STIR: A Smart and Trustworthy IoT System Interconnecting Legacy IR Devices

Zhen Ling, Chao Gao, Chuta Sano, Chukpozohn Toe, Zupei Li, and Xinwen Fu

***Abstract*—Legacy-infrared (IR) devices are pervasively used. They are often controlled by IR remotes and cannot be controlled over the Internet. A trustworthy and cost-effective smart IR system that is able to change an IR controllable device into a smart Internet of Things (IoT) device and interconnect them for smart city/home applications is offered in this article. First, a printed circuit board (PCB) consisting of an IR receiver and multiple IR transmitters side by side which are capable of transmitting about 20 m indoors is designed and implemented. This IR transceiver board is the first of its kind. Second, the IR transceiver can be linked up with a Raspberry Pi, for which we develop two software tools, recording and replaying any IR signals so as to put the corresponding IR device in control. Third, a smartphone can be connected to the Pi by means of a message queuing telemetry transport (MQTT) cloud server so that the commands can be sent by the smartphone to the legacy IR device over the Internet. We have also identified the deficiency of TLS mutual authentication implemented by the popular MQTT open-source package Mosquitto for a trustworthy IoT system and patched the system. We analyze the factors that affect the IR signal transmission distance, discuss the security concerns of our IR transceiver, and illustrate the scenarios for attacks. For instance, TV can be turned off remotely by a drone equipped with the transceiver.**

***Index Terms*—Drone, infrared (IR), Internet of Things (IoT), smart home.**

Manuscript received August 13, 2019; revised December 15, 2019; accepted December 27, 2019. Date of publication January 3, 2020; date of current version May 12, 2020. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB0803400, Grant 2018YFB2100300, and Grant 2017YFB1003000; in part by the National Science Foundation of China under Grant 61972088 and Grant 61532013; in part by the U.S. NSF under Grant 1461060, Grant 1642124, Grant 1547428, Grant 1915780, and Grant 1931871; in part by the Jiangsu Provincial Natural Science Foundation for Excellent Young Scholars under Grant BK20190060; in part by the Jiangsu Provincial Key Laboratory of Network and Information Security under Grant BM2003201; in part by the Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grant 93K-9; and in part by the Collaborative Innovation Center of Novel Software Technology and Industrialization. *(Corresponding author: Zhen Ling.)*

Zhen Ling is with the School of Computer Science and Engineering, Southeast University, Nanjing 210096, China (e-mail: zhenling@seu.edu.cn).

Chao Gao, Chukpozohn Toe, and Zupei Li are with the Department of Computer Science, University of Massachusetts Lowell, MA 01854 USA (e-mail: cgao@cs.uml.edu; zli1@cs.uml.edu; ctoe@cs.uml.edu).

Chuta Sano is with the Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: csano@cmu.edu).

Xinwen Fu is with the Department of Computer Science, University of Central Florida, Orlando, FL 32816 USA, and also with the Department of Computer Science, University of Massachusetts Lowell, MA 01854 USA (e-mail: xinwenfu@ucf.edu).

Digital Object Identifier 10.1109/JIOT.2019.2963767

I. Introduction

THE Internet of Things (IoT) is a world-wide network of uniquely addressable and interconnected objects [1], [2]. IoT has broad applications, including smart home, smart city, smart grid, and smart transportation. However, there still exist a large number of legacy devices, such as infrared (IR) controllable devices, which cannot be controlled over the Internet.

The IR technology is used in many different fields, including scientific research, business, and military. The IR spectrum can be divided into five categories: 1) near IR; 2) short wavelength IR; 3) mid wavelength IR; 4) long wavelength IR; and 5) far IR [3]. These different categories give rise to various IR applications. For example, near IR can be used for electronic devices, such as remote controls and mobile phones. Short wavelength IR is mainly used for long-distance telecommunications. Long wavelength IR can be used for thermal imaging. Consumer IR devices containing TVs, fans, air conditioners, as well as toys often use near IR for wireless communication. They often contain IR receivers, which are governed by means of IR remotes.

A smart IR remote equipped with Internet-based functions is developed and expounded in this article. In this way, legacy IR devices can be controlled over the Internet and are not bound to the short range of IR communication. Particularly, a custom IR signal recording and replaying the circuit board which is capable of being connected to a Raspberry Pi [or similar low-cost computers including microcontrollers (MCUs)] is devised. A software IR recording tool for the Pi is designed to record IR signals transmitted from an IR remote of any legacy consumer IR device. Then, the software IR replay tool can replay the signals and control the legacy IR device. The Raspberry Pi can be connected to the Internet and a smartphone can be used to remotely control the legacy IR device through the Pi over the Internet. The smartphone and Pi can be interconnected with an IoT broker on the cloud such as Amazon EC2. The prototype is of low cost as we use a Raspberry Pi Zero W with the WiFi capability that costs only $10. The cost can be further reduced if an MCU is used in place of the Pi.

The major contributions of this article are listed as follows.

1) We propose a low-cost strategy to transform any IR remote controllable devices into smart IoT devices. To the best of our knowledge, this is the first work to fully address IR playback through both hardware and software with much better performance compared with the Linux IR tool Linux IR remote control (LIRC) [4].

2327-4662 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

An IR transceiver module is made for the Raspberry Pi along with the needed software.

2) We have identified the deficiency of TLS mutual authentication implemented by the popular message queuing telemetry transport (MQTT) open-source package Mosquitto [5], [6] for a trustworthy IoT system. Mosquitto's client authentication does not identify a particular client through its certificate for authentication. We patch the system by saving the certificate hash in a database and explicitly identifying a client.
3) We also discuss and analyze the factors that affect the IR signal transmission range in our context.
4) We explore the security implications emerging from IR communication by the use of our IR transceiver and demonstrate attack scenarios. For example, a remote-controlled drone equipped with our device can turn off/on a TV as shown in the YouTube video https://youtu.be/rPbzPbWrbf8 or YouKu video http://v.youku.com/v_show/id_XMzQ0Njc5MzM3Ng. To demonstrate the attack range, the video at https://youtu.be/H27L0H4Kt5M shows that our IR transceiver is able to turn off a lightstrip in a corridor from a distance of about 23 m while another video at https://youtu.be/OJ17QU9OvBc demonstrates that the IR transceiver can turn off the lightstrip behind glass from about 13 m away. Those attacks shall be carefully investigated for smart city and home applications involving legacy IR devices.

The remainder of this article is organized as follows. We introduce the background knowledge of IR communications, Raspberry Pi, and MQTT in Section II. Our smart IR system is elaborated in Section III, including both the hardware and software of our smart IR transceiver, smart IR controller, as well as cloud server. Besides, we also discuss the security concerns of the smart IR transceiver in Section IV. We evaluate our smart IR system in Section V. Related work is discussed in Section VI and the conclusion is given in Section VII.

