Authenticating Smart Home Devices via Home Limited Channels

XIAOYU JI, CHAOHAO LI, XINYAN ZHOU, JUCHUAN ZHANG, YANMIAO ZHANG, and WENYUAN XU, Zhejiang University

Nowadays, most Internet of Things devices in smart homes rely on radio frequency channels for communication, making them exposed to various attacks such as spoofing and eavesdropping attacks. Existing methods using encryption keys may be inapplicable on these resource-constrained devices that cannot afford the computationally expensive encryption operations. Thus, in this article, we design a key-free communication method for such devices in a smart home. In particular, we introduce the Home-limited Channel (HLC) that can be accessed only within a house yet inaccessible for outside-house attackers. Utilizing HLCs, we propose `HlcAuth`, a challenge-response mechanism to authenticate the communications between smart devices without keys. The advantages of `HlcAuth` are low cost, lightweight as well as key-free, and requiring no human intervention. According to the security analysis, `HlcAuth` can defeat replay attacks, message-forgery attacks, and man-in-the-middle (MiTM) attacks, among others. We further evaluate `HlcAuth` in four different physical scenarios, and results show that `HlcAuth` achieves 100% true positive rate (TPR) within 4.2 m for in-house devices while 0% false positive rate (FPR) for outside attackers, i.e., guaranteeing a high-level usability and security for in-house communications. Finally, we implement `HlcAuth` in both single-room and multi-room scenarios.

CCS Concepts: • **Security and privacy** → **Security protocols**; • **Networks** → *Network protocol design*;

Additional Key Words and Phrases: Smart home, home-limited channel, challenge-response, key-free

ACM Reference format:
Xiaoyu Ji, Chaohao Li, Xinyan Zhou, Juchuan Zhang, Yanmiao Zhang, and Wenyuan Xu. 2020. Authenticating Smart Home Devices via Home Limited Channels. *ACM Trans. Internet Things* 1, 4, Article 24 (August 2020), 24 pages.
https://doi.org/10.1145/3399432

1 INTRODUCTION

Modern smart home systems are equipped with numerous Internet of Things (IoT) devices, ranging from powerful cameras that are capable of processing real-time videos to small resource-constrained devices such as smart light bulbs. Gartner has reported that a typical smart home will include 500 smart devices by 2022 [32]. These smart devices have greatly improved the quality of daily life by allowing users to interact and control home appliances in both local and remote

This work is supported by China NSFC Grants No. 61702451, No. 61925109, and No. 61941120.
Authors' addresses: X. Ji, C. Li, X. Zhou, J. Zhang, Y. Zhang, and W. Xu (corresponding author), College of Electrical Engineering, Zhejiang University, Room 325, Jiao 2 Building, Zhejiang University, 38 Zheda Road, Hangzhou, Zhejiang, China, 310027; emails: {xji, lchao, xinyanzhou, juchuanzhang, yanmiaozhang, xuwenyuan}@zju.edu.cn.

2577-6207/2020/08-ART24 $15.00
https://doi.org/10.1145/3399432

manners. However, the proliferation of IoT smart devices in smart homes also brings security vulnerabilities and induces privacy concerns [8, 9, 16, 18, 23, 26, 27, 40]. It has been reported that with Android-based lighting controls, one can hack into the hotel's control system and gain access to the light switches, TV, and curtains of each room. Particularly, by breaking the communication between these smart devices, outside attackers can use simple tools to sniffer your home network and launch various attacks to control your smart devices without physical access. In this article, we propose a secure communication scheme to eliminate attacks from outside attackers.

In a typical smart home, smart devices and gateways form a home network, and they communicate via one of the standard wireless communication protocols, e.g., Zigbee, Z-Wave, WiFi, and so on. When a user wants to control a device, he maneuvers the application. Then, the application sends the command to a server via the Internet, which in turn relays the command to the gateway. Finally, the gateway transmits the command to the target device in a wireless way. Although numerous secure communication protocols [11, 37] can be applied to protect the communication between applications and gateways, the key challenge is **how to protect the wireless communication between the gateways and all smart devices, especially devices that are resource-constrained**. This is the very focus of our article.

Traditional encryption-based methods may be computationally too expensive for resource-constrained smart devices. For instance, validating an RSA-based signature may overwhelm a light bulb. Much effort has been devoted to designing a lightweight authentication scheme [22, 25], e.g., by utilizing a session key to reduce the cost of public key operations. Nevertheless, the security level of encryption-based methods always depends on key management strategies. Disseminating keys to hundreds of smart devices that have different interfaces and architectures is painful. Poor practices, such as choosing weak keys or sharing one key among all devices, are not unusual and result in more vulnerabilities. To address the issue, we design a key-free communication strategy that does not rely on distributing cryptography keys and is applicable to all devices that have various degrees of computation capability.

The basis of our strategy is that it is the gateway that sends out control commands to smart devices, and gateways are typically located inside a home. We consider the inside area as a trusted environment, since the home area is an enclosed and private space. As such, validating the authenticity of commands is equivalent to ensuring that the sender of the commands is located inside the trusted areas, e.g., a house or an apartment. Thus, we can get rid of the complication imposed by key management protocols and turn to the home-area physical property for help instead. To validate whether a sender is located inside a house, we propose and investigate the *home-limited channels* (HLCs) and design a *HLC-based challenge-response* protocol (hereafter `HlcAuth`) for key-free and secure communication in smart homes. The key of HLC is that only when both the sender and a receiver are inside a home, can they reliably communicate. If either party is outside a home, then they can no longer hear each other.

Radio frequency (RF) channels do not satisfy the requirements of the HLC, because the attenuation through walls or floors is not enough to completely block the signals. To construct an HLC, we investigate a few communication medias, and choose three candidates—infrared, ultrasound, and modulated visible light (MVL). All three candidates are imperceptible and boundary-attenuated. In particular, we build a theoretical model to demonstrate how the attenuation of signals reduces the success rate of various attacks. To illustrate the practicability of our scheme, we further design and implement a smart home prototype based on `HlcAuth` in both single-room and multi-room scenarios. The architecture diagram is shown in Figure 1.

Our proposed HLC-based command authentication scheme—`HlcAuth` works as follows. A gateway sends a control command to a device using the traditional wireless channel. To validate whether the command is indeed sent by the gateway, the device initiates a challenge-response

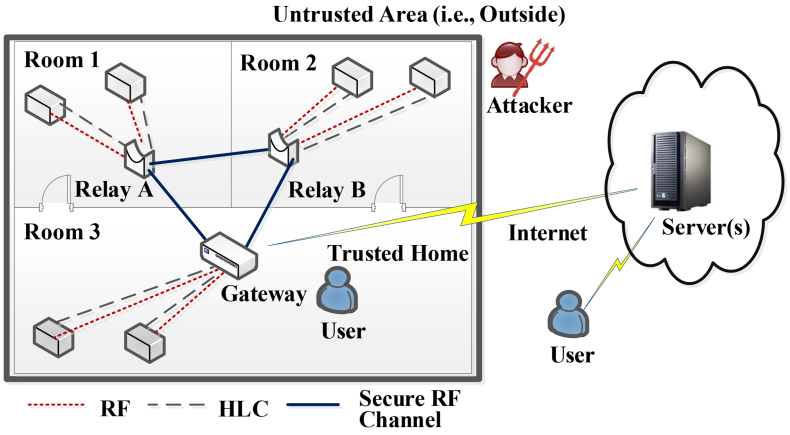


Fig. 1. The architecture of a HlcAuth-based smart home system.

query to the gateway. If the gateway passes the challenge-response test within an allowed time window, then the device concludes that the command is valid. All messages that are associated with the challenge-responses are transmitted over HLC. The underlying principle is that no compromised device can be in the home area or no attacker can enter the home, only the gateway inside the home can receive the challenge and send a response over an HLC.

To the best of our knowledge, HlcAuth is the first work that utilizes home-limited channel for secure communications in smart homes. In summary. the contributions of our article are listed as follows:

- We proposed the concept of home-limited channels and investigated candidate communication medias.
- We designed HlcAuth, a light-weight challenge-response protocol, for authenticating smart devices without using any cryptography key.
- We evaluated the performance of HlcAuth from the aspects of usability and security. Results show that HlcAuth can achieve 100% TPR within 4.2 m for legal users and 0% FPR for local outside attackers under four different scenarios.
- We implemented a smart home prototype based on HlcAuth in both single-room and multi-room scenarios.

The rest of the article is organized as follows. The smart home architecture and threat model are presented in Section 2. The home-limited channel (HLC) is introduced in Section 3, with the definition and properties in Section 3A, the boundary-attenuated model in Section 3B, and the HLC candidates in Section 3C. The protocol design and transmission scheme are given in Section 4. The security analysis is discussed in Section 5. In Section 6, the implementation of HlcAuth is presented. Experiment setups and results are provided in Section 7. Related work is presented in Section 8. Finally, Section 9 concludes the article.

2 BACKGROUND AND THREAT MODEL

In this section, we first introduce the components of a typical smart home system and describe the connection mechanism between them, and then we present our threat model.

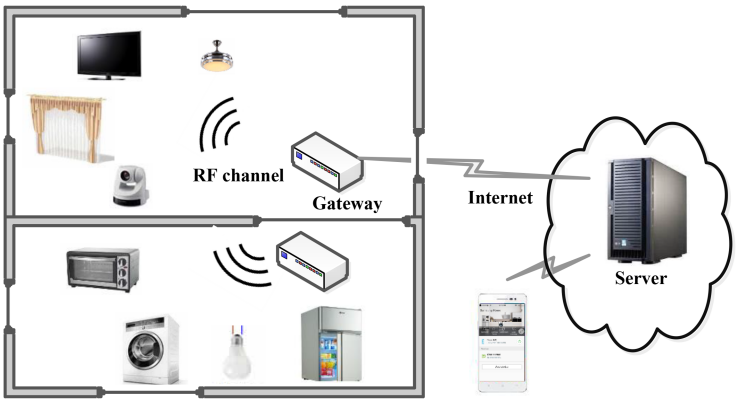


Fig. 2. A typical smart home system.

