On Using Camera-based Visible Light Communication for Security Protocols

Wen-Yi Chu
National Taiwan University
Taipei, Taiwan
b05902037@ntu.edu.tw

Ting-Guang Yu
National Taiwan University
Taipei, Taiwan
b05902078@ntu.edu.tw

Yu-Kai Lin
National Taiwan University
Taipei, Taiwan
b99901105@csie.ntu.edu.tw

Shao-Chuan Lee
National Taiwan University
Taipei, Taiwan
r05922001@csie.ntu.edu.tw

Hsu-Chun Hsiao
National Taiwan University
Taipei, Taiwan
hchsiao@csie.ntu.edu.tw

***Abstract*—In security protocol design, Visible Light Communication (VLC) has often been abstracted as an ideal channel that is resilient to eavesdropping, manipulation, and jamming. Camera Communication (CamCom), a subcategory of VLC, further strengthens the level of security by providing a visually verifiable association between the transmitter and the extracted information. However, the ideal security guarantees of visible light channels may not hold in practice due to limitations and tradeoffs introduced by hardware, software, configuration, environment, etc. This paper presents our experience and lessons learned from implementing CamCom for security protocols. We highlight CamCom's security-enhancing properties and security applications that it enables. Backed by real implementation and experiments, we also systematize the practical considerations of CamCom-based security protocols.**

I. Introduction

Visible light communication (VLC) is a wireless communication technique that uses visible light for data transmission. Compared to radio-frequency (RF) communication, VLC is favored for its ubiquity of transmitting devices and wide spectrum, as well as inherent security. The inherent security improves upon VLC's fundamental characteristics, including directional propagation, human visibility, and spatial confinement. Hence, VLC is commonly assumed to be an *ideal* channel that is resilient to eavesdropping [3] and jamming [9], and can easily detect manipulation of transmitted data [3], under the assumption that any attempt to manipulate the transmission will be easily detected by human observers.

Camera Communication (CamCom) is a type of VLC using a camera, rather than a photodiode, as the receiver. Besides inheriting the advantages of VLC, CamCom is able to receive from multiple light sources simultaneously, enabling communication with multiple devices at the same time. Moreover, CamCom can strengthen VLC's inherent security by supporting physical-visual association, preventing impersonation of legitimate devices. That is, the user can visually or programmically bind a piece of transmitted information to the physical presence of the transmitting device through pixels that not only encode transmitted information but also indicate the physical location of the transmitting device.

Although CamCom offers new opportunities for security applications, there has been insufficient investigation of practical considerations and challenges in this context. Our experience suggests that the ideal security guarantees of visible light channels may not hold in practice due to limitations and tradeoffs introduced by hardware, software, configuration, environment, etc. Among the few related studies, [9] provides an overview of such challenges with VLC. However, the study is not backed by real experiments. Other studies [3] [8] focus on the security of the physical layer. Moreover, most of the research considers VLC in general, and does not explore additional opportunities and challenges brought by CamCom.

In this paper, we present our experience and lessons learned from implementing a CamCom scheme for security protocols. To provide a broader perspective for future research, we first summarize CamCom's security-enhancing properties and explain how these properties can be used to simplify existing security protocols or even enable new ones. For instance, CamCom can be used to securely bootstrap multiple devices simultaneously, as well as bind a piece of information to its originating location.

We then systematize these practical considerations with respect to hardware, software, and environment, backed by real implementation and experiments. Specifically, we implement a secure multi-device pairing protocol using CamCom as an out-of-band (OOB) channel, and experimentally demonstrate the influence of hardware (e.g., frame per second, RF channel), software and configuration (e.g., camera configuration, VLC modulation), and environment (e.g., obstacles, ambient light), on the pairing success rate and pairing time.

For instance, our experiment shows the best focal length varies with the distance between the receiver and the lighting sources. However, in practice, the lighting sources may be placed at different distances, making it difficult to select

DOI 10.1109/SPW50608.2020.00034

an appropriate focal length value. Interestingly, we found that by setting the focal length to infinity, the resulting performance is near-optimal for all the tested distances. The reason for this is that although an infinite focal length blurs the captured images, these blurry images do not affect (and sometimes even help) the implemented decoding scheme. Moreover, our experiment also shows that the number of supported devices is limited by many factors, such as the scalability of the primary RF channel, the size of the camera view, and the protocol design.

This paper is organized as follows. Section II lists the unique properties of VLC and CamCom. Section III describes security protocols utilizing CamCom. Section IV presents the experiment results of our CamCom implementation. Section V discusses practical considerations we came across while implementing a CamCom-based security application. Section VI shows related work of VLC. Section VII concludes this work and briefly discusses future directions.

II. Background and Security Assumptions

A. Background: VLC and CamCom

Using visible lights, VLC channels are characterized by the following properties:

(P1) **Directional propagation**: Both RF and visible light are electromagnetic waves. While RF channels with centimeter-level wavelengths exhibit significant diffraction, nanometer-level visible light is highly directional, therefore requiring the receiver be present in line of sight.