II. Background

A. Infrared Communications

In the consumer IR communication, the sender IR module first modulates a signal (i.e., a sequence of 0s and 1s) into pulses of IR electromagnetic waves, i.e., the carrier. The receiver receives the electromagnetic waves and demodulates the waves into the original signal. Fig. 1 shows three types of consumer IR signal modulating schemes [7]: 1) pulse distance; 2) pulse length; and 3) bi-phase, where on means the IR source, such as an IR LED is turned on, while off means the IR source is turned off. Fig. 1(a) shows the pulse distance modulation. A bit is composed of a carrier modulated pulse and space, where pulse length on_length is a constant. The space length differentiates 1 and 0. off_length_1 refers to 1 and off_length_0 refers to 0. The transmitting time for 1 is on_length + off_length_1 and no_length + off_length_0 for 0. Fig. 1(a) shows the time taken to transmit 1 is longer than 0. Fig. 1(b) shows the pulse length modulation. Each bit is also made up of a carrier modulated pulse and space. However, the pulse length makes a distinction between 1 and 0 while the space length is a constant. Specifically, the pulse length for 1 and 0 is denoted as on_length_1 and on_length_0, respectively. The space length is denoted as off_length. Fig. 1(b) shows 1 requires a longer carrier modulated pulse than 0. The bi-phase modulation is shown in Fig. 1(c), within which each bit has equal length. In other words, half of the length is a carrier modulated pulse, denoted as on, while the other is space, denoted as off. 1 is represented by the transition on → off. 0 is represented by the transition off → on.

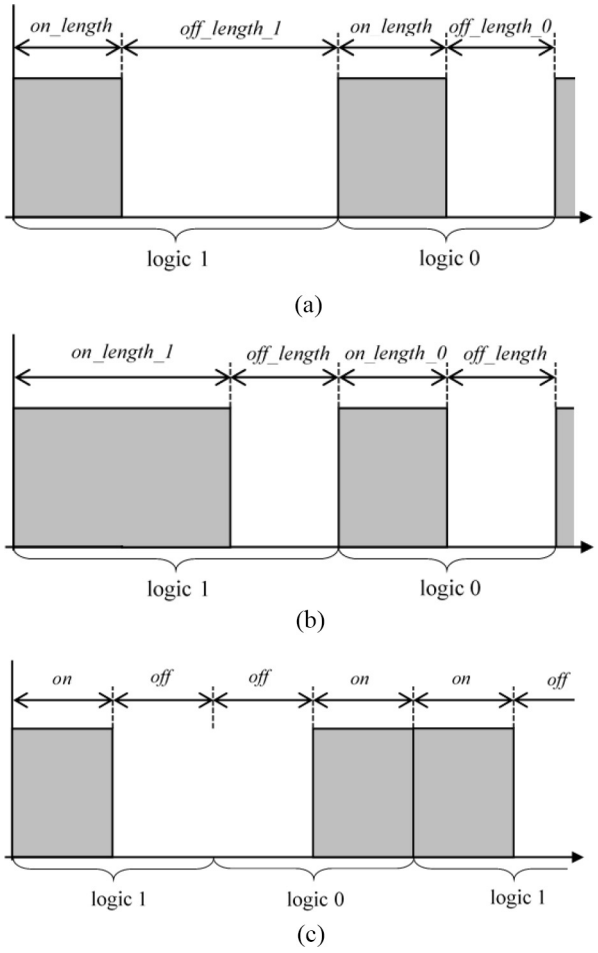

Fig. 1. IR signal encoding. (a) Pulse distance. (b) Pulse length. (c) Bi-phase.

Popular IR transmission protocols for consumer electronics include the Philips RC-5 protocol [8] and NEC protocol [9]. The bi-phase modulation is used by RC-5. Each bit lasts for 1.778 ms. The pulse and space are 889 μs and the carrier frequency is 36 kHz. An RC-5 message is composed of a 2-bit header 11, a toggle bit, a 5-bit address, as well as a 6-bit command. The toggle bit changes with each button press in order to distinguish two successive button presses impacted on a remote. The 5-bit address identifies the IR device [10]. RC-5 is widely employed by manufacturers specialized in audio and video equipment in the United States and Europe, as exemplified by speakers and TVs.

Pulse distance modulation with on_length of 562.5 μs, off_length_1 of 1.6875 ms, and off_length_0 of 562.5 μs is used by the NEC protocol. The transmission time is 2.25 ms

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

and 1.125 ms for 1 and 0, respectively. An NEC message starts with a header of 9-ms pulse and 4.5-ms space, followed by a 16-bit address (8-bit address and 8-bit logical inverse of the address) and a 16-bit command (8-bit command and 8-bit logical inverse of the command). An additional 562.5-μs pulse indicates the end of transmission. In case of a key being continuously pressed, the command is sent out once and a repeat code is then sent every 108 ms. The repeat code is composed of a 9-ms pulse, a 2.25-ms space and, finally, a 562.5-μs pulse to mark the end of the message transmission [11].

B. Raspberry Pi

A Raspberry Pi is a lightweight computer that runs on an ARM CPU. A Pi may runon various operating systems. In this article, all the Raspberry Pis use Raspbian, a Debian-based Linux OS system. In our experiments, Raspberry Pi 3 and Zero W were adopted despite other similar devices.

C. MQTT

MQTT [5] is a popular lightweight protocol to implement IoT communications. MQTT is a topic-based, publisher, and subscriber messaging system. A message contains a message topic and message payload. The topic is a unique string that serves as the identifier for a type of message. A publisher is any client that sends messages. A subscriber is any client that listens for incoming messages of a particular topic. The term client or node refers to any system/device that connects to a broker and publishes and/or subscribes to topics while the broker relays the messages between clients.

Mosquitto [6], an open-source implementation of MQTT, supports MQTT over TLS [12]. Mosquitto provides a broker along with publish and subscribe tools. Paho-mqtt [13] is a library in Python for MQTT and provides APIs to subscribe and publish to topics.

III. Smart and Trustworthy IR System

In this section, the architecture of the smart IR system is introduced. Then, the hardware and software of the smart IR transceiver are presented in detail, followed by the explanation of the smart IR controller and trustworthy cloud server which are aimed at controlling the legacy IR devices from anywhere over the Internet. Finally, factors that affect the IR transmission range are discussed.

A. Architecture of Smart IR System

Fig. 2 shows the smart IR system architecture. It is made up of four components: 1) a legacy IR device; 2) a smart IR transceiver; 3) a cloud server/broker; and 4) a smart IR controller. The legacy IR device can be any consumer IR device. The smart IR transceiver is able to control the legacy device by recording IR signals of the legacy IR remote and replaying them. The smart IR transceiver is connected to the cloud server via the Internet. The smart IR controller can be a smartphone with our STIR app, which record and replay IR signals via the cloud server under the command of users.