2.1 Smart Home Architecture

As shown in Figure 2, a modern smart home system generally consists of four parts: (1) smart devices, (2) gateway(s), (3) server(s), (4) one or several clients. Typically, smart devices communicate with a home gateway over radio frequency channels using home area network (HAN) protocols (e.g., ZigBee, Wi-Fi), and the gateway communicates with a server and users' mobile devices over the Internet.

(1) *Smart devices* can be further grouped into two categories: resource-constrained devices and resource-rich devices. Resource-constrained devices have limited processing and storage capabilities, thus they cannot afford the computationally expensive crypto-operations, such as bulbs and temperature monitors. While the latter ones have enough computational resource to perform complex operations such as encryption and data fusion.
(2) The *gateway* in the smart home usually acts as an aggregation point for smart devices, and serves as a bridge from the home network to external networks. It performs functions such as monitoring, controlling, and providing security and management capabilities for smart devices.
(3) The *server* is a trusted entity, and is responsible for long-term storage and analysis over large data streams.
(4) A client can be a webpage/website application or a mobile phone application. Users can operate smart devices through the home gateway directly or indirectly through an application.

The gateway takes responsibility for controlling the network data, devices and network interoperability. It can broadcast commands and queries to devices in the HAN, whenever needed. Smart devices transmit home data to the gateway using a single-hop link, and the communication between smart devices should be forwarded by the gateway. In addition, a smart home system could have multiple home gateways, which are distributed in the different rooms. They generally communicate with each other over secure RF channels with encryption.

2.2 Threat Model

The attacker's goal is to control smart devices and get user information by exploiting the vulnerabilities in home area networks. There are several ways to attack the smart home system, such as physical access, Internet attack, and HAN attack. First, we assume prior work like "ZKP

authentication" [11] are employed to protect the communication between gateways and applications, such that an attacker cannot inject forged messages to control smart devices by this link. Second, we assume that attackers cannot gain physical access to the smart home while they can launch various attacks over RF and HLC channels, and hereafter we call them local outside attackers. In addition, the in-home gateway and smart devices are trusted entities and are not compromised. Here, we describe the characteristics and abilities of local outside attackers in detail, as follows.

- **No Physical Access into a House.** Since the smart home is an enclosed and private space, malicious attackers generally cannot gain physical access to the home. Numerous work and reports [9, 18, 23, 26, 40] have shown that local attackers who are close to HAN yet outside the trusted home can hack into the HAN and control smart devices. Therefore, this article focuses on the defense of local outside attackers.
- **Multiple Attacks.** Attackers may launch the following attacks over the RF and HLC channel. (1) Replay attacks, whereby an attacker records a valid command transmission and repeats it. (2) Man-in-the-middle (MiTM) attacks, whereby an attacker secretly relays and possibly alters the communication between the gateway and smart devices. (3) Message-Forgery attacks, whereby an attacker sends a fake command on behalf of a legal gateway.
- **Attacking Equipment.** We assume that attackers can acquire both sensors (e.g., infrared and ultrasound sensors) and RF signal transceiver modules for eavesdropping, intercepting and injecting over RF and HLC channels. For instance, an attacker may secretly leave a remote controllable WiFi module and an infrared transceiver module on the outside wall of the victim's home to implement an attack remotely.

3 HOME-LIMITED CHANNEL

In this section, we first give the definition of the home-limited channel (HLC) and further elaborate properties that required for it. Then, we introduce the boundary-attenuated model to demonstrate the security of HLCs theoretically. Finally, we present three HLC candidates—infrared, ultrasound and modulated visible light (MVL), and describe them in detail.

3.1 Definition and Properties

We define a *home-limited channel* as the channel of which the signal transmission range is within a home. For instance, signals transmitted over the inside infrared channel cannot be detected outside, since infrared cannot penetrate the boundary (e.g., walls and doors) of a house. To achieve adequate security and usability, the following properties should be considered.

- **Boundary-attenuated** is the major property that ensures the security of HLCs. To prevent local outside attackers from eavesdropping or injecting messages over HLCs, the signals over HLCs should be intensely attenuated when propagating through the boundary of a residence, e.g., walls and doors. Thus it's difficult for local outside attackers to launch replay, message-forgery, or MiTM attacks.
- **Imperceptible.** The message transmitted over HLCs should be transparent to users, which means the transmission signals are supposed to be inaudible and unobservable. For example, infrared signals are invisible while ultrasound ones are inaudible. This also helps to defend local outside attackers, who can stay close to the residence and have a chance to peek inside the house through windows.
- **Lightweight and Energy efficient**. On the one hand, the data traffic over HLCs should be lightweight, since numerous smart home devices are resource-constrained. On the other hand, the transmission process over HLCs should be energy efficient. Moreover, the extra-hardware of HLC sensors should be low-cost and easy to install.

3.2 Boundary-attenuated Model

To further understand the security of HLCs, we introduce the boundary-attenuated model of the HLCs [41]. This model can help us theoretically answer the following question: why and how HLCs can reduce the success rate of various attacks, such as eavesdropping and spoofing?

Let H denote the fading process while $X[n]$, $W[n]$, and $Y[n]$ denote the time-domain samples of the transmitted signal, ambient noises, and received signal at the nth sample point, respectively. Given a sampling period of $n = 1 \ldots N$, the detection problem can be formulated as a binary hypothesis testing problem as follows:

$$\begin{aligned} \mathcal{H}_0 &: Y[n] = W[n], \\ \mathcal{H}_1 &: Y[n] = HX[n] + W[n], \end{aligned} \tag{1}$$

where $\mathcal{H}_0$ indicates there is no transmission and the received signal is purely noise throughout the sampling period, and $\mathcal{H}_1$ indicates the received signal contains a message. Our detection algorithm can be modeled as a random function F: $\mathbb{R}^N \rightarrow \{0,1\}$, where F maps the N dimensional received vector $\boldsymbol{Y} = (Y[1], Y[2], \ldots, Y[N])$ onto the $\{0,1\}$, where $F = 1$ indicates that a message is detected and $F = 0$ indicates that no message is detected. Here, we define the probability of detection (P_D) as the probability that the average amplitude of received signals is larger than the threshold value γ, i.e., $F = 1$. We define the average signal to noise ratio (SNR) as

$$SNR = \frac{P}{\sigma^2}; P = \frac{1}{N}\sum_{n=1}^{N} X[n]^2, \tag{2}$$

where P is the average signal power and σ^2 is the noise variance. To calculate the P_D, we introduce an energy detector, denote it as $T(\mathbf{Y}) = \frac{1}{N}\sum_{n=1}^{N} Y[n]^2$. According to the central limit theorem, we have the following approximates if the noise variance is known:

$$\begin{aligned} T(\mathbf{Y})|\mathcal{H}_0 &\sim \mathcal{N}\left(\sigma^2, \frac{1}{N}2\sigma^4\right), \\ T(\mathbf{Y})|\mathcal{H}_1 &\sim \mathcal{N}\left(P|H|^2 + \sigma^2, \frac{1}{N}2\left(P|H|^2 + \sigma^2\right)^2\right). \end{aligned} \tag{3}$$

With approximations, we have

$$\begin{aligned} P_D &= Prob(T(\mathbf{Y}) > \gamma|\mathcal{H}_1) = Q\left(\frac{\gamma - (P|H|^2 + \sigma^2)}{\sqrt{\frac{2}{N}}(P|H|^2 + \sigma^2)}\right) \\ &= Q\left(\frac{\frac{\gamma}{\sigma^2} - (SNR|H|^2 + 1)}{\sqrt{\frac{2}{N}}(SNR|H|^2 + 1)}\right) \\ &= \frac{1}{\sqrt{2\pi}\sqrt{\frac{2}{N}}(P|H|^2 + \sigma^2)} \int_{\gamma}^{\infty} e^{-\frac{(t-(P|H|^2+\sigma^2))^2}{\frac{4}{N}(P|H|^2+\sigma^2)^2}} dt, \end{aligned} \tag{4}$$

where $Q()$ is the tail distribution function of the standard normal distribution. By replacing the parameters in Equation (4) with the ones in legitimate and attacking scenarios, we can answer the two questions.

Why eavesdropping is difficult? Successful eavesdropping over HLCs is the pre-requirement for spoofing a command. Given that the signal source P and the environment noise σ^2 are identical for both legitimate receivers inside a trusted home and attackers outside the home as shown

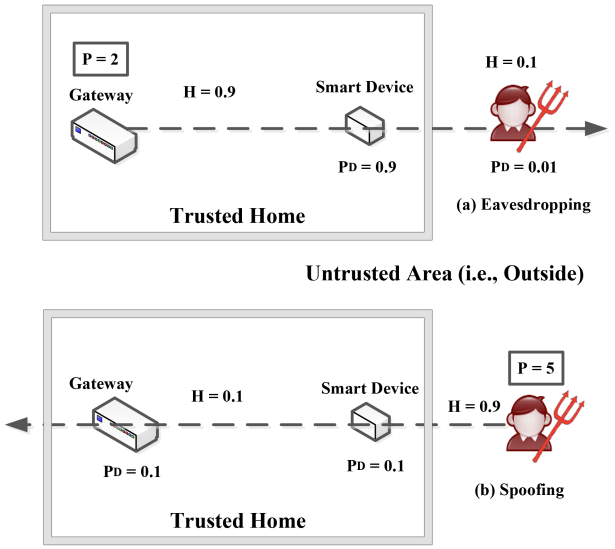


Fig. 3. An illustration of why an attacker is unable to (a) eavesdrop and (b) spoof on HLCs. Because of the high attenuation of the walls (ΔH), an attacker can only detect a HLC message with a probability of $P_D = 0.01$. Despite that an attacker may transmit HLC messages with a higher power ($P = 5$), the gateway can at most receive the message with a probability of $P_D = 0.1$ due to the attenuation (ΔH).

in Figure 3(a), the main difference lies at the fading process H. First, H depends heavily on the transmission medium and will sharply reduce when passing the physical boundaries. Second, according to Equation (4), the smaller H leads to the lower P_D. Therefore, the P_D of local outside eavesdroppers is much lower than the one of legal users due to the boundary attenuation.