(P2) **Human-visibility**: By definition, the VLC carrier is detectable by humans.

(P3) **Spatial confinement**: As the spectrum of visible light is different from the RF counterpart, VLC is immune to RF interference. In addition, visible light can be blocked by solid objects.

Camera communications (CamCom) is a subcategory of VLC, where the receiver is a commodity image sensor, such as a smartphone camera. CamCom additionally retains the following properties:

(P4) **Parallel channels:** A useful property of CamCom is the ability to receive from multiple light sources simultaneously, enabling communication with multiple devices at the same time.

(P5) **Precise positioning:** The receiver can determine its relative position to the transmitter with millimeter precision [11]. The receiver can also determine the relative location of the transmitter using computer vision techniques.

B. Security of VLC and CamCom

Given these five properties (P1–P5), VLC and CamCom are able to enhance physical-layer security in several aspects:

(S1) **Physical authenticity**: Combining P1 and P2, VLC demonstrates strong authenticity, in which data transferring is resistant to physical-layer modification and impersonation. Attacks attempting to tamper with the transmission have to be within line of sight, and thus can be detected by human observers watching the VLC carrier and surroundings. This makes VLC highly suitable as an out-of-band (OOB) channel for secure pairing.

(S2) **Ease of isolation**: Because visible light cannot penetrate obstacles (P3), an adversary behind a wall will be unable to eavesdrop on the VLC transmission. That is, secret exchanges via VLC can be easily isolated from potential eavesdroppers, whereas secret exchanges over RF channels cannot be confined without a Faraday cage. However, an adversary within line of sight can still observe the lighting patterns.

(S3) **Resilience to Jamming**: P3 ensures that VLC channels are free from RF interference. In addition, P4 ensures that many VLC channels can co-exist in close proximity without interfering with each other. Thus, data transmitted via visible lights is more robust to jamming than via RF.

(S4) **Visually verifiable association**: P5 strengthens physical authenticity (S1) by allowing a user to visually associate transmitted information to the transmitting device. While S1 binds the transmitted data to the physical presence of the transmitter, S5 binds the transmitted data to the precise physical location of the transmitter. This is particularly useful for preventing impersonation by nearby devices.

C. Insecurity of RF Channels

When designing a security protocol using VLC, VLC is often employed to establish an auxiliary channel in addition to the primary RF channel. Hence, it is important to mention typical assumptions about the RF adversary.

On RF channels, the adversary typically follows the Dolev-Yao Model [2], in which the attacker may eavesdrop, intercept and synthesize messages at their will. With complete control over the RF channel, the attacker may actively jam the channel to block messages, or hijack a session.

III. Security Protocols using CamCom

In this section, we describe example applications that utilize the unique advantages of CamCom (i.e., P4, P5, and S4) for better security. Due to the page limit, we elaborate on the first application (A1), which will be evaluated in subsequent sections.

(A1) Secure Multi-Device Pairing

CamCom can be used to simplify the protocol design of secure pairing, based on the security properties S1 and S4. Specifically, with CamCom, we can extend the pairing from

between one server and one client to between one server and multiple clients, without increasing the pairing time. Pairing with multiple devices at once is supported as long as the devices can visually fit inside the camera view. In addition, a camera on the receiver side and lighting sources (e.g., LEDs) on the transmitter side are required.

Algorithm 1 describes a typical VLC-based secure pairing protocol based on the Diffie-Hellman (DH) key exchange [5].

To support pairing with multiple devices, the server runs the protocol with each individual device, but the multi-device pairing is only successful if the server completes pairing with every device.

Algorithm 1 DH key exchange over VLC and RF channels

Symbols: Server A, Client device (VLC transmitter) B
A : generates DH key pair $sk_A = a, pk_A = g^a$
B : generates DH key pair $sk_B = b, pk_B = g^b$
$B \xrightarrow{RF} A$: pk_B
$B \xrightarrow{VLC} A$: hash(pk_B)
A : verifies pk_B with hash(pk_B)
$A \xrightarrow{RF} B$: pk_A
A : calculates shared secret $s_A = {pk_B}^{sk_A} = g^{ab}$
B : calculates shared secret $s_B = {pk_A}^{sk_B} = g^{ab}$
$A \xrightarrow{RF} B$: sends $E(s_A, N_A)$, using s_A as the key
$B \xrightarrow{RF} A$: returns $E(s_B, N_A, N_B)$, using s_B as the key
$A \xrightarrow{RF} B$: sends $E(s_A, N_B)$, using s_A as the key
if nonce correct **then**
  $A \xrightarrow{RF} B$: sends pairing success signal
  $A \xleftrightarrow{RF} B$: start encrypted communication using s_A and s_B
else
  A : abort
  B : not receiving success signal within timeout, abort
end if

Secure pairing often involves two channels: one primary channel that is high-capacity but insecure (e.g., RF), and the other out-of-band channel that is low-capacity but secure (e.g., VLC). The out-of-band channel delivers a little amount of information that can be used to validate the authenticity of the data chunk sent over the primary channel.