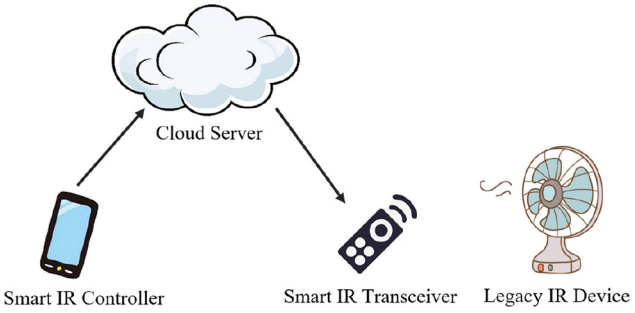

Fig. 2. System architecture.

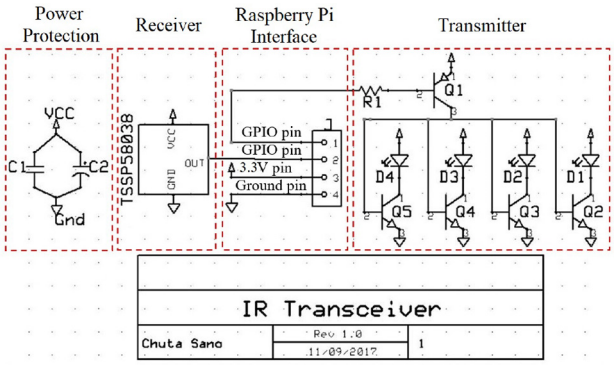

Fig. 3. Schematic of the IR transceiver board.

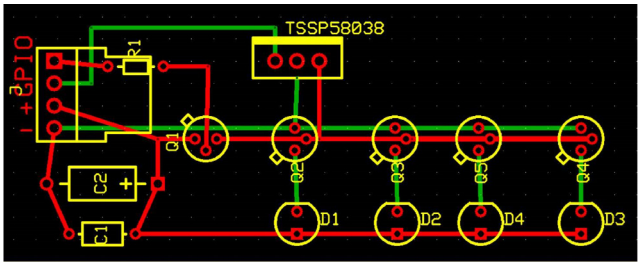

Fig. 4. PCB design of the IR transceiver board.

TABLE I
Parts List

ID on PCB Design	Part Name
C1	Ceramic 0.1μF capacitor (COM-08375)
C2	100μF Capacitor with 25V + rating
D1, D3	Wide IR LED (IR333C/H0/L10)
D2, D4	Narrow IR LED (IR333-A)
Q1	PNP Transistor (PN2907)
Q2, Q3, Q4, Q5	NPN Transistor (2N3904)
VS1838B	38K IR Receiver Module
R1	1K 1/4 W 5% Resistor

B. Smart IR Transceiver

The smart IR transceiver includes two parts: 1) hardware and 2) software. We elaborate on them in detail as follows.

1) Hardware of Smart IR Transceiver: The smart IR transceiver consists of a Raspberry Pi and an IR transceiver board. Fig. 3 illustrates the schematic of the IR transceiver board. Fig. 4 shows the printed circuit board (PCB) design of the board and Table I lists its parts. The IR transceiver board is composed of a transmitter circuit, an I/O interface, a receiver

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

module, and a power protection circuit. A Pi is connected to the transceiver's I/O interface, which includes four GPIO pins, J-1, J-2, 3.3 V, and GND. The transceiver receives power from the 3.3-V pin and the ground is connected to a ground pin on the Raspberry Pi. The Raspberry Pi sends signals through the GPIO pin (J-1) to the transceiver board, which turns the IR LEDs/transmitters on or off. The receiver module of the transceiver board records the signals from a legacy IR remote and sends it to the Pi through the GPIO pin (J-2). The receiver module uses a 38-kHz IR module of 940-nm peak wavelength.

The transmitter circuit of the IR transceiver board is optimized. First, the power supply is routed through the power protection circuit consisting of a 100-μF capacitor (C2) and a 0.1-μF capacitor (C1). Since the IR signal is fired in short bursts, C2 can store charge to increase the stability in case the Raspberry Pi fails to transmit steady current to the LED. C1 removes small electric noise and protects the whole circuit.

Second, four IR LEDs (D1$\sim$D4) with wavelength of 940 nm are placed in parallel to maximize the coverage of the IR signal. In Section V-C, we will evaluate the impact of the combination of LEDs with wide coverage and narrow coverage on the transmission range.

GPIO pin commands pass through the PNP (Q1) transistor to each IR LED, which is connected to a driver transistor (NPN transistor). The transistor Q1 is used as a buffer to amplify the weak signal from the GPIO pin and push the required amount of current into the bases of the NPN driver transistors (Q2 $\sim$ Q5). Each NPN transistor is capable of amplifying the power of the command signal. To protect the circuit, they are designed to turn off the LEDs if the current is greater than 100 mA. These transistors increase the intensity of the IR light, leading to a higher range of IR transmission.

2) Software of Smart IR Transceiver: Software of the smart IR transceiver runs on the Raspberry Pi and has two parts, one for communication with the IR hardware transceiver board and the other for communication with the cloud server to receive commands from the smart IR controller. The smart IR transceiver software can handle intricacies of low level arising from the IR communication.

PiGPIO library written in C++ is adopted by the receiver module software. It keeps account of the duration between low and high inputs received from the receiver in raw format, in which positive numbers represent the on duration of LED in microseconds, while the negative ones represent the off duration in microseconds. The most important function used in the receiver module software is *gpioSetAlertFunc()*. It records the GPIO state changes (from high to low or from low to high) and timestamps it in microseconds. This function samples the status of the GPIO pin and records the status (high or low) and timestamp after the status is changed. The samples are captured by the direct memory access (DMA) hardware and written to a large cyclic buffer in the memory. The user is able to access this buffer every millisecond and the default buffer size is 120 ms of samples. The default sampling rate is 200 kHz. Recall that the IR signal bandwidth is 38 kHz. According to the Nyquist–Shannon sampling theorem [14], a signal waveform can be recovered when the sampling frequency is greater than or equal to twice the signal

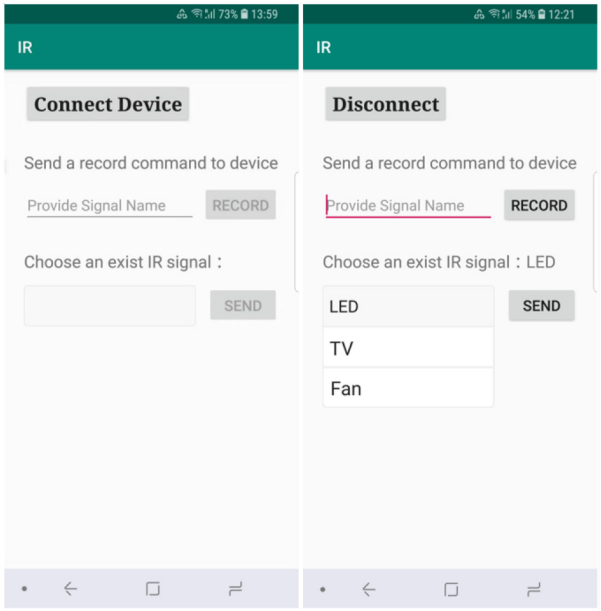

Fig. 5. Android application.

frequency. Therefore, we can correctly record and replay the recorded signal using the IR transceiver board.