Why spoofing is difficult? To spoof a command, a local outside attacker has to transmit a message over HLCs in a way that it can be detected by the receiver inside a trusted home. The attacker could increase the transmission power. However, as long as the attenuation of the walls is strong enough, legitimate gateways and smart devices will not be able to detect messages reliably with the decrease of H, as shown in Figure 3(b).

3.3 HLC Candidates

According to the definition and properties of HLCs, we choose three candidates—infrared, ultrasound and modulated visible light (MVL). Then, we give a brief introduction to these three candidates and demonstrate why they meet our requirements.

- **Infrared** (IR) is a type of electromagnetic radiation that falls just outside the visible spectrum. Therefore, it is invisible to normal users. The wavelength of common IR emitters is 940 nm, which makes it reflected by walls and doors rather than penetrating them. Infrared-based communication has been widely used, and some smart devices have been equipped with IR sensors, such as smart TVs and smart cameras.
- **Ultrasound** is sound waves with frequencies higher than the upper audible limit of human hearing ($f > 20$ kHz). When sound travels through a medium, its intensity diminishes with distance and the attenuation is generally proportional to the square of sound frequency [4]. Therefore, the intensity of the ultrasound will sharply decrease when passing through the boundary of the smart home. Moreover, these inaudible sounds are used in many different fields, including imaging and communication.

Table 1. The List of HLC Candidates and the Corresponding Characteristics and Applications

HLCs	Characteristics	Applications
Infrared	invisible, boundary-attenuated	smart TV, smart camera
Ultrasound	inaudible, boundary-attenuated	domestic robot, VR gear
MVL	boundary-attenuated	smart bulb

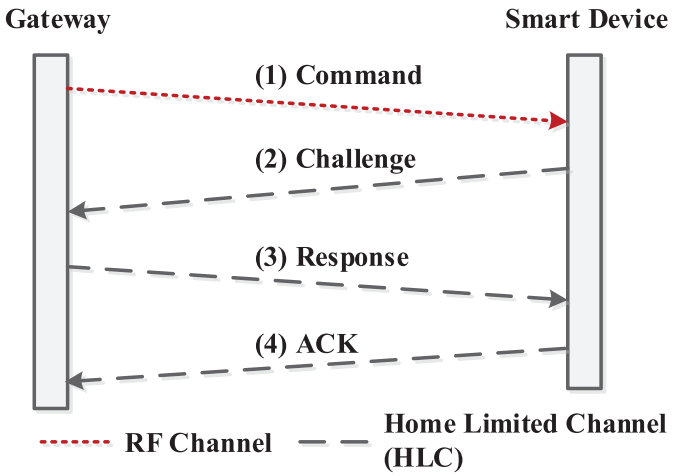


Fig. 4. The overview of HlcAuth. The *command* message is transmitted over the RF channel while the *challenge*, *response*, and *ACK* message is transmitted over the HLC.

- **Modulated Visible Light (MVL)**. Visible light can be observed by human beings in common sense. However, we can modulate the pulse width of the visible light signals to make it above the human eye's resolution so that they are indistinguishable for normal users. With the similar propagation characteristics to infrared, modulated visible light cannot reach the outside of the smart home. In addition, most existing smart home have installed light sensors or light emitters, such as smart bulb.

We summarize the characteristics and typical applications of three candidates and present them in Table 1.

4 DESIGN OF HLCAUTH PROTOCOL

Although the security properties of HLCs can efficiently prevent the smart home from various attacks, it still leaves us two questions: (1) Since numerous smart devices are resource-constrained, how can we implement our scheme in a lightweight way? (2) Given that local outside attackers still have chances to eavesdrop or inject over HLCs, how can we further improve the security of communications between smart devices? To address the two questions, we propose HlcAuth, which exploits a challenge-response mechanism and authenticates smart device communication without encryption. The overview of HlcAuth is shown in Figure 4.

Challenge-Response. We utilize a challenge-response mechanism to realize the mutual authentication between the gateway and smart devices. Smart devices require the gateway to prove its trustworthiness by answering a correct *response*. In similar, the gateway verifies the identity of smart devices by checking the validity of the *challenge*. Both *challenge* and *response* messages are transmitted over HLCs, which largely improves the security of the communication.

Table 2. Notations

Symbol	Definition
HLC	Home-limited channel
HG	Home gateway
SD	Smart device
ID_g	Identity of the home gateway
ID_a	Identity of smart device A
Seq_{cm}	The sequence number of each *command*
Seq_{cl}	The sequence number of each *challenge*
$Token_{cm}$	A unique authentication token for each *command*
$Token_{cl}$	A unique authentication token for each *challenge*
CMD	The command to operate devices
$h()$	One-way hash function
P_{cm}	The package of the command message
P_{cl}	The package of the challenge message
P_{rs}	The package of the response message
P_{ack}	The package of the ack message
T_{gn}	The nth timestamp of the home gateway
T_{dn}	The nth timestamp of smart device A
CRC	Cyclic redundancy check
DS_a	The status of smart device A
$\|\|$	Concatenation operation

Key-free. The main difference between `HlcAuth` and traditional secure protocol is key-free, which means the authentication between smart devices and the gateway does not rely on encryption keys. The security of communication relies on the boundary-attenuated property of HLCs. Without the overhead of encryption keys, resource-constrained smart devices can also achieve high-level security.

4.1 Protocol Design

Here, we describe the detailed protocol of `HlcAuth` and summarize notations in Table 2. `HlcAuth` includes four phases: RF command initiation, HLC challenge, HLC response and command execution. Figure 5 depicts the flowchart of the `HlcAuth` scheme.

4.1.1 Phase I: Command (RF). HG performs the following steps to initiate a standard RF *command*, which is transmitted using the existing HAN protocol (e.g., Zigbee, Z-Wave), to SDs.

S1. HG first generates a unique short authentication token $Token_{cm}$ and then records its current local timestamp T_{g1}. Both are used to prevent replay attacks.
S2. HG sends the *command* message, which includes the $P_{cm} = \{ID_a \,||\, ID_g \,||\, Seq_{cm} \,||\, CMD \,||\, Token_{cm} \,||\, CRC_{cm}\}$ to Device A over RF channels.
S3. HG computes $Q_{cm} = h(ID_a \,||\, Seq_{cm} \,||\, CMD \,||\, Token_{cm})$ and stores the $(Seq_{cm}, Token_{cm}, Q_{cm})$ into its cache.

4.1.2 Phase II: Challenge (HLC). Upon receiving the *command* message from HG, device A sends a *challenge* message to authenticate HG.

S4. Device A generates a unique short random authentication token $Token_{cl}$ and computes $Q_{cm'} = h(ID_a \,||\, Seq_{cm} \,||\, CMD \,||\, Token_{cm})$. At the same time, it records its current local timestamp T_{d1}.

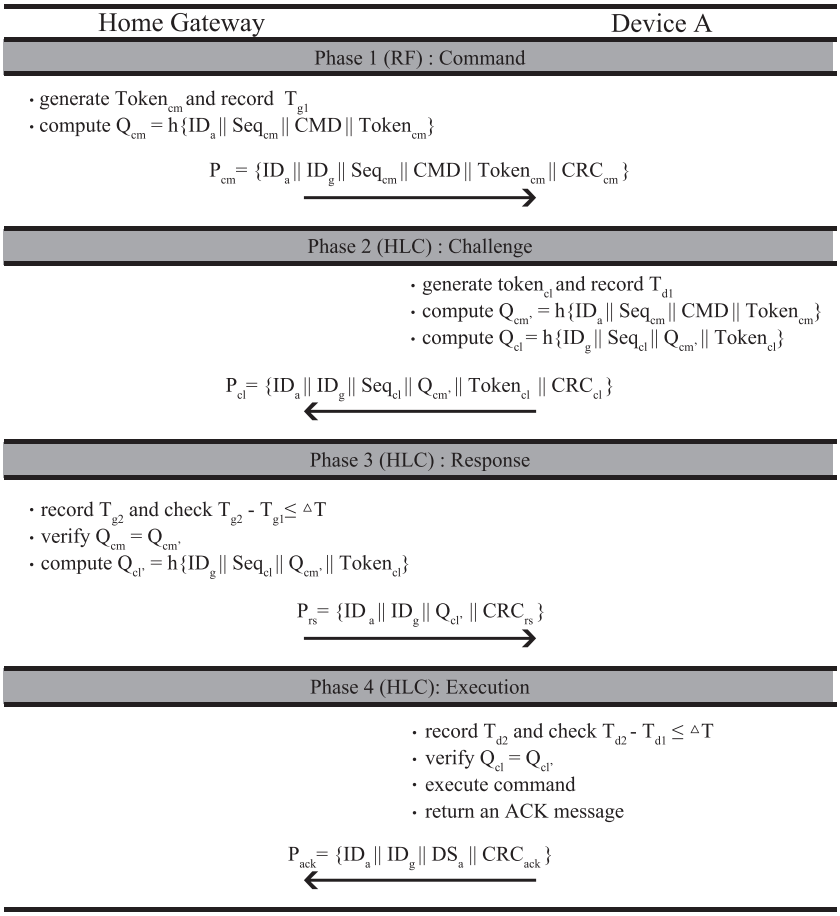


Fig. 5. The control flow of HlcAuth includes four phases: RF command initiation, HLC challenge, HLC response, and command execution.

