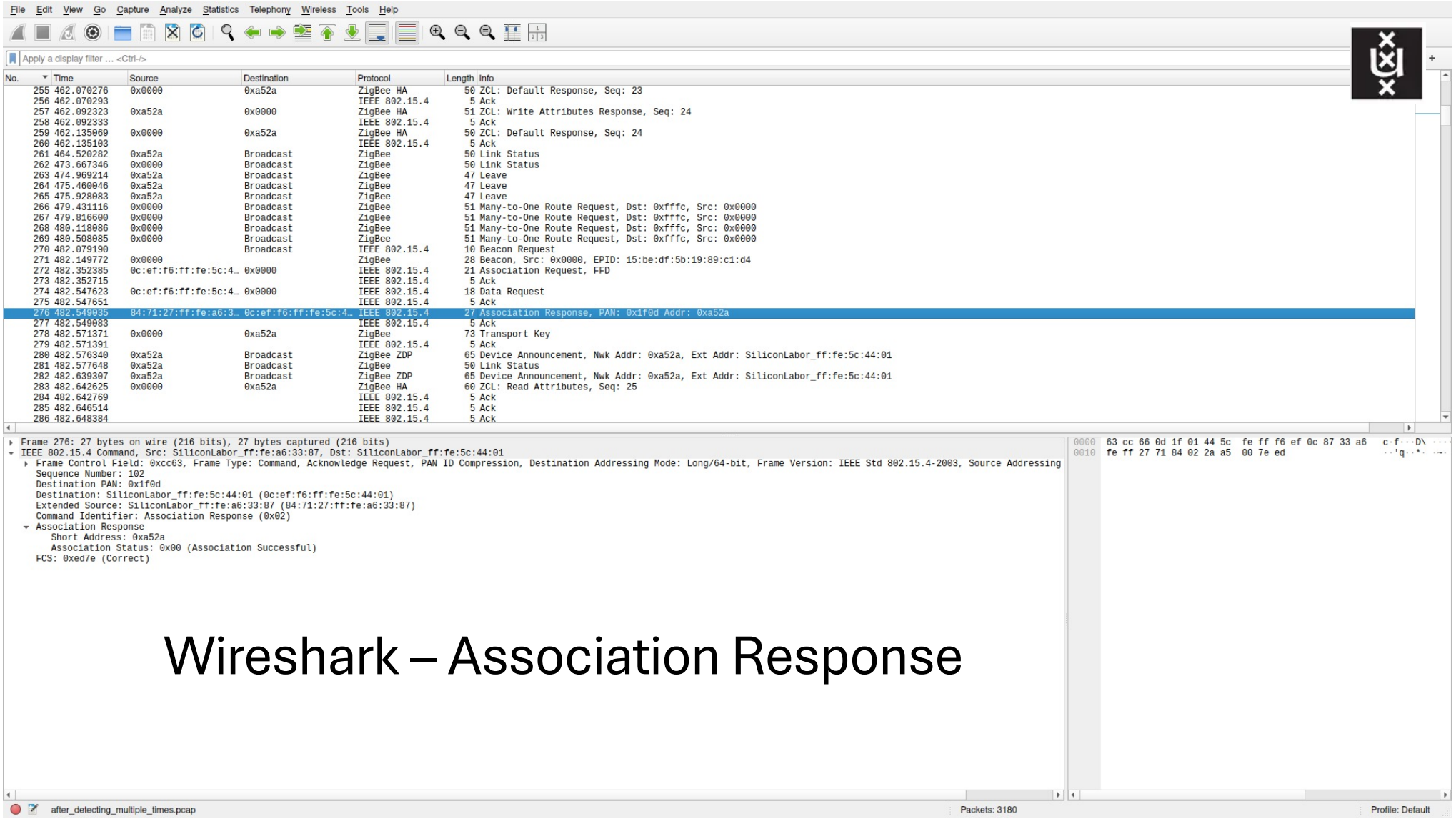
- Transport Key
- Association Request
- Association Response
- Device Announcement
- Status Zone Change