The STIR app installed on the smart IR controller allows the user to name the recording, which is saved to "ircodes.txt" in the Raspberry Pi by default. Each recording has two lines: the first one is "name: <name>," namely, the name entered from the STIR app on the smart IR controller. The second is a list of integers delimited by spaces to represent the recorded IR signal.

We implement the transmitter software in C++. The transmitter software uses the PiGPIO library to have the nanosecond accuracy for the GPIO control. This is necessary to correctly replay the IR signals. The transmitter software communicates with the smart IR controller to obtain the name of the IR signal to be replayed. The transmitter software then attempts to read the matching line with the same name from the *ircodes.txt* and replays the previously recorded signal. Our transmitter software has implemented the previously mentioned raw format scheme, pulse-distance, pulse-length, and bi-phase encoding, and the NEC and RC-5 protocols.

Using the IR transceiver hardware and software, we can now connect our IR transceiver hardware board to a Raspberry Pi as a smart IR transceiver and control an IR device by recording and replaying its IR signals. Since a Raspberry Pi has the Internet capability, we can connect the Raspberry Pi to the Internet and control the IR device over the Internet through a smart IR controller.

C. Smart IR Controller

We implement the STIR Android app that remotely controls the IR transceiver circuit board connected to the Raspberry Pi through the MQTT protocol. Fig. 5 shows the Android app, which can connect to the smart IR transceiver through the cloud server. We can use the app to send a command to the IR transceiver circuit board so as to record the IR signals

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

from any legacy IR controllers. Additionally, we can choose any recorded IR signal (such as TV, LED, and fan shown in Fig. 5) and send it to the smart transceiver that can transmit the corresponding IR signals to the IR device.

D. Trustworthy Cloud Server

Mosquitto is chosen as the message broker for the MQTT protocol which has been expounded in Section II-C. Our Mosquitto cloud server is built on an Amazon EC2 virtual machine backed with the SSL/TLS transfer security. Both the IR transceiver and smart IR controller can be connected to this server. The smart IR controller sends a control command to a specific topic, to which the smart IR transceiver subscribes. Upon receiving the control command, the smart IR transceiver can send the signal via its IR transceiver board to the corresponding IR device. A Mosquitto broker is installed on the Pi of the smart IR transceiver so that the smart controller and IR transceiver can locally communicate through the MQTT broker on the Pi. The Raspberry Pi of the smart IR transceiver can be configured to be a wireless access point (AP), denoted as *Rasp-AP* so that the smart IR controller can connect to Rasp-AP and send a control command to the broker on the Pi. The Pi then commands the transceiver board and sends the IR signal to the corresponding IR device.

We found that the implementation of the current mutual authentication by Mosquitto, which supports SSL/TLS, is flawed to some extent. For mutual authentication and connection encryption, the configuration of the Mosquitto server should contain the certificate of the certificate authority (CA), the server certificate issued by the CA and the server private key. Thus, a device which is connected to the Mosquitto server should have the CA certificate, the client private key, as well as the client certificate signed by the CA. As for the server authentication, the CA certificate is used by the device to validate the server certificate and check if the subject/domain name in the server certificate matches the domain name the client intends to connect to. However, the Mosquitto's client certificate-based authentication strategy has security pitfalls. The Mosquitto server only checks if the client certificate is signed by the trusted CA, without verification of the particular identity of the client who is allowed to use the server or not. Therefore, if an attacker obtains any valid client certificate generated by the CA, he/she can spoof the device and may communicate with the server in a malicious way, for example, sending fake messages.

With the purpose of addressing the flaw of Mosquitto's certificate-based client authentication, the Mosquitto source code is revised and the hash of the client certificate of a device along with the client certificate ID is stored in a database at the Mosquitto server. In *src/net.c* of the Mosquitto, we add a new function called *client_certificate_verify*(). After the validation of the client certificate by Mosquitto, the new function searches the ID of the certificate sent by the connecting device in the server database. If the certificate ID does not exist, the client is disconnected. If yes, the new function hashes the client certificate provided by the connected device and compares it to the hash value associated with the client certificate ID in the server database. If they are the same, the client is allowed to connect to the server. Otherwise, the server is disconnected from the client.

E. IR Transmission Range

A legacy IR receiver is typically packaged by a combination of a photodiode and an integrated circuit, which can receive an IR signal and decode the received IR signal. There are two common IR wavelengths, 850 and 940 nm for IR LEDs. With more power and long transmission range, the 850-nm LED produces a slight red glow, visible to human eyes. With a short transmission range and lower power, 940-nm LED is not visible to human eyes, but it is cheaper and commonly used to control small consumer electronics.

The transmission range of the IR signal is mainly up to the radiant intensity of the IR emitter/transmitter together with the irradiance of the legacy IR receiver [15]. The maximum transmission range $d_{\max}$ of a smart IR control system can be calculated with the use of the following equation [15]:

$$d_{\max} = \sqrt{\frac{I}{E_{\min}}} \tag{1}$$

where I is the radiant intensity of the IR emitter and $E_{\min}$ is the minimum irradiance of the legacy IR receiver.

The radiant intensity of an IR LED mainly rests upon the current flow through the LED. The higher the current is, the greater the intensity is. Nonetheless, each type of LED has its limit on the maximum current it can pass. The radiant intensity is also pertinent to the viewing angle of the IR LED. The viewing angle θ is defined as the angle from which LED light spreads. Outside of θ, the intensity falls under 50% of its maximum brightness [16]. If multiple LEDs are placed in parallel, the radiation intensity of the overlapping portion of the viewing angle can be enhanced by superposition. This is the reason for our use of multiple IR LEDs to increase the transceiver's transmission range.

Except for the above-mentioned factors, there are other factors that cast an influence on the transmission range. The reflectivity of the surrounding walls, the optical transmittance of the window in front of the receiver, and external noise, such as lights from sun or light bulbs, all influence the coverage and range of the signal. Therefore, noise suppression capabilities of legacy IR receivers can affect the IR signal receiving distance too.

IV. Security Implications

In this section, a discussion on the threats from the IR transceiver that could be employed to attack IR controllable devices, as exemplified by TVs, air conditioners, fans, and thermal control units, is given, since the relevant attacks may result in damages from daily inconveniences to property loss.

A. Replay Attack

Since consumer IR devices are lack of regular encryption of their IR codes, security concerns can arise. For instance, TVBGone [17] can turn on/off TVs by replaying known codes.

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

It takes about two minutes for TVBGone to replay all the codes, and this is definitely a congruent time frame for the attacks of any TV [17]. As for our IR transceiver, it is able to keep account of any IR codes from IR controllable devices and run these codes accordingly. Under the control of a Raspberry Pi, this IR transceiver is prone to being customized to attack any device that is controlled by IR.

