Security Analysis of Zigbee

Xueqi Fan, Fransisca Susan, William Long, Shangyan Li

{*xueqifan, fsusan, wlong, shangyan*}*@mit.edu*

May 18, 2017

Abstract

This paper analyzes the security of Zigbee - a wireless communication protocol for Internet-of-Things devices. We start with the components in a network using Zigbee standard. We then give the readers an overview of the security policy, measures, and architecture. After the series of introductions of the standard, we discuss the devices and methods used to find security vulnerabilities and corresponding results. Lastly, we present a set of recommendations to Zigbee standard that will likely improve their security.

1 Introduction

Internet of Things (IoT) has become increasingly popular in the past few years. Subsequently, the security of the IoT devices becomes crucial, especially many devices have access to highly personalized and sensitive data. Zigbee is one of the most widely used standards for wireless communication between different IoT devices and has been adopted by many major companies, like Samsung and Philips. Zigbee is an open standard for low-power, low-cost wireless personal area networks that interconnect devices primarily for personal uses. The standard aims to provide a two-way and reliable communication protocol for applications with a short range, typically 10-100 meters. Zigbee is implemented with different application standards used in a variety of application areas, including home automation, smart energy, remote control and health care.

Even though Zigbee was designed with the importance of security in mind, there have been trade-offs made to keep the devices low-cost, low-energy and highly compatible. Some parts of the standard's security controls are poorly implemented, which inevitably lead to security risks. This paper highlights the main security risks and results of attempted attacks on a few IoT devices implemented with Zigbee standard.

2 Responsible Disclosure

In order to perform security analysis on Zigbee protocol, we purchased the Samsung SmartThings Hub v2, the Smart Outlet, and the Iris Contact Sensor. According to the Digital Millennium Copyright Act (DMCA) security research exemption for consumer devices, which was in effect since October 28, 2016, and lasts for two years, we are legally conducting this security analysis of Zigbee protocol by testing on these purchased Zigbee devices[1].

In more details, the exemption "authorized security researchers who are acting in good faith to conduct controlled research on consumer devices so long as the research does not violate other laws."[1] Our project satisfies this description because first, we have only been using open sourced programs as tools to test Zigbee devices and the devices are legally acquired. Then, we are performing the analysis and "hacking" with good-faith since we aim to examine the vulnerabilities of Zigbee protocol

as a final project for 6.857. This paper will also be published on 6.857 course website as additional evidence for "good-faith." Lastly, the Zigbee devices we chose are included in the exemption because they are designed for use by individual consumers, instead of industry.

3 Security Policy

3.1 Principals

First, we introduce the five principals in Zigbee's security policy. A graph is included to illustrate the technical components of a Zigbee network.

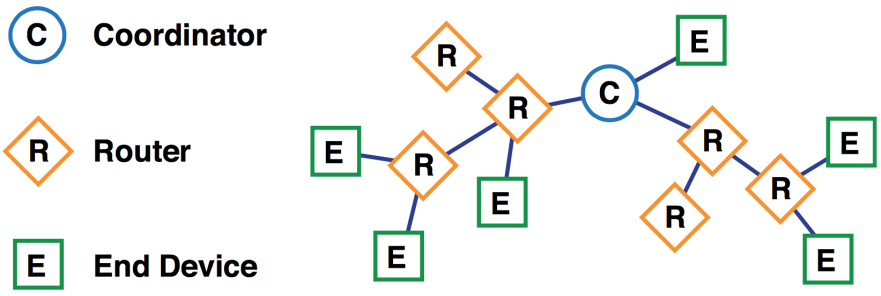

Figure 1: Zigbee Overview

3.1.1 Owner

The owner of Zigbee devices purchase the devices and need to establish the network with the coordinator and add other routers and end devices to the network. The owner can also remotely control the devices.

3.1.2 Other Users

Other users in the household are also a principal in the policy. They can remotely control the devices and might be able to control the network by the permission of the owner.

3.1.3 Coordinator

Each Zigbee network must have one coordinator that manages the overall network[2]. A coordinator usually functions as the trust center that provides security control of the network. The coordinator is responsible for establishing the network. In that process, it chooses the channel that is used in the network for the devices to communicate. Then the coordinator gives permission to other devices to join or leave the network and keeps track of all the end devices and routers. Also, it configures devices and enables end-to-end security between devices. More importantly, the coordinator stores and distributes the network keys. In a Zigbee network, the coordinator cannot sleep and needs to be continuously powered[3].