S5. Device A sends a *challenge* message $P_{cl} = \{ID_a \,||\, ID_g \,||\, Seq_{cl} \,||\, Q_{cm'} \,||\, Token_{cl} \,||\, CRC_{cl}\}$ to HG over HLCs.

S6. Device A computes the hash value $Q_{cl} = h(ID_g \,||\, Seq_{cl} \,||\, Q_{cm'} \,||\, Token_{cl})$ and stores the $(Seq_{cl}, Token_{cl}, Q_{cl})$.

4.1.3 Phase III: Response (HLC). After receiving the *challenge* message from device A, HG first verifies the integrity of the *command* message and then sends the *response* message to device A over HLCs.

S7. Upon receiving the *challenge* message from device A, HG records its current local timestamp T_{g2} and checks whether $(T_{g2} - T_{g1}) \leq \Delta T$. If it holds, then HG retrieves the corresponding Q_{cm} from its cache, else sets $Q_{cl'}$ to zero.

S8. HG verifies whether $Q_{cm'} = Q_{cm}$. If not, then HG sets $Q_{cl'}$ to zero, else it computes the $Q_{cl'} = h(ID_g \,||\, Seq_{cl} \,||\, Q_{cm'} \,||\, Token_{cl})$.

S9. HG sends a *response* message $P_{rs} = \{ID_a \,||\, ID_g \,||\, Q_{cl'} \,||\, CRC_{rs}\}$ to device A over HLCs.

Table 3. HLC Challenge Frame

Segment	ID_a	ID_g	Seq_{cl}	$Q_{cm'}$	$Token_{cl}$	CRC_{cl}
Length	8 bits	4 bits	128 bits	64 bits	128 bits	16 bits

Table 4. HLC Response Frame

Segment	ID_a	ID_g	$Q_{cl'}$	CRC_{rs}
Length	8 bits	4 bits	64 bits	8 bits

Table 5. HLC ACK Frame

Segment	ID_a	ID_g	DS_a	CRC_{ack}
Length	8 bits	4 bits	8 bits	4 bits

4.1.4 Phase IV: Execution (HLC). In this phase, device A verifies the locality of HG and the integrity of the *challenge* message. If the *response* message passes the verification, then device A executes the *command* and returns an ACK message, which includes its status to HG over HLCs.

S10. Upon receiving the *response* message from HG, device A records its current local timestamp T_{d2} and checks whether $(T_{d2} - T_{d1}) \leq \Delta T$. If it holds, then device A retrieves the corresponding Q_{cl} from own cache, else aborts the process.

S11. Device A verifies whether $Q_{cl'} = Q_{cl}$, if yes, then it executes the *command*, else aborts the process.

S12. Device A returns an ACK message $P_{ack} = \{ID_a \,||\, ID_g \,||\, DS_a \,||\, CRC_{ack}\}$ to HG over HLCs.

4.2 Packet Transmission Scheme

Here, we elaborate the packet transmission scheme of `HlcAuth`, which includes the preamble, frame design, and modulation scheme of three HLC candidates.

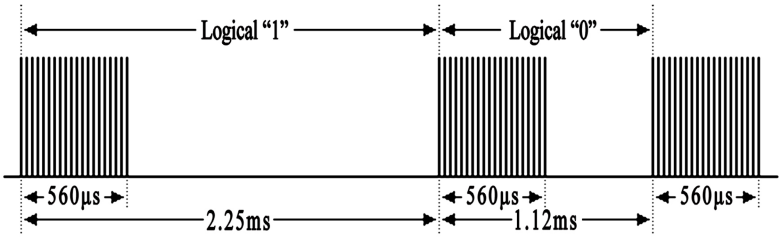
4.2.1 Preamble and Frame Design. A preamble is typically used to synchronize the transmission timing and clock between two or more devices. In our scheme, the preamble mainly serves as the start point of one message, and thus our preamble consists of a 5 ms pulse burst followed by a 1 ms space.

The goal of the frame design is to minimize the overhead while ensuring the integrity of the message. The detailed frame design of the *challenge*, *response*, and *ack* message are summarized in Tables 3, 4, and 5, respectively.

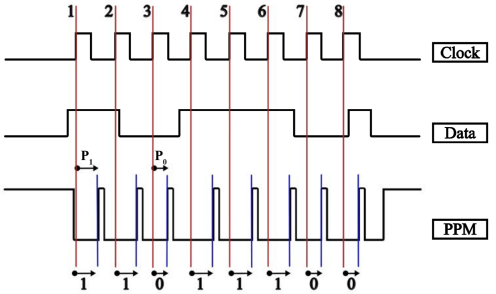
Each *challenge* frame contains the following information: ID_a, ID_g, Seq_{cl}, $Q_{cm'}$, $Token_{cl}$, and CRC_{cl}. In our scheme, we utilize the MD5 algorithm [35] to calculate the hash value and use half of the hash result (64 bits) to reduce the size of the payload. Since the computation complexity of the MD5 is $O(n)$ and the length of each frame is no more than 350 bits, we can properly apply MD5 on resource-constrained devices [39]. To improve the reliability of the transmission of HLC signals, we introduce the cyclic redundancy check (CRC) to detect accidental changes to raw data.

For the *response* and *ack* frame, we remove the identity of the frame and secure token to diminish the overhead. The hash function applied in the *response* and *ack* frame is the same as the *challenge*'s. Since the length of the *ack* frame is limited, we use CRC-4 rather than CRC-8 or CRC-12.

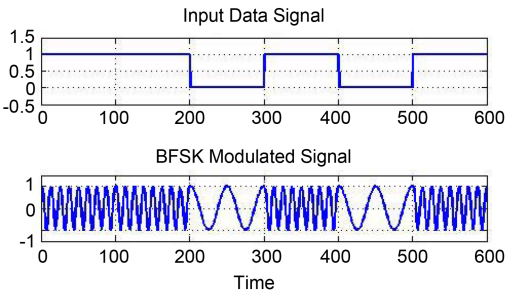
4.2.2 Modulation Scheme. An adequate modulation scheme can improve the accuracy of the transmission and decrease the total transmission time. In `HlcAuth`, we utilize the NEC IR



(a) NEC modulation.



(b) Pulse-position modulation.



(c) Binary Frequency-shift keying.

Fig. 6. Modulation scheme. We utilize the NEC for infrared, Pulse Position Modulation (PPM) for modulated visible light (MVL) and Binary Frequency-shift keying (BFSK) for ultrasound, respectively.

modulation scheme [1] for infrared, Pulse Position Modulation (PPM) [14] for modulated visible light (MVL), and Binary Frequency-shift keying (BFSK) [43] for ultrasound, respectively.

Infrared. The most commonly used modulation scheme for infrared is NEC. As shown in Figure 6(a), the NEC modulation uses pulse distance encoding to encode the message bits. The logical "0" is presented as a 562.5 μs pulse burst followed by a 562.5 μs space while the logical "1" is modulated as a 562.5 μs pulse burst followed by a 1.6875 ms space.

Modulated visible light. Pulse-position modulation (PPM) is a form of signal modulation in which M message bits are encoded by transmitting a single pulse in one of 2^M possible required time shifts. As shown in Figure 6(b), when the data reaches "high level" at the rising edge of the clock, the PPM sets the position of the pulse to p_1 to present the logical "1." Otherwise, the PPM sets the position as p_0 to present the logical "0." Compared to the on-off keying (OOK) modulation, PPM requires a larger signal bandwidth but provides higher power efficiency, which is critical for energy-intensive smart devices [6]. In this article, we set the length of a single pulse as 1 ms, which is far below the resolution of the human eye.

Ultrasound. As shown in Figure 6(c), BFSK is the simplest FSK, which uses a pair of discrete frequencies to transmit binary (0 and 1 s) information. It is robust to movements, reverberations, and noise and has a large tolerance to Doppler shift so that it can transmit ultrasound signal in a long distance [17]. Although the operating frequency of our ultrasound transducer is 40 kHz, it still provides us with a bandwidth of 2 kHz to apply BFSK. The logical "1" is transmitted through 39 kHz while the logical "0" is through 41 kHz.

5 SECURITY ANALYSIS

In this section, we analyze the security of `HlcAuth` against various types of attacks, including replay, message-forgery and man-in-the-middle (MiTM) attacks. We first assume that the message

transmitted over RF channels is transparent to attackers. To achieve a successful attack, local outside attackers need eavesdrop and intercept the communication between smart devices and the home gateway, and then inject or modify packets over HLCs. However, the security properties of HLCs and the challenge-response mechanism of `HlcAuth` make these attacks invalid.

5.1 Replay Attack

For the replay attack scenario, local outside attackers aim to control the smart devices inside. To achieve this, they can intercept the RF *command* packet and further repeat it without modification. However, each valid *command* with a unique $Token_{cm}$ and a Seq_{cm} has timeliness. In the phase of RF command initiation, the home gateway will store the (Seq_{cm}, $Token_{cm}$, Q_{cm}) into its cache, and retrieve them until receiving the *challenge* packet. After verifying the validity of the legal *challenge*, the home gateway will remove the (Seq_{cm}, $Token_{cm}$, Q_{cm}) locally. Therefore, the replayed *command* packet cannot pass the verification, since there is no corresponding (Seq_{cm}, $Token_{cm}$, Q_{cm}) in the home gateway's cache.