B. Brute-Force Attack

The brute-force attack is capable of traversing all possible bits of a given protocol and attacking any unknown device with the use of that protocol. In terms of the NEC protocol, a potential command is built on the unique address and command bits. Since 8 bits are included in the address and command, respectively, a total of $2^{16} = 65\,536$ possible combinations come into shape. According to our experiments, it takes 67.5 ms for each command to be sent. However, with a total of 65 536 distinct NEC messages, the time duration for the worst-case scenario is just beyond one hour. In order to give an illustration of the brute-force attack, the NEC brute-force attack is deployed against a remote-controlled lightstrip [18]. Its objective is to turn on/off the lightstrip. With the employment of this method, the light can be hacked in milliseconds. In the attack, 0x00 and 0x00 were set to be the starting bits for address and command, respectively. As a matter of fact, address 0x00 and command 0x02 are used by the lightstrip.

In total, 11 bits are adopted by RC-5, with 5-bit addresses and 6-bit commands. Therefore, 2^{11} different messages can be found in the RC-5 protocol. 24.9 ms is needed for sending an RC-5 command signal. The RC-5 brute-force attack can be finished within 51 s.

C. Drone Attack

There remains one difficulty in attacking IR devices, that is, the IR transceiver—the doer of this attack—needs to approach the target IR devices as close as possible. In order to circumvent accessibility-related issues, the Raspberry Pi and IR transceiver could be placed on a drone. Since the former is Internet-based, the IR transceiver can be controlled by people from anywhere.

The drone attack in our experiment is performed with a DJI Phantom 2 [19]. With the purpose of minimizing weight and maximizing flight duration, the on-board camera was removed. Powered by a mini battery [20], a Raspberry Pi 3 was mounted onto the drone together with an early version of our IR transceiver. In accordance with the experiment, it has been found that the TV inside the second-floor conference room can be turned on and off by the drone which flew outside.

D. Securing IR Communication

For the defense against the above-mentioned attacks, one may want to encrypt the IR communication. The major issue of encrypting IR messages is the overhead. It can be seen from the discussion above that an IR message is often fewer than two bytes. Using encryption or the message integrity code will add tens of extra bytes to the IR message. This may affect the transmission delay of the IR message and may be a reason we do not see much work on encrypting IR communication. The advance of MCUs actually makes encrypting the IR communication possible [21] with hardware crypto acceleration. A capable MCU can be used at the receiver side of IR communication for this purpose. However, for high-value assets, a more capable communicating venue, such as WiFi, may be used rather than IR.

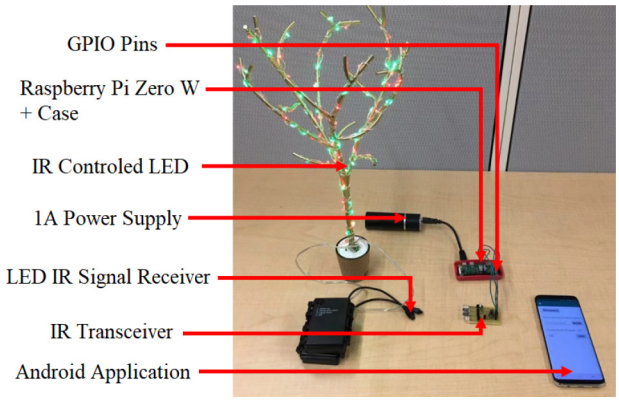

Fig. 6. IR device setup.

V. Evaluation

In this section, a prototype of STIR, the smart and trustworthy IoT system connecting legacy IR devices to the Internet is introduced first. Then, the cost of the prototype, evaluation of its performance, and the attacks against IR communication are discussed. Finally, limitations and potential improvements to STIR are given for further discussions.

A. Prototype of STIR

Fig. 6 shows the prototype of STIR and a test setup. On the left is an IR controlled LED strip which is wrapped around a tree-like artifact together with its IR receiver. On the right is the IR transceiver circuit board connected to the Raspberry Pi Zero W [22] through GPIO pins. From bottom to top, the four connected pins found on the IR transceiver are linked up with GPIO 22 (sender pin), GPIO 23 (receiver pin), a 3.3-V power pin, and a ground pin on the Raspberry Pi Zero W, respectively. The Raspberry Pi Zero W is connected to a battery.

The IR transceiver circuit board controlled by the Raspberry Pi Zero W GPIO pins can be used to record any IR signal and replay the signal accordingly so as to control the IR LEDs. The Raspberry Pi Zero W can have communication with the smartphone (Samsung Galaxy s8) via the MQTT protocol. Our IR recording and replaying software is much more reliable and efficient than the LIRC package [4] shipped with Linux. LIRC fails at a high rate while attempting to learn/record a signal. Even if it obtains a successful learning result, approximately 30 s are needed to complete this procedure because of its reliability issues. Nevertheless, our software does not fail and implements various APIs to abstractify GPIO interaction, simplifying any generic IR communication needs.

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

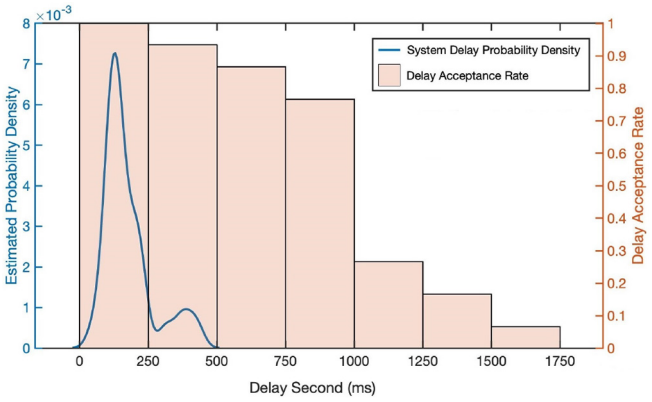

Fig. 7. Acceptance response delay time.

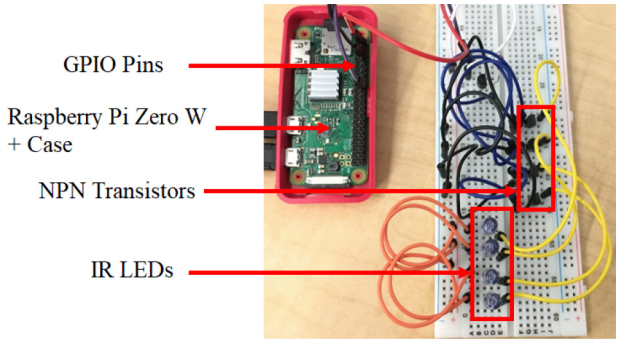

Fig. 8. LED radiation intensity test setup.