3.1.4 Router

Routers in a Zigbee network act as intermediate nodes between the coordinator and the end devices. Routers have to join the network first by the permission of the coordinator. Then they can route

traffic between end devices and the coordinator, as well as transmit and receive data. A router also able to allow other routers and end devices to join the network. Similar to the coordinator, routers also cannot sleep as long as the network is established[3].

3.1.5 End Device

A Zigbee end device is the simplest type of device on a Zigbee network, and it is often low-power or battery-power. End devices are what the customers are more familiar with, like motion sensors, contact sensors, and smart light bulbs. The end devices also must join the network first to communicate with other devices. However, unlike the coordinator and the routers, the end devices do not route any traffic and cannot allow other devices to join the network. As a result of the inability to relay messages from other devices, the end devices can only communicate within the network through their parent nodes, often routers. Also different from the other two types of devices, the end devices can enter low power mode and sleep to conserve power[2]. This feature makes battery power possible for end devices.

4 Security Measures

Zigbee claims to provide state-of-the-art security tools allowing its member companies to create some of the most secure IOT wireless devices. Its security is based on symmetric-key cryptography, in which two parties must share the same keys to communicate. Zigbee uses the highly secure 128-bit AES-based encryption system [13]. Zigbee protocol is built on the IEEE 802.15.4 wireless standard, which has two layers, the physical layer (PHY) and the medium access control layer (MAC). Zigbee builds the network layer (NWK) and the application layer (APL) on top of PHY and MAC. As a low-cost protocol, Zigbee assumes an 'open trust' model where the protocol stack layers trust each other. Hence, cryptographic protection only exists between devices, but not between different layers in a device. This allows keys reusing among layers of the same device. For simplicity of the interoperability of devices, Zigbee uses the same security level for all devices on a given network and all layers of a device. Furthermore, it establishes the principle 'the layer that originates a frame is responsible for initially securing it'[4].

In addition, Zigbee command includes a frame counter to stop replay attacks (in which an attacker could record and replay a command message). The receiving endpoint always checks the frame counter and ignores duplicate messages.

Zigbee also supports frequency agility, in which its network is relocated in case of a jamming attack. [6]

4.1 Security Model

To satisfy a wide range of applications while maintaining low cost and power, Zigbee claims to offer two network architectures and corresponding security models: distributed and centralized. They differ in how they admit new devices into the network and how they protect messages on the network. [6]

A distributed security model provides a less-secured and simpler system. It has two devices types: routers and end devices. Here, a router can form a distributed security network when it can't find any existing network. Each router can issue network keys. As more routers and devices join the network, the previous routers on the network send the key. To participate in distributed security networks, all router and end devices must be pre-configured with a link key that is used to encrypt the network key when passing it from a router parent to a newly joined node. All the devices in the network encrypt messages with the same network key.

A centralized security model provides higher security. It is also more complicated as it includes a third device type, the Trust Center (TC), which is usually also the network coordinator. The *Trust Center* forms a centralized network, configures and authenticates routers and devices to join

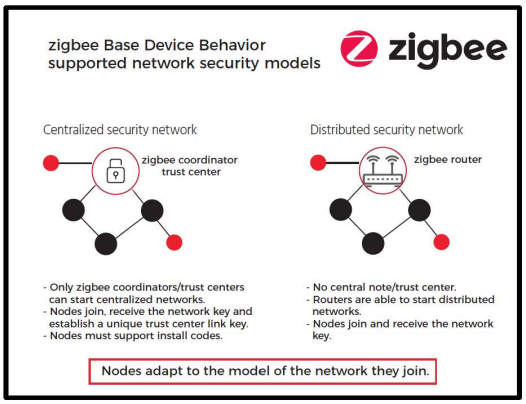

Figure 2: Centralized vs. Distributed Zigbee Network

a network. The TC establishes a unique TC Link Key for each device on the network as they join and link keys for each pair of devices as requested. The TC also determines the network key. To participate in a centralized security network model, all entities must be pre-configured with a link key that is used to encrypt the network key when passing it from the TC to a newly joined entity. Both systems are illustrated in Figure 2 [13].