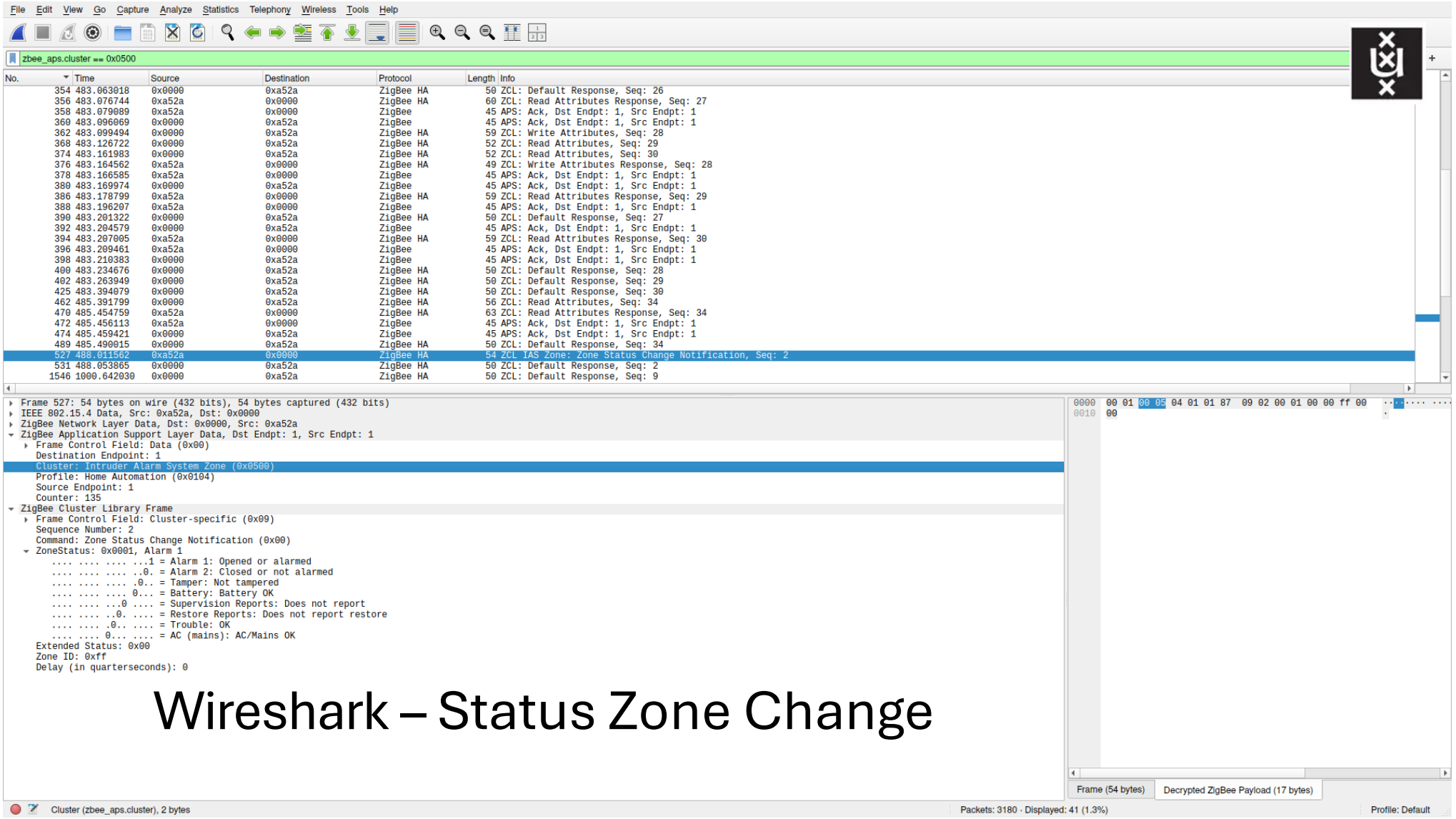
Wireshark – Transport Key



Wireshark – Association Request



Wireshark – Association Response



Wireshark – Status Zone Change

Experiments - Overview

Intercepting communications between the controller and the sensor.

Introduce a new/intruder device on the network.

Jamming the human presence sensor.

Jamming the communications between the sensor and the hub.

Results - Overview

We verified during the experiments the human presence detection capabilities were not affected with HackRF when jamming signal.

- Aluminum foil covering its surface did the job.

Disable the Zigbee network communications.

- The app shown no notifications when the sensor has detected an intruder.
- After the intruder left the area and the service was back online no incidents were reported.

Results - Intercepting

We got the key by listening in on the handshake.

Triggering the handshake is difficult. We tried to pair another device with the coordinator to force it to make a new handshake. This required access to the mobile phone app.

Furthermore, registering a new mobile phone app required pressing a button on the cooridnator.

Results - Jamming

Communication - With jamRF and RFJamming-FMRadio-SDR works 2.4GHz band (between coordinator and zigbee device)

The switch and human presence sensor both stopped communicating with the hub.

Jamming 5.8GHz to interfere with the sensor's detection was not successful.

Detection - We used gaussian noise and impulse noise in gnuradio.

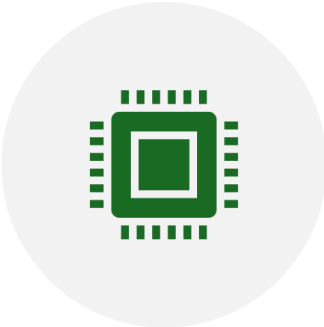
The sensor still detected human presence, and the status changed on the app.

We tried to see if only the first detection would work but sadly future detections also worked. (The continuous noise wave doesn't make a status for the sensor.)

If this did work, an attacker could place a jamming device and come back later to do crimes.

Conclusions

THE SENSOR IS PROVEN EFFICIENT WHEN DETECTING INTRUDERS.



WE COULDN’T TEST HIS COMMUNICATIONS DURING THE JAMMING BECAUSE WE HAD HARDWARE LIMITATIONS SO WE CAN'T CONFIRM THE SENSOR SENT THE CHANGE STATUS PACKAGE TO THE CONTROLLER.

WHEN JAMMING THE 2.4GHZ COMMUNICATION HUB IT WAS POSSIBLE TO NOT NOTIFY THE OWNER THAT AN INTRUDER WAS IN THEIR PROPERTY AND BECAUSE THERE IS NO INTERNAL MEMORY NO NOTIFICATION WAS SAVED IN HISTORY ON THE APP.

Thank you!

Questions?

Diogo Marques
diogo.marque@os3.nl
University of Amsterdam

Henk Netten
henk.netten@os3.nl
University of Amsterdam

Muhammad Mansour
muhammad.mansour@os3.nl
University of Amsterdam