TABLE II
TRANSMISSION DISTANCE FOR DIFFERENT LEDS

	3.3V		5V	
Number of LEDs	Wide LED	Narrow LED	Wide LED	Narrow LED
4	40.50m	22.80m	47.31m	38.10m
3	33.20m	21.27m	35.76m	25.50m
2	18.74m	13.92m	21.20m	17.71m

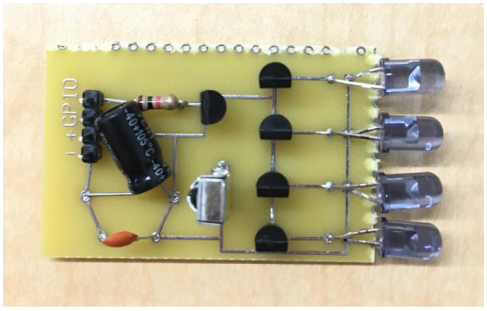

Fig. 9. Environmental factors test setup.

We also study the delay when a command is sent from the smartphone to the IR transceiver circuit board over the cloud. The experiment is performed 30 times, the average delay is 175 ms, and the estimated density function is shown in Fig. 7. We also survey 30 people to study tolerable delay. We artificially add delay at the smartphone sending a command to turn on the lightstrip and ask the subject if it is tolerable. The survey shows that the delay of our smart IR system can be acceptable to 99% $(1 \times 0.85 + 0.93 \times 0.15)$ of people.

B. Cost

Owing to the variations of cost arising in PCB printing (such as the costly construction of one module), the cost of the IR transceiver module is approaching $8.00. Our experiments were performed on the Raspberry Pi Zero W, costing only $10.00. As a result, the direct cost of the whole setup can be minimized to approximately below $18.00 with a Raspberry Pi Zero W.

C. Transmission Range

As previously mentioned, two different types of IR LEDs are adopted in our experiments, i.e., wide IR LEDs and narrow IR LEDs, with 940-nm wavelength to evaluate the effect of the radiation intensity and viewing angle of the emitter on the transmission range. The wide IR LED has maximum 45 mW/Sr (milliwatt per steradian [23]) radiant intensity with a 40° viewing angle and the narrow IR LED has maximum 80 mW/Sr radiant intensity with a 20-degree viewing angle. We build the IR transceiver circuit on a breadboard shown in Fig. 8 and evaluate the IR LED maximum transmission distance in a large dim empty room. On the left side of the figure is the Raspberry Pi Zero W [22], which is used to connect and control the IR transceiver circuit breadboard through GPIO pins.

Table II shows the IR LED maximum transmission range for different types of LEDs, the number of LEDs, and the power supply. The maximum range rests with the finding of the longest distance at which our transceiver can operate the lightstrip module. For both types of LEDs, more numbers of LEDs lead to better range. It has been found that with the use of the same type of LED, the LED with a 5-V power supply has better range than that with 3.3-V power supply because more current can be supplied by the former to the LED. Theoretically, the narrow LED has better range, but the experimental results show that the wide LED can transmit farther. In our view, this is pertinent to the LED viewing angle and the sensitivity of the receiver. The sensitivity of an IR receiver is the determinant of the weakest signal strength that the receiver can receive and resolve. When the signal strength is lower than the receiver sensitivity, the IR signal cannot be correctly demodulated by the receiver. When the IR device is controlled from a long distance, it cannot be guaranteed whether the center of the transmitter points at the receiver. As the wide LED is equipped with a larger viewing angle, it can help the receiver reliably receive a sufficiently powerful IR signal.

The impact of the environmental factors cast on the IR signal transmission range with the actual PCB, not the breadboard in Fig. 8, is also evaluated. In accordance with the previous experimental results, the four wide IR LEDs can achieve the longest signal transmission distance. Then an IR transceiver circuit board with four wide IR LEDs on the same side of the PCB is assembled, as shown in Fig. 9, to test the role played by the environmental factors. The IR receiver module used in our test environment is encapsulated in epoxy resin

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

TABLE III
TRANSMISSION DISTANCE IN DIFFERENT ENVIRONMENTS

Situation	3.3V	5V
Inside, large room with no pockets	20.18m	22.93m
Outside, in shade, through a window, receiver inside a room	8.78m	10.58m
Outside, receiver facing the sun	0.60m	0.75m
Outside, receiver facing away from the sun	3.07m	3.21m
Outside, in shade	10.71m	13.21m

TABLE IV
TRANSMISSION DISTANCE IN DIFFERENT WEATHER CONDITION

Situation	Sunny	Cloudy
Outside, in shade, through a window, receiver inside a room	8.78m	6.99m
Outside, receiver facing the sun	0.60m	1.08m
Outside, receiver facing away from the sun	3.07m	3.68m
Outside, in shade	10.71m	9.89m

and consists of a PIN photodiode, a preamplifier, and an IR filter. The IR sensitive area of the IR photodiode is denoted as the IR-photosensitive surface of the IR receiver. The signal transmission range over 3.3 and 5-V power supply is thus evaluated.

Table III presents the transmission range of the IR transceiver module used in five different scenarios. In the first case (*Inside, large room with no pockets*), both the IR transceiver and receiver are placed indoors. In the second case (*Outside, in shade, through a window, receiver inside a room*), the IR transceiver is put outdoors under the shadow of the building and controls the receiver (which was located indoors) through the window. As for the three scenarios, both the IR transceiver and receiver are placed outdoors. In the third case (*Outside, receiver facing the sun*), the IR transceiver is directly exposed to sunlight, and the IR-photosensitive surface of the receiver faces the sunlight. In the fourth case (*Outside, receiver facing away from the sun*), the IR transceiver is directly exposed to sunlight, but the receiver's IR-photosensitive surface faces away from the sunlight. In the fifth case (*Outside, in shade*), both the IR transceiver and receiver are placed under the shadow of the building. All the experimental results were obtained at noon (approximately) on a sunny day.

It can be seen from Table III that the transmission range with the PCB and breadboard is different due to divergent materials. As expected, the 5-V power supply can provide a longer transmission range than the 3.3-V one. The distance can be maximized in a room with minimal noise. Glass windows between the transmitter and receiver can decrease the range, mainly because of its optical transmittance and reflectively. Therefore, in order to perform a drone attack presented in Section IV-C, the drone needs to be at most around 10 m away from the target device. All the three outside cases in Table III indicate that the IR transmission range is under the sway of ambient light. The experimental results show that the receiver is more susceptible to the sunshine than the IR transmitter. Based on the related findings, a shorter transmission range can be obtained when the receiver is outside under the sun.

Here comes the evaluation on the impact of outdoor weather conditions cast on IR signals for the 3.3-V power supply. The experimental results are obtained around noon on a sunny day and a cloudy day, respectively. Table IV illustrates the transmission range of the IR transceiver module for different weather conditions. It has been noted that according to the experimental results, on a cloudy day, it has little effect on the transmission distance, regardless of the direction received by the receiver.