5.2 Message-forgery Attack

Assume that a local outside attacker can capture previous legal *command* messages and obtain all possible combinations of the (ID_a, ID_g, CMD). He would intentionally masquerade as a legal home gateway and attempt to control smart devices by sending a fake *command* packet.

According to the challenge-response mechanism of the `HlcAuth`, a potential message-forgery attack requires the following two steps: (*i*) forge an RF *command*; (*ii*) forge an HLC *response*. After sending the fake *command* packet, the attacker will face the following situations.

(1) When device *A* receives the forged *command* packet, it will initiate an HLC *challenge* to the home gateway. Since this *challenge* packet is built on the forged *command*, it will generate an invalid $Q_{cm'}$, which cannot be consistent with any Q_{cm} stored in the home gateway's cache. Thus, the *challenge* message cannot pass the verification, and the device will not execute the forged *command*.

(2) One possible way for the attacker to bypass the failure of the *challenge* check is sending a forged HLC *response* before the home gateway returns the legal one, which includes the termination information. To address this issue, `HlcAuth` sets two barriers to defend such attacks.

- First, each HLC *challenge* packet has a unique Seq_{cl} and $Token_{cl}$, which is used to generate the $Q_{cl'}$ in the subsequent *response* packet. Therefore, the attacker has to eavesdrop the whole *challenge* packet before forging the *response* packet. Furthermore, the success of the attack depends on the integrity of the forged *response* packet transmission. Considering that both eavesdropping and transmitting are over HLCs, local outside attackers have extremely low probability to successfully implement message-forgery attacks.
- Second, the time for completely sending a forged *response* packet is limited. According to our later experiments, the duration of sending a forged *response* packet over HLCs is more than 300 ms while the interval between legal *challenge* and *response* packet is 27 ms. That means the attacker cannot send a forged *response* packet integrally.

The above two points prove that `HlcAuth` can resist the message-forgery attack.

5.3 MiTM Attack

Man-in-the-middle (MiTM) attacks intercept the communication between the home gateway and smart devices and impersonate both parties. We consider MiTM attackers with two types of goals: (1) modify the ID_a of a *command*; (2) modify the CMD of a *command*. Since we assume RF channels are available to attackers, thus it is feasible for them to modify (ID_a, CMD, CRC_{cm}) in RF packets.

Table 6. Device List

Device	Model
Camera	CRC910 camera
Humidifier	KW-JSQ05 mini humidifier
Speaker	SADA D-201 speaker
Table lamp	TEDI DNP 62 LED lamp
*Control board	Ardiuno UNO REV3 Development Board
*WiFi module	ESP8266 WiFi module
*Infrared receiver	HX1838B infrared receiver
*Infrared transmitter	SOURCEKIT 3W 940 nm infrared emitter

The devices with * are the components of our designed module.

To further implement MiTM attacks, the attacker needs five steps: (1) intercept the valid *challenge* packet; (2) eavesdrop the valid *challenge* packet; (3) send a forged *challenge* packet; (4) intercept the valid *response* packet; (5) send a forged *response* packet. In this process, the attacker will face the following challenges:

First, predicting the precise timing of inside events is hard for local outside attackers. Together with extremely low probability to successfully eavesdrop or inject HLCs, it is difficult for them to get or transmit complete packet correctly. Once they make a mistake in any step above, the attack fails. Compared to the message-forgery attack, MiTM attacks need one more HLC transmission, which will further decrease their success rate.

5.4 Jamming Attack

Outside attackers may use extremely large noise to interfere with inside authentication. First, the jamming attacks on radio frequency channels (e.g., jamming the legal RF *command*) are not the focus of this article, and previous work [44, 45] has made much effort to eliminate this threat. Second, the jamming signals over HLCs will be intensely attenuated when passing through the physical boundary, which leads to the failure of the jamming attack. Moreover, we do not consider military-level attackers with powerful tools (e.g., military-grade infrared emitter).

6 IMPLEMENTATION

To illustrate the practicality of `HlcAuth`, we first design and deploy a smart home prototype based on the `HlcAuth` and then discuss the compatibility of implementing our scheme on existing smart home devices.

It takes three steps to build up a smart home prototype, which is described below:

- ***Device Hardware Modification***. Considering that most existing smart home devices do not provide software or hardware debugging interface, we decide to build the prototype based on device modification. We first modify four types of traditional household appliances, including table lamps, speakers, cameras and humidifiers. The detailed device list is presented in Table 6. Then, we design a standardized module that contains an MCU, HLC sensors, and a WiFi module to help devices communicate with the gateway and execute the authentication over HLCs.
- ***Gateway Design***. We design the home gateway by utilizing the MCU with stronger computational capabilities, independent power supply module and the same HLC sensors applied on smart devices. To guarantee the HLC signals can cover the whole room, the gateway is equipped with two sets of HLC sensors on both sides.

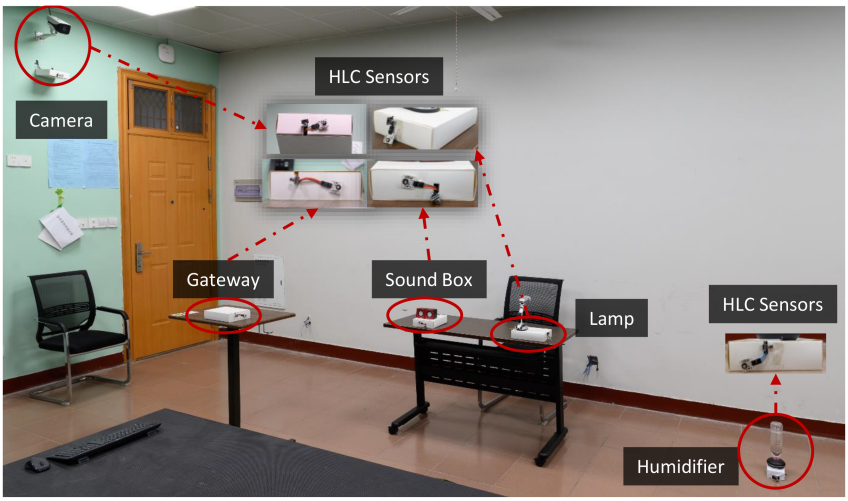


Fig. 7. Smart Home Prototype: single-room scenario.

- ***Software Implementation.*** After the design of smart devices and the gateway, we further write the control program of the `HlcAuth` into their MCU. It is worth to mention that the code for each device is universal, which brings convenience to the deployment of our scheme.

6.1 Prototype Design and Deployment

We implement a smart home prototype based on `HlcAuth` in both single-room and multi-room scenarios.

Single-room scenario. After the above three steps were done, we start to deploy our prototype in a 5 m ∗ 3 m ∗ 5 m room. The demonstrative prototype is shown in Figure 7. All smart devices are placed where they normally are. For instance, table lamps and the humidifier are placed on the desk while the speaker is nearby the computer, and the camera is attached to the wall. To ensure the HLC signals can cover the whole space, the gateway is placed in the center of the room. We utilize infrared as the communication media of the HLC in this prototype due to its relatively high performance, which will be presented in the later evaluation. After the tests, all the smart devices can operate properly based on `HlcAuth`. Even if we move the device 5 m away from the gateway, it still can perform the command correctly.

Multi-room scenario. In addition, we also implement `HlcAuth` for the multi-room scenario, and the prototype is shown in Figure 8. We design smart relays, which are small-size, low cost, and equipped with HLC sensors, and put them in each room. The smart relay is responsible for forwarding the command from the gateway and executing the authentication process. Since both the gateway and the smart relay can afford computationally expensive crypto-operations, the communication between them can use a secure RF channel. For instance, imagine that the gateway wants to control a smart light in another room $\mathcal{R}_2$. The gateway will first send the RF command to the smart relay in $\mathcal{R}_2$ (Room 2). Then the relay in $\mathcal{R}_2$ and the light utilize `HlcAuth` to validate the authenticity of each other, and the relay will return the results to the gateway. Thus, the gateway can control the device in another room with the help of relays.

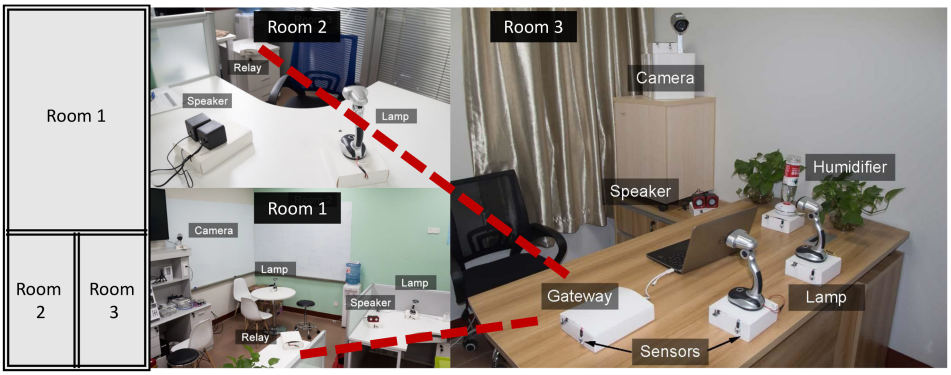


Fig. 8. Smart Home Prototype: multi-room scenario.

6.2 Scheme Compatibility

After illustrating the practicality of our scheme, we further discuss its compatibility with existing smart home devices. Depending on whether the device has HLC sensors, we divide the devices into two categories:

Sensor-integrated devices. Sensor-integrated devices are the future trend of smart home devices, which are equipped with HLC sensors and generally have high computational capabilities. Several existing smart devices fall into this category, including smart TV (infrared sensor) and smart camera (infrared sensor). To implement `HlcAuth` on these devices, manufacturers only need to patch the communication program in the process of production.