4.2 Security Assumptions

Aside from the open trust model between layers, the security of Zigbee ultimately depends on the following assumptions [4]:

1. The safekeeping of symmetric keys. Zigbee assumes that secret keys are not available outside of the device in an unsecured way, meaning that all transmission of keys must be encrypted. An exception to this is during pre-configuration of a new device, in which a single key might be sent unprotected, creating a brief vulnerability. Here, if the keys are stolen because the adversary has physical access to the devices, many information then become available. Zigbee's security policy does not protect against attack to hardware due to its low-cost nature.

2. The protection of mechanism employed. All Router and End Device nodes should support both centralized security and distributed security by adapting to the security scheme employed by the network that they join [14].

3. The proper implementation of cryptographic mechanism and associated security policies involved. Here, Zigbee developers are assumed to follow the complete protocol in practice. Zigbee also assumes the availability of almost perfect random number generators.

4.3 Security Keys

Zigbee network and devices use a *network* key and *link* keys to communicate. The recipient party always knows which keys are used in protecting the messages.

A network key is a 128-bit key shared by all devices in the network, which is used for broadcasting communications. There are two types of network keys: standard and high-security. The type usually controls how a network key is distributed as the network key must itself be protected by encryption when it is passed to the joining node [13]. For this encryption, a pre-configured link key is used; this key is known by both the Trust Center and the joining device for centralized security; this key is known by all nodes in distributed security.

A link key is a 128-bit key shared by two devices. There are two types of link keys: global and unique. The type determines how the device handles various TC messages (APS commands). In a centralized security network, there are three kinds of link keys: 1) global link key used by the TC and all nodes in the network, 2) unique link key used for a one-to-one relation between TC and a node, later replaced by the Trust Center link key, and 3) application link key, that is used between a pair of devices. Here, link keys related with the TC are usually pre-configured using an out-of-band method, for instance, QR code in the packaging, while link keys between entities are often generated by the Trust Center and encrypted with the network key. In a distributed security network, link keys only exist between a pair of devices.

4.3.1 Security Key Types

Centralized Security Model
In a centralized security network, the keys for the network layer are as follows:

- **Network key**, as detailed above.
- **Pre-configured global link key**, which is used to encrypt the network key when it is passed from the TC to the devices. This link key is the same for all nodes in the network. [13] It may be Zigbee-defined key or manufacturer-defined:
  - The Zigbee-defined key, `5A 69 67 42 65 65 41 6C 6C 69 61 6E 63 65 30 39` (`ZigbeeAlliance09`), which allows nodes from different manufacturers to join the network.
  - A manufacturer-defined key that only allows nodes from the specific manufacturer to join the network.
- **Pre-configured unique link key**, which is also used to encrypt the network key when sent from the TC to a node. This link key is exclusive for each (TC, node) pair so it is different for every node. This link key is usually pre-configured or pre-programmed into the relevant nodes either in the factory or during commissioning [13]. In the new version, Zigbee 3.0, the pre-configured unique link key is usually in the form of an install code, a random 128-bit number protected by a 16-bit CRC (cyclic redundancy check) pre-installed in the devices. [6]

In an older version of Zigbee protocol, the nodes usually use the Zigbee defined pre-configured global link key but most devices compatible with Zigbee 3.0 use the pre-configured unique link key or manufacturer defined pre-configured global link key.

Once network-level security is set up, application-level security can be set up for more secure communication. The keys for the application layer are as follows:

- **Pre-configured global link key**, as explained above. This key is used for communication between the TC and all other nodes.
- **Pre-configured unique link key**, as explained above. This key is used for communication between the TC and one other node.
- **Trust Center Link Key (TCLK)**, which is used between the TC and one other node. This 128-bit key is derived from the pre-configured unique link key using Matyas-Meyer-Oseas (MMO) hash function or randomly generated by the TC. [6,13] This key is passed from the TC to the relevant node with encryption using the network key and (if exists) the pre-configured unique link key for the node. This Trust Center Link Key then is used to encrypt all subsequent communication between the TC and the relevant node, replacing the pre-configured unique link key. However, the node still keeps the pre-configured link key in case it needs to rejoin in the future.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.