D. Limitations and Improvements

A legacy IR device often does not implement acknowledgment to commands sent from a remote. The IR communication often relies on the user checking the status of the device and performing accordingly. To implement the acknowledgment, the IR device needs a channel to send acknowledgments back to the smart IR transceiver. For example, an extra IR transmitter can be installed at the device and an extra IR receiver at the remote. Our smart IR transceiver has an IR receiver. However, it may not be feasible to adopt an IR transmitter at the legacy IR device. In such a case, it is a challenge to implement IR acknowledgment. One potential solution will be to use a surveillance camera to monitor the status of the device while it could be an expensive solution in many scenarios.

In this article, we fix the client authentication vulnerability of the open-source MQTT message broker Mosquitto and set up an MQTT server at Amazon EC2 for trustworthy control of the smart IR transceiver. The commercial Amazon AWS IoT can also be used to achieve the goal. AWS IoT implements TLS mutual authentication between the server and the client. The client shall be equipped with a certificate of the CA, client certificate, and client private key. The Raspberry Pi Zero W can handle computing power required by the public-key cryptography.

An MCU can replace the Raspberry Pi Zero W used in STIR to further reduce the cost. We have explored the use of MCUs and crypto modules in IoT applications and demonstrated that hardware and cost may not be the bottleneck of IoT security and privacy in various application domains [21]. We find many of those recently released MCUs, such as Espressifs ESP32, TIs CC3220, Microchips cryptographic co-processor ATECC608A, and SAML11, which implement hardware security, system/firmware security, network security, and data security. We have validated the performance of the cryptographic and networking operations of IoT devices based on various MCUs and crypto modules. Those MCUs often cost a few dollars. For example, ESP32 may cost less than three dollars.

VI. RELATED WORK

STIR is the first of its kind controlling legacy IR devices through a cloud in a trustworthy way. Since STIR has been presented in June 2018 at a conference, commercial products began to show up in 2019 [24] and might be inspired from this article.

We now review early work on IR recording and replay. The LIRC, a software package which implements the low-level details of interacting with IR signals, runs a helper daemon

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

at the core [4]. Our software, with remarkable usability, is a significant advancement of the LIRC. IrSlinger [25] is a simple IR sender program that uses the PiGPIO daemon for precise timing on Raspberry Pi.

Toolkits have also been developed to play with IR signals. TVBGone, a device of light weight, is capable of turning on/off TV [17]. Governed by an IC chip, it hardcodes possible TV power toggle signals in the IC chip. Its precise timing is effected and gained via a ceramic resonator. Arduino Universal Remote [26], with the use of a breadboard, can only keep account of and playback one signal without security and remote control over the Internet. IR photo transistor, rather than IR transceiver, is employed to record signals by IRRemberizer [27] to keep a record of signals from 30 to 60 kHz. However, IRRemberizer suffers from less reliability and declining range because the photo resistor is of high likelihood to be affected by interference from exterior IR sources [27]. Based on this consideration, a 38-kHz IR receiver module with high stability and range in the process of recording is used in our design.

Parsovs [28] discussed and analyzed the practical security and usability issues related to TLS client certificate authentication (CCA) in the Apache's mod_ssl module and a number of browsers. This article first illustrates the TLS handshake process and protocol messages exchanged between a server and a client during the server authentication and client authentication. Next, we describe the parameter configuration for client authentication in the mod_ssl module of Apache and explain how the mod_ssl module performs certificate verification. We then discuss the practical issues with TLS CCA, some of which are described as follows.

1) We state that CCA requires additional verification at the application level because the server only verifies whether the commonName attribute in the client certificate contains the name and personal information of the natural person in the expected form.
2) An attacker can enumerate the server's trust settings with certificates that can be easily obtained. If the attacker has any root certificate existing in the server's trust store, he can determine the SSL verification depth.
3) The client's certificate is transmitted in plaintext, so an attacker can intercept and use the content of the client certificate to track the client. If the client certificate contains the user's personal data and is signed by any trusted spare CAs, an attacker could impersonate the user.
4) The server must reliably determine the freshness of the client certificate to ensure the security of the communication. However, the Apache's mod_ssl module does not support client certificate freshness verification. Finally, we test and analyze multiple websites that support TLS CCA and use the mod_ssl module and provide recommendations for service providers, mod_ssl developers, and browser vendors, respectively, to resolve security issues found in deployment of TLS CCA.

Fig. 10 illustrates the messages in the TLS 1.3 handshake [29]. It can be observed in TLS that both the certificate-based server authentication and certificate-based client authentication only check the validity of the corresponding certificate. However, to defeat the man-in-the-middle (MITM) attack, either the server or the client shall verify if the connecting party is the subject presented in the certificate. For example, in server authentication, the domain name in the server's certificate should match the domain name that the client wants to access [30]. In client authentication, the subject in the client certificate shall be registered with the server and verified [31]–[33].

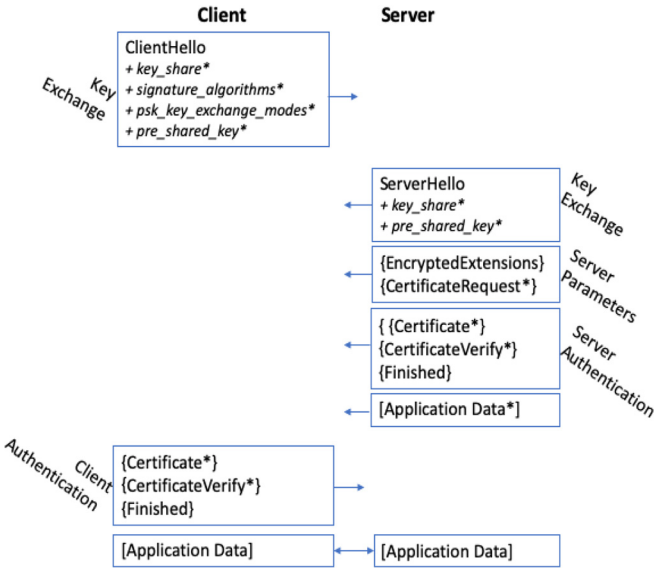

Fig. 10. TLS 1.3 handshake. + refers to extensions in a message; ∗ refers to optional or situation-dependent messages/extensions; {} refers to messages encrypted using keys derived during handshake; [] refers to messages protected using keys derived through handshake [29].

VII. Conclusion

A low-cost system was introduced in this article to convert legacy IR controlled devices into smart IoT devices connected to the Internet by means of a Raspberry Pi Zero W with a hardware IR transceiver module. Factors that affect the transmission range of the IR transceiver are discussed and analyzed. Besides, with mutual authentication of our patched MQTT server Mosquitto, the security of network communication is guaranteed. A powerful technology often has two sides. We also discussed the potential security threats arising from our IR transceiver. A drone equipped with our smart transceiver may fly close to victim IR devices and incur damages from daily inconveniences to property loss.

Acknowledgment

Any opinions, findings, conclusions, and recommendations in this article are those of the authors and do not necessarily reflect the views of the funding agencies.