Non-sensor-integrated devices. Non-sensor-integrated devices represent most existing smart home appliances, which lack HLC sensors. However, these devices can provide additional I/O ports to equip HLC sensors. Therefore, we can apply our scheme on these devices by hardware modification and program patching. The above smart home prototype provides an example.

7 EVALUATION

In this section, we evaluate the performance of `HlcAuth` from the aspects of usability and security.

7.1 Experiment Setup

We build gateways and smart devices using Arduino UNO REV3 Development Boards with ESP8266 WiFi module. Each device is equipped with a 3 W, 940 nm SOURCEKIT infrared emitter, an HX1838B infrared sensor receiver module, a KY-008 650 nm red MVL transmitter, a GY-485-44009 RS485 light intensity sensor, and an HC-SR04 ultrasound transducer. The experiment setup is shown in Figure 9.

Evaluation Metrics. We introduce true positive rate (TPR) and false positive rate (FPR) to evaluate the usability and security of our scheme individually:

$$\begin{aligned} TPR &= \frac{TP}{TP + FN}, \\ FPR &= \frac{FP}{FP + TN}. \end{aligned} \tag{5}$$

To quantify usability, TPR is used, which depends on true positive (TP) and false negative (FN). TP denotes the number of *commands* sent by the gateway that are correctly executed, while FN represents the number of *command* that the device doesn't execute as expected. We utilize FPR to quantify security, and FPR depends on false positive (FP) and true negative (TN). FP denotes

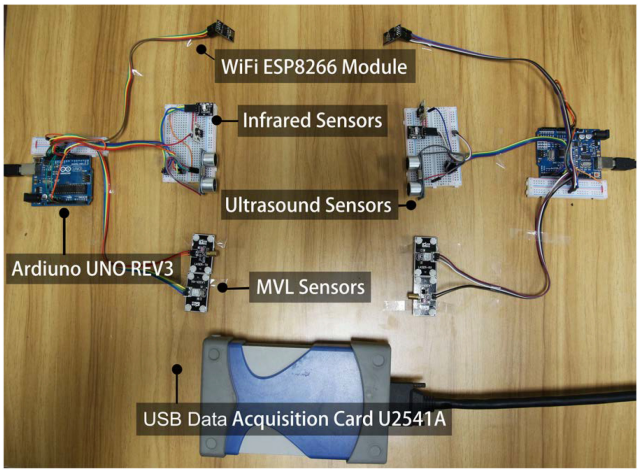


Fig. 9. Experiment setup.

the number of the forged *commands* from the attacker that were executed, while TN denotes the number of forged commands being rejected successfully.

7.2 Usability

To measure the usability of `HlcAuth`, we first evaluate the performance of HLC candidates from the aspects of distances and angles, and then we present the overhead of time consumption, energy consumption, and the cost of our scheme.

7.2.1 Performance of HLC Candidates. According to Section 3, We choose three HLC candidates—infrared, ultrasound and modulated visible light (MVL). To measure the usability of `HlcAuth`, we evaluate the performance of three candidates from the aspects of distances and angles.

Distance: We measure the TPR of infrared, ultrasound and MVL by varying the distances from 0 to 5 m. The device and the gateway are placed facing each other. Particularly, we define the angle between the signal emitter and the receiving sensor is 0 in this situation. We test the TPR once every 20 cm, and we send the command 100 times at each point. The results are shown in Figure 10. Both infrared and MVL can achieve 100% TPR within 4.2 m, and will degrade slightly at a further distance. It is worth mentioning that the TPR of infrared and MVL is 97% and 95%, respectively, even when the distance reaches 5 m. However, ultrasound can only achieve 100% TPR at a distance between 0.6 and 2.2 m, and the TPR will decline to 0 out of this range. The ultrasound cannot succeed in such a short distance, because it is modulated by FSK. The signal is distorted and cannot be demodulated at a further distance. Considering the application scenarios (e.g., a large living room), the range of ultrasound is insufficient to guarantee a high TPR. Thus, we cannot use ultrasound to transmit HLC.

Angle: Given that the transceiver of the device and the gateway is not always facing each other, we evaluate the impact of angles on the TPR. We test the performance of three candidates by placing the device and the gateway at an angle from 0° to 180° at 2 m at an interval of 15°, and each HLC is tested 100 times at each point. The results are presented in Figure 11. Both infrared and ultrasound can achieve relatively high TPR within 45°, while MVL can only communicate at 0° because of its high directionality. As the angle increases, the TPR of ultrasound decreases to 0 at 75° while the TPR of infrared decreases slower and reaches the minimum (25%) at 120°.

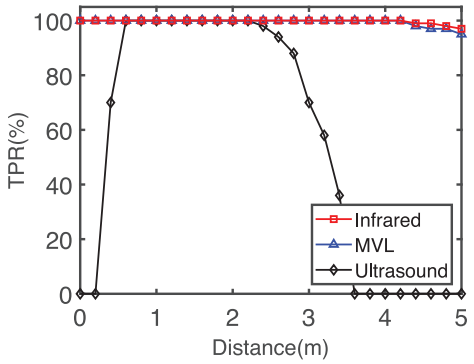


Fig. 10. The TPR of infrared, ultrasound, and MVL-based HlcAuth at various distances. Both infrared and MVL can achieve a TPR of 100% within 4.2 m and 95% at 5 m, while ultrasound can only achieve a TPR of 100% between 0.6 and 2.2 m.

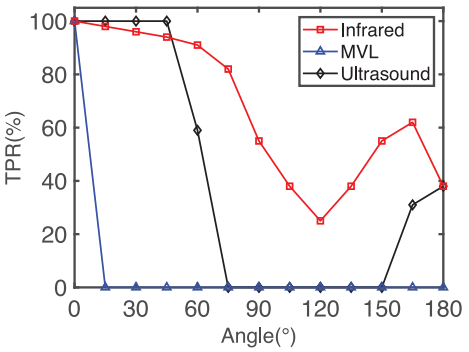


Fig. 11. The TPR of infrared, ultrasound, and MVL-based HlcAuth at various angles at 2 m. Both infrared and ultrasound can achieve over a TPR of 94% within 45° while MVL is only available at 0°.

Table 7. The Time Consumption of Executing the Entire HLC

HLCs	t_h	t_{cl}	t_{rs}	t_{ack}	t_s
Infrared	3 ms	788 ms	195 ms	64 ms	1,053 ms
Ultrasound	3 ms	3,480 ms	840 ms	280 ms	4,606 ms
MVL	3 ms	688 ms	168 ms	55 ms	917 ms

Both using infrared and MVL are no more than 1,053 ms, while using ultrasound is 4,606 ms due to limited bandwidth.

Interestingly, when the angle is approaching 180°, there is a small rise as a result of the reflection of the wall.

In conclusion, infrared is the best HLC candidate due to its relative high TPR at a long distance and a large range of angle, while ultrasound and MVL are both deficient due to attenuation or high directionality.

7.2.2 Time Consumption. Here, we measure the time overhead introduced by HlcAuth. In this experiment, we calculate the transmission duration by utilizing the serial port to record the starting time and the end time of the transmission. The time consumption includes the following factors:

$$Time\ Consumption: t_s = 2 * t_h + t_{cl} + t_{rs} + t_{ack} + t_\sigma, \tag{6}$$

where the t_h represents the time of executing the MD5 hash function once, t_{cl}, t_{rs} and t_{ack} denote the average duration of sending a *challenge*, a *response*, and a *ACK* message, respectively. t_σ is the total time of all other operations. Moreover, t_σ is negligible in this experiment. Table 7 summarizes the measurements.

The results show that the time consumption of using infrared and MVL is no more than 1053 ms, and the duration of transmission dominates the time overhead. However, the time consumption of using ultrasound is 4, 606 ms, which is limited by the bandwidth of the two different frequencies of ultrasound.

7.2.3 Energy Consumption. We measure the amount of energy consumed by executing HlcAuth in the Arduino UNO REV3 Development Board, i.e., the platform we used for implementation. A USB data acquisition (DAQ) card U2541A is utilized for obtaining the operation current, which is

Table 8. The Average Energy Consumption of Receiving or Sending 1 Byte

Signal	Infrared		Ultrasound		Laser	
Mode	Receive	Send	Receive	Send	Receive	Send
Energy	86.3 μJ	2,520 μJ	800 μJ	6,000 μJ	78.63 μJ	3,050 μJ

The consumption of ultrasound is higher than others due to the low bit rate caused by the limited bandwidth.

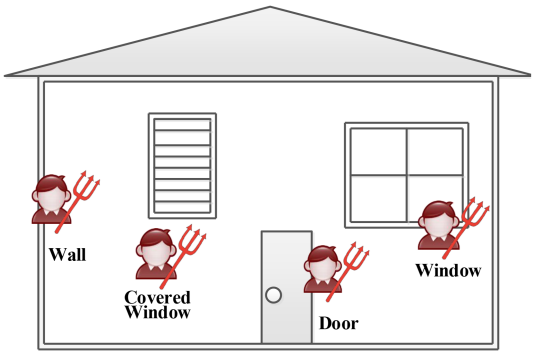


Fig. 12. Experiment setup with attackers appearing at four possible locations: the wall, the covered window, the closed door, and the glass window.

used to calculate the energy consumption. Table 8 shows the average energy consumption when a single 1-byte HLC packet is received or sent by the smart device. The average energy consumption of receiving 1 byte of both infrared and MVL signals are less than 100 μJ, while sending requires a little more than 2 mJ, which is about half of the WiFi transmission [12]. The ultrasound costs 800 μJ for receiving a bit, and 8,000 μJ for sending a bit. As such, the total energy consumption of executing a complete `HlcAuth` process is less than 3.1 J for smart devices, which has little effect on their work cycles (almost tens of kJ).