References

[1] G. Xu, W. Yu, D. Griffith, N. Golmie, and P. Moulema, "Toward integrating distributed energy resources and storage devices in smart grid," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 192–204, Feb. 2017.
[2] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.

[3] L. Soto. (Nov. 13, 2015). *The Uses of Infrared Technology in Your Everyday Life*. [Online]. Available: http://www.yahalamedia.com/the-uses-of-infrared-technology-in-your-everyday-life.html

[4] C. Bartelmus. (May 26, 2016). *Linux Infrared Remote Control*. [Online]. Available: http://www.lirc.org/

[5] OASIS. (Oct. 29, 2014). *MQTT Version 3.1.1*. [Online]. Available: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.pdf

[6] I. Eclipse Foundation. (Feb. 2, 2019). *Mosquitto*. [Online]. Available: https://mosquitto.org/

[7] *Infrared Remote Control Implementation With MSP430FR4xx*, Texas Instrum. Incorp., Dallas, TX, USA, Jun. 2015. [Online]. Available: http://www.ti.com/lit/an/slaa644b/slaa644b.pdf

[8] *Philips RC-5 Protocol*, SB-Projects, New York, NY, USA, Dec. 2017. [Online]. Available: https://www.sbprojects.net/knowledge/ir/rc5.php

[9] *NEC Protocol*, SB-Projects, New York, NY, USA, Dec. 2017. [Online]. Available: https://www.sbprojects.net/knowledge/ir/nec.php

[10] A. LLC. (Sep. 13, 2017). *Philips RC5 Infrared Transmission Protocol*. [Online]. Available: https://techdocs.altium.com/display/FPGA/Philips+RC5+Infrared+Transmission+Protocol

[11] A. LLC. (Sep. 13, 2017). *NEC Infrared Transmission Protocol*. [Online]. Available: https://techdocs.altium.com/display/FPGA/NEC+Infrared+Transmission+Protocol

[12] *Transport Layer Security*, Wikipedia, San Francisco, CA, USA, Feb. 2019. [Online]. Available: https://en.wikipedia.org/wiki/Transport_Layer_Security

[13] *Paho MQTT*, E. Paho, Washington, DC, USA, Feb. 2019. [Online]. Available: https://www.eclipse.org/paho/clients/python/docs/

[14] *Nyquist–Shannon Sampling Theorem*, Wikipedia, San Francisco, CA, USA, Sep. 2018. [Online]. Available: https://en.wikipedia.org/wikiNyquistShannon_sampling_theorem

[15] *General Overview of IR Transmission in Free Ambient*, Vishay Semicond., Malvern, PA, USA, Sep. 2008. [Online]. Available: http://inside.mines.edu/∼whoff/courses/EENG383/reference/appnote2.pdf

[16] J. Baylis. (Jul. 30, 2015). *What is an LED Viewing Angle?* [Online]. Available: http://www.directsignwholesale.com/blog/2015/what-led-viewing-angle

[17] *TV-B-Gone Kit*, Adafruit Ind., New York, NY, USA, Aug. 2018. [Online]. Available: https://cdn-learn.adafruit.com/downloads/pdf/tv-b-gone-kit.pdf

[18] *BINZET 10ft 30 LEDs Warm White Starry Starry Light String Light 3xAA Battery Powered Flexible Indoor String Lights Wedding Party Light*, Amazon, Seattle, WA, USA, Jan. 2019. [Online]. Available: https://www.amazon.com/BINZET-Battery-Operated-Festival-Control/dp/B015PUX7YU

[19] *Phantom 2*, DJI, Shenzhen, China, Jan. 2019. [Online]. Available: https://www.dji.com/phantom-2

[20] *Anker PowerCore+ Mini*, Amazon, Seattle, WA, USA, Jan. 2019. [Online]. Available: https://www.amazon.com/Anker-PowerCore-Lipstick-Sized-Generation-Batteries/dp/B005X1Y7I2/ref=asc_df_B005X1Y7I2/?tag=hyprod-20&linkCode=df0&hvadid=312067196837&hvpos=1o3&hvnetw=g&hvrand=16756121846164922802&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9001901&hvtargid=aud-467077737785:pla-540937509069&psc=1

[21] B. Pearson *et al.*, "On misconception of hardware and cost in IoT security and privacy," in *Proc. IEEE Int. Conf. Commun.*, Shanghai, China, 2019, pp. 1–7.

[22] RaspberryPi. (Feb. 2, 2019). *Raspberry Pi Zero W*. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-zero-w/

[23] M. T. Inc., (Nov. 24, 2003). *Fundamentals of the Infrared Physical Layer*. [Online]. Available: http://ww1.microchip.com/downloads/en/appnotes/00243a.pdf

[24] (Jan. 2019). *Monoprice Wireless Smart Universal IR Controller*. [Online]. Available: https://www.amazon.com/Monoprice-Wireless-Smart-Universal-Controller/dp/B07N8H7BG6

[25] B. Schwind. (Mar. 9, 2017). *IR Slinger*. [Online]. Available: https://github.com/bschwind/ir-slinger

[26] X. Alto. (Feb. 15, 2017). *An Arduino Universal Remote: Record and Playback IR Signals*. [Online]. Available: http://blog.bschwind.com/2016/05/29/sending-infrared-commands-from-a-raspberry-pi-without-lirc/

[27] Sensacell. (Aug. 6, 2014). *IR Rememberizer—IR Remote Control Recorder/Player*. [Online]. Available: http://blog.bschwind.com/2016/05/29/sending-infrared-commands-from-a-raspberry-pi-without-lirc/

[28] A. Parsovs, "Practical issues with TLS client certificate authentication," in *Proc. Netw. Distrib. Syst. Security Symp. (NDSS)*, Feb. 2014, pp. 1–13.

[29] E. Rescorla. (Aug. 2018). *The Transport Layer Security (TLS) Protocol Version 1.3*. [Online]. Available: https://tools.ietf.org/html/rfc8446#section-4.6.2

[30] Oracle Corporat. (2010). *Server Authentication During SSL Handshake*. [Online]. Available: https://docs.oracle.com/cd/E19424–01/820–4811/aakhc/index.html

[31] B. Campbell, J. Bradley, N. Sakimura, and T. Lodderstedt. (Aug. 2019). *OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens*. [Online]. Available: https://tools.ietf.org/html/draft-ietf-oauth-mtls-17

[32] M. Shen, X. Tang, L. Zhu, X. Du, and M. Guizani, "Privacy-preserving support vector machine training over blockchain-based encrypted IoT data in smart cities," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7702–7712, Oct. 2019.

[33] M. Shen, Y. Deng, L. Zhu, X. Du, and N. Guizani, "Privacy-preserving image retrieval for medical IoT systems: A blockchain-based approach," *IEEE Netw.*, vol. 33, no. 5, pp. 27–33, Sep./Oct. 2019.

Authorized licensed use limited to: Southeast University. Downloaded on July 08,2020 at 12:30:33 UTC from IEEE Xplore. Restrictions apply.