7.2.4 Cost Analysis. The cost of implementing `HlcAuth` is mainly composed of the cost of extra hardware. The infrared, ultrasound and MVL sensor modules utilized in our experiments and the prototype do not exceed $1, $2 and $2 individually. This increment is lower than the cost of MCU upgrade [30].

7.3 Security

Though we have proved the security enhancement that HlcAuth can provide in Section 5, we nonetheless validate the scheme experimentally from the perspective of attenuation test and time limitation.

7.3.1 Attenuation Test. We assume that local outside attackers can locate at any position outside the house, and let malicious gateways to mimic them who try to send a forged command or modify the legal command. As shown in Figure 12, we select four positions in this experiment: an attacker stays behind (1) a 6-cm-thick metal door, (2) a 30-cm wall, (3) a covered window, and (4) a 3-cm-thick single-glass window, respectively. The HLC signal emitters of attackers are attached at the outside surface of each boundary. Then, we move the indoor receivers (i.e., HLC sensors) from 0 to 5 m away from the boundary at a 20 cm step. The outside emitters and the inside receivers are facing each other.

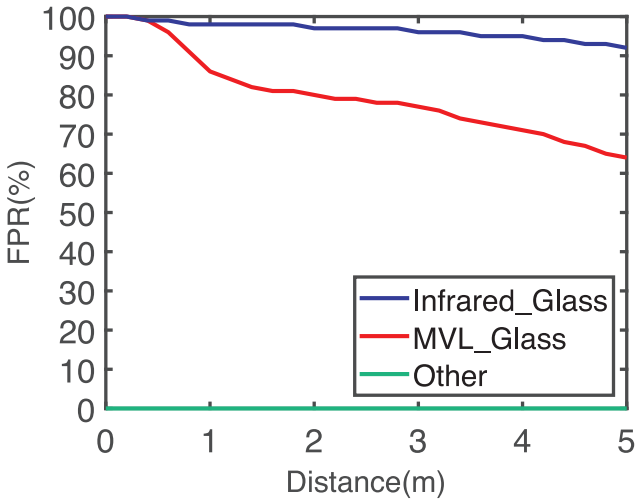


Fig. 13. The attenuation evaluation of four physical boundaries for three HLC signals. Other includes Infrared_Door, Infrared_Wall, Infrared_Covered, MVL_Door, MVL_Wall, MVL_Covered, Ultrasound_Glass, Ultrasound_Door, Ultrasound_Wall, Ultrasound_Covered.

We perform the attenuation test of various physical boundaries for three candidate HLC signals. For each case (e.g., MVL_Glass, $Infrared\_Glass$, $Ultrasound\_Wall$), outside HLC signal emitters send a 100-bit message to inside receivers and repeat 100 times at each distance. We calculate the average FPRs and present the results in Figure 13. As mentioned in Section 3.2, lower H leads to lower FPR. Except for MVL_Glass and $Infrared\_Glass$, the FPRs are 0, which means that attackers cannot successfully send or receive packets reliably when locating outside the boundary of the smart home. Notably, the high-power infrared and MVL signals can have a chance to penetrate the glass window. In these cases, local outside attackers may exploit the vulnerabilities of boundary-attenuated property to apply OOB channel-based attacks. However, attackers still fail to conduct these attacks successfully due to the following constrained timing property of the challenge-response mechanism.

7.3.2 Constrained Timing. Recall from Section 5, the constrained timing means that local outside attackers don't have sufficient time to completely transmit the forged *response* message before the home gateway starts to send the legal one. Otherwise, the attack will fail. To validate this point, we measure and further compare the following two time durations.

- ***Transmission time*** Δt_1**.** The time duration of completely sending a single *response* message by local outside attackers.
- ***Process time*** Δt_2**.** The time duration between the home gateway receiving the *challenge* message and starting to send the legal *response* message.

By calling the internal timer of our system and performing 100 times test, we calculate the average transmission time and process time. Results show that $\bar{\Delta t_1}$ is 330 ms, which is much longer than the 27 ms $\bar{\Delta t_2}$. Therefore, attacks fail, since the HLC receiver cannot demodulate successfully when the legal message and the forged message collide. The results show that the FPR is 0 when using infrared, ultrasound or MVL to build `HlcAuth`.

8 RELATED WORK

8.1 Smart Home Security

Current smart home security analyses focus on three aspects: devices, communication protocols and applications (Apps). First, Denning et al. outlined a set of emergent threats to smart homes due to the vulnerability of smart devices [5]. Notra et al. [33] dissect the behavior of three household devices, including the Phillips Hue light bulb, the Belkin WeMo power switch and the Nest smoke-alarm. The results show that these devices can be compromised. Reports [8, 16] present the smart home devices can be easily hacked, since manufacturers have not considered security and privacy as a design priority. Second, the communication protocols applied in the smart home was found insecure. Molina [31] utilized the KNX package flaws in the protocol for the remote control on a HA system. Recently, Fernandes et al. [7] presented a set of security analysis on Samsung-owned *SmartThings* smart home programming platform and discovered two intrinsic design flaws that lead to significant overprivilege in SmartApps. Due to the attacks, we need to prevent the smart home devices from being compromised, make the communication more secure, and detect the potential flaws in applications at the software level. Our article focuses on building a more secure communication.

To improve the security of the communication in smart home systems, existing work focuses on building up a lightweight authentication scheme between pairs of smart home devices. Kumar et al. [22] used a short authentication token and established a secure session key to reduce the cost of the public key operations. However, the system in this scheme requires third-party service providers involvement. And the secure information that used to produce a session key has to be stored in home devices in advance. Li et al. [25] proposed that each node get private/public key pair from a certificate agent(CA) over an OOB channel and then carry out an authenticated key exchange protocol. However, this work does not include the implementation and no security analysis on OOB data distribution provided. Different from the above schemes, we propose a secure and lightweight communication protocol based on home-limited OOB channels with minimal additional cost. Particularly, we evaluate the security of `HlcAuth` from the aspects of theoretical analysis and real-world experiments. Moreover, we implement our scheme in both single-room and multi-room scenarios.

8.2 Proximity-based Authentication

Proximity-based user authentication has been playing an increasingly critical role for the scenarios, where the service provider grants access to the objects within a given area (e.g., a room). The majority of previous work can be divided into two categories: context-based and OOB channel based. First, context-based authentication leverage shared physical context to authenticate co-located devices. Amigo [42], Ensemble [19], and Proximate [28] utilize the time-varying radio environment as proof-of-proximity to generate a shared key and form secure associations between devices. Bardram et al. [3] use the user's activities to verify the object's location. Han et al. [15] exploit inter-event times to implement device pairing across heterogeneous sensing types. Second, Out-of-band (OOB) Channel is an independent transmission channel from a defined telecommunications frequency band between a pair of connected stream sockets. Typically, OOB channels are used to for device pairing [2, 10, 29, 36, 38, 46] at bootstrap phase as well as provide alternatives to traditional forms of communication [34]. According to the physical channel that signals communicate over, OOB channels can be categorized into acoustic [10, 46], light [36], seismic, magnetic, thermal, and movement [29]. Two works [20, 21] gave overviews of previous work on OOB channels from the aspects of theory and practice, respectively. Traditional device pairing methods based on OOB channels are generally considered as secure. However, Halevi et al. [13] demonstrate the

feasibility of eavesdropping over acoustic OOB channels, including IMD pairing, PIN-Vibra and BEDA. This work presents the potential vulnerability of OOB channels, which should be taken into account in our protocol design. In our article, we utilize the boundary-attenuated property of OOB channels, together with a challenge-response mechanism, to address the limitations of previous work. Notably, our scheme can effectively defend against local outside attacks based on OOB channels.

We have reported an initial work on this topic in Reference [24]. This article is significantly enhanced in both depth and completeness. We highlight the following major differences: (a) We implement `HlcAuth` in both single-room and multi-room scenarios to illustrate the practicality of our scheme. (b) We provide a theoretical model to illustrate the security property of home-limited channels. (c) We add the description of the `HlcAuth` packet transmission scheme with respect to the preamble and the modulation scheme. (d) We greatly extend our experiments in terms of performance and cost, including the time overhead and the amount of energy consumed by executing `HlcAuth` in a resource-constrained platform.

9 CONCLUSION

We designed and implemented a novel solution to enhance the security of existing smart home–`HlcAuth`. In particular, we introduced Home-Limited Channel (HLC), of which the signal transmission range is limited to the home. Based on the boundary-attenuated property of HLCs, we utilized a challenge-response mechanism to realize the mutual authentication between the gateway and smart devices without a key. The security analysis revealed that the `HlcAuth` can defend replay attacks, message-forgery attacks, and man-in-the-middle (MiTM) attacks. The further experiments showed that our scheme can satisfy the usability (e.g., 100% TPR within 4.2 m, low time and energy consumption, and low cost) for the in-home devices while being resilient against various attacks conducted by local outside attackers (0% FPR). `HlcAuth` is also a viable alternative for lightweight authentication as it does not require a secure key and human involvement.

REFERENCES

[1] Altium. 2017. NEC Infrared Transmission Protocol. Retrieved from http://techdocs.altium.com/display/FPGA/NEC+Infrared+Transmission+Protocol.

[2] Dirk Balfanz, Diana K. Smetters, Paul Stewart, and H. Chi Wong. 2002. Talking to strangers: Authentication in ad-hoc wireless networks. In *Proceedings of the Network and Distributed System Security Symposium (NDSS'02)*.

[3] Jakob E. Bardram, Rasmus E. Kjær, and Michael Ø. Pedersen. 2012. Context-aware user authentication—Supporting proximity-based login in pervasive computing. In *Proceedings of the International Conference on Ubicomp: Ubiquitous Computing*.

[4] NDT Resource Center. 2019. *Attenuation of Sound Waves*. Retrieved from https://www.nde-ed.org/EducationResources/CommunityCollege/Ultrasonics/Physics/attenuation.htm.

[5] Tamara Denning, Tadayoshi Kohno, and Henry M. Levy. 2013. *Computer Security and the Modern Home*. ACM, 94–103.

[6] Hany Elgala, Raed Mesleh, and Harald Haas. 2011. Indoor optical wireless communication: Potential and state-of-the-art. *IEEE Commun. Mag.* 49, 9 (2011), 56–62.

[7] Earlence Fernandes, Jaeyeon Jung, and Atul Prakash. 2016. Security analysis of emerging smart home applications. In *Proceedings of the IEEE Conference on Security & Privacy*. 636–654.

[8] D. Fisher. 2015. Pair of Bugs Open Honeywell Home Controllers Up to Easy Hacks. Retrieved from https://threatpost.com/pair-of-bugs-open-honeywell-home-controllers-up-to-easy-hacks/113965/.

[9] Behrang Fouladi and Sahand Ghanoun. 2013. Security evaluation of the Z-wave wireless protocol. *Black Hat* 24 (2013), 1–2.

[10] Michael T. Goodrich, Michael Sirivianos, John Solis, Gene Tsudik, and Ersin Uzun. 2006. Loud and clear: Human-verifiable authentication based on audio. In *Proceedings of the IEEE International Conference on Distributed Computing Systems*. 10–10.

[11] Slawomir Grzonkowski and Peter M. Corcoran. 2011. Sharing cloud services: User authentication for social enhancement of home networking. *IEEE Trans. Consumer Electron.* 57, 3 (2011), 1424–1432.

[12] XiangFa Guo, Mobashir Mohammad, Sudipta Saha, Mun Choon Chan, Seth Gilbert, and Derek Leong. 2016. PSync: Visible light-based time synchronization for Internet of Things (IoT). In *Proceedings of the 35th Annual IEEE International Conference on Computer Communications (INFOCOM'16)*. IEEE, 1–9.

[13] Tzipora Halevi and Nitesh Saxena. 2013. Acoustic eavesdropping attacks on constrained wireless device pairing. *IEEE Trans. Info. Forensics Secur.* 8, 3 (2013), 563–577.

[14] Jon Hamkins. 2007. Pulse position modulation. *Handbook of Computer Networks: Key Concepts, Data Transmission, and Digital and Optical Networks*, Vol. 1. Wiley, 492–508.

[15] Jun Han, Albert Jin Chung, Manal Kumar Sinha, Madhumitha Harishankar, Shijia Pan, Hae Young Noh, Pei Zhang, and Patrick Tague. 2018. Do you feel what I hear? Enabling autonomous IoT device pairing using different sensor types. In *Do You Feel What I Hear? Enabling Autonomous IoT Device Pairing Using Different Sensor Types*. IEEE.

[16] Arik Hesseldahl. 2015. A hackers-eye view of the Internet of Things. Retrieved from https://www.vox.com/2015/4/7/11561182/a-hackers-eye-view-of-the-internet-of-things.

[17] Sverre Holm, Ole B. Hovind, Svein Rostad, and Rune Holm. 2005. Indoors data communications using airborne ultrasound. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, Vol. 3. IEEE, iii–957.

[18] Sławomir Jasek. 2016. Gattacking Bluetooth smart devices. In *Proceedings of the Black Hat USA Conference*.

[19] Andre Kalamandeen, Adin Scannell, Eyal de Lara, Anmol Sheth, and Anthony LaMarca. 2010. Ensemble: Cooperative proximity-based authentication. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*. ACM, 331–344.

[20] Alfred Kobsa, Rahim Sonawalla, Gene Tsudik, Ersin Uzun, and Yang Wang. 2009. Serial hook-ups: A comparative usability study of secure device pairing methods. In *Proceedings of the Symposium on Usable Privacy and Security*. 10.

[21] Arun Kumar, Nitesh Saxena, Gene Tsudik, and Ersin Uzun. 2009. A comparative study of secure device pairing methods. *Pervas. Mobile Comput.* 5, 6 (2009), 734–749.

[22] Pardeep Kumar, Andrei Gurtov, Jari Iinatti, Mika Ylianttila, and Mangal Sain. 2016. Lightweight and secure session-key establishment scheme in smart home environments. *IEEE Sensors J.* 16, 1 (2016), 254–264.

[23] Andrew Laughlin. 2017. Could your smart home be hacked? Retrieved from https://www.which.co.uk/news/2017/06/could-your-smart-home-be-hacked/.

[24] Chaohao Li, Xiaoyu Ji, Xinyan Zhou, Juchuan Zhang, Jing Tian, Yanmiao Zhang, and Wenyuan Xu. 2018. HlcAuth: Key-free and secure communications via home-limited channel. In *Proceedings of the Asia Conference on Computer and Communications Security*. ACM, 29–35.

[25] Yue Li. 2013. Design of a key establishment protocol for smart home energy management system. In *Proceedings of the 5th International Conference on Computational Intelligence, Communication Systems and Networks*. 88–93.

[26] Zhen Ling, Junzhou Luo, Yiling Xu, Chao Gao, Kui Wu, and Xinwen Fu. 2017. Security vulnerabilities of Internet of Things: A case study of the smart plug system. *IEEE Internet Things J.* 4, 6 (2017), 1899–1909.

[27] Hongbo Liu, Zhenhua Liu, Yingying Chen, and Wenyuan Xu. 2011. Localizing multiple jamming attackers in wireless networks. In *Proceedings of the International Conference on Distributed Computing Systems*.

[28] Suhas Mathur, Robert Miller, Alexander Varshavsky, Wade Trappe, and Narayan Mandayam. 2011. Proximate: Proximity-based secure pairing using ambient wireless signals. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*. ACM, 211–224.

[29] R. Mayrhofer and H. Gellersen. 2009. Shake well before use: Intuitive and secure pairing of mobile devices. *IEEE Trans. Mobile Comput.* 8, 6 (2009), 792–806.

[30] MICROCHIP. 2019. *MCU Price*. Retrieved from https://www.microchip.com/paramchartsearch/Chart.aspx?branchID=1005.

[31] Jesus Molina. 2014. Learn how to control every room at a luxury hotel remotely: The dangers of insecure home automation deployment. *Black Hat USA* (2014), 13.

[32] Icontrol Networks. 2015. 2015 State of the Smart Home Report. Retrieved from https://www.slideshare.net/iangertler/2015-state-of-the.

[33] Sukhvir Notra, Muhammad Siddiqi, Hassan Habibi Gharakheili, Vijay Sivaraman, and Roksana Boreli. 2014. An experimental study of security and privacy risks with emerging household appliances. In *Proceedings of the IEEE Conference on Communications and Network Security (CNS'14)*. IEEE, 79–84.

[34] Adrian Perrig and Dawn Song. 1999. Hash visualization: A new technique to improve real-world security. In *Proceedings of the International Workshop on Cryptographic Techniques and E-Commerce*. 131–138.

[35] Ronald Rivest. 1992. The MD5 message-digest algorithm. Retrieved from https://dl.acm.org/doi/pdf/10.17487/RFC1321.

[36] N. Saxena, J. E. Ekberg, K. Kostiainen, and N. Asokan. 2006. Secure device pairing based on a visual channel. In *Proceedings of the IEEE Symposium on Security and Privacy*.

[37] Jatinder Singh, Thomas Pasquier, Jean Bacon, Hajoon Ko, and David Eyers. 2016. Twenty security considerations for cloud-supported Internet of Things. *IEEE Internet Things J.* 3, 3 (2016), 269–284.

[38] Frank Stajano. 1999. The resurrecting duckling. In *Proceedings of the International Workshop on Security Protocols*. Springer, 183–194.
[39] Yang Su, Yansong Gao, Omid Kavehei, and Damith C. Ranasinghe. 2019. Hash functions and benchmarks for resource constrained passive devices: A preliminary study. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 1020–1025.
[40] Symantec. 2015. Insecurity in the Internet of Things. Retrieved from https://pdfs.semanticscholar.org/6d7f/60b16adead96aafa9e975207980eb32671b5.pdf.
[41] Rahul Tandra and Anant Sahai. 2008. SNR Walls for signal detection. *IEEE J. Select. Top. Signal Process.* 2 (03 2008), 4–17. DOI: https://doi.org/10.1109/JSTSP.2007.914879
[42] Alex Varshavsky, Adin Scannell, Anthony LaMarca, and Eyal De Lara. 2007. Amigo: Proximity-based authentication of mobile devices. In *Proceedings of the International Conference on Ubiquitous Computing*. Springer, 253–270.
[43] Bob Watson. 1980. FSK: Signals and demodulation. *Watkins-Johnson Co. Tech–notes* 7, 5 (1980).
[44] Wenyuan Xu, Ke Ma, Wade Trappe, and Yanyong Zhang. 2006. Jamming sensor networks: Attack and defense strategies. *IEEE Netw.* 20, 3 (2006), 41–47.
[45] Wenyuan Xu, Wade Trappe, Yanyong Zhang, and Timothy Wood. 2005. The feasibility of launching and detecting jamming attacks in wireless networks. In *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 46–57.
[46] Bingsheng Zhang, Qin Zhan, Si Chen, Muyuan Li, Kui Ren, Cong Wang, and Di Ma. 2014. PriWhisper: Enabling keyless secure acoustic communication for smartphones. *IEEE Internet Things J.* 1, 1 (2014), 33–45.

Received February 2019; revised February 2020; accepted May 2020