As described in Algorithm 1, when the protocol starts, the server begins listening and the client begins broadcasting its DH public key through RF and its hashed public key through VLC. In our implementation, we take the last 14 bytes of the SHA1 hash to balance security and speed.

Upon receiving the public key, the server verifies it with the hashed public key. Once verified, the server broadcasts its DH public key. Then both the server and client calculate a shared key for authentication. The server and client then perform nonce-based authentication to ensure a common secret is computed. The pairing succeeds if both the server and client can correctly retrieve their respective nonces.

Using this protocol, the attacker cannot easily tamper with pk_B since it is verified with the hashed public key sent over the VLC channel. It is possible that an attacker broadcasts their own public key $pk_{A'}$ to the client. Then the attacker may observe pk_B and construct a shared secret with the client. However, this results in pairing failure from the server's point of view, since ${pk_B}^{sk_A} = s_A \neq s_B = {pk_{A'}}^{sk_B}$. The server and the client then abort the connection.

(A2) Location-bound Authentication

Once all devices have been securely bootstrapped (e.g., by A1), each shares a secret key with the server. However, a user still cannot intuitively verify the claimed identity of a physically encountered device (e.g., a drone), as all devices may look alike. Moreover, a malicious device can replay a legitimate device's identity proof, whether it is sent over a RF or visible light channel.

To address this, we can leverage CamCom's unique property (S4) to generate a location-and-time-bound proof to authenticate a physical entity, which is particularly useful for tracking and managing the location of moving devices. For example, a device can generate an identity proof by computing a Message Authentication Code (MAC) over its current location and time using its secret key shared with the user. After receiving the proof from VLC, the user can visually verify whether the proof was indeed transmitted from the claimed location (based on S4), and whether the proof was sent recently.

Note that using a regular camera, the server can check the client's position in a 2D plane displayed on the screen, and thus can use the relative position on the screen to verify the client's claimed position. With an advanced camera supporting depth sensing, the server can determine the client's position in a 3D space for precise validation.

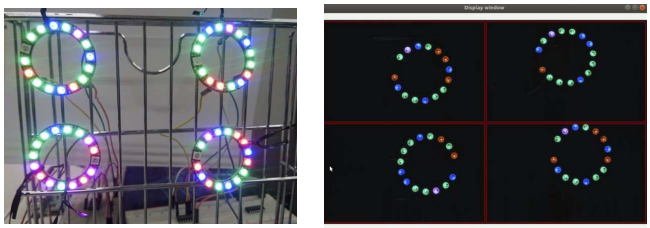
Since CamCom requires a camera as the receiver, one may consider a pure computer vision (CV) approach to passively track and identify devices (i.e., the devices will not actively emit signals). However, it is difficult to accurately track identical-looking devices with pure CV approaches.

IV. Experiment Results

This section reports the implementation details and experiments of A1. We take *pairing time* and *pairing success rate* as our evaluation metrics. The pairing time is the time (in seconds) that a pairing takes; the pairing success rate is the percentage of successful trials among all trials. A successful pairing indicates that the server completes a pairing with *every client device* within a timeout of 150 seconds.

A. CamCom Implementation

We implemented a color-based CamCom scheme designed to improve throughput by taking advantage of an array of LEDs and their colors. On the client side (the transmitting device), each client has a LED ring of 16 bulbs for transmitting messages. We select four colors (red, green, blue and magenta) that are maximally distant on the spectrum to



(a) LED rings with 16 bulbs

(b) Image captured by camera

Figure 1: The LED rings

increase the probability of accurate decoding. As each bulb on the ring illuminates one of the four colors at a time, each bulb can represent two bits of data and the ring in total can represent 32 bits at a time. Since the LED bulbs form a ring without apparent orientation, we purposely disable one of the 16 bulbs, so as to easily reconstruct the sequence of the bulbs. In our implementation, we set one bulb to be non-illuminated for the camera to locate the ring, one as the flag field two as the packet sequence number, and the rest carrying 24-bit information.

On the server side, after the server receives an image of LED rings via the camera, the server decomposes the image into pixels. We use a computer vision library to detect the range of each bulb on the ring and select a pixel within the range of the bulb to determine its color. By composing the bits decoded from the colors and orders, we can recover the message sent from the client side.

Our implementation of Algorithm 1 uses SHA1 for hash and AES256 for encryption.

B. Experiment Settings

The server side consists of a camera (GS3-u3-23S6C), a laptop (DELL-XPS-13-9343) and a Raspberry Pi 3. The camera is the receiver of the LED light. The laptop runs a picture decoder and a cipher calculator. The Raspberry Pi 3 serves as the Bluetooth (for RF) communication device. We prepare four client devices, each with a Raspberry Pi 3 and a LED ring connected by a CD74HCT125E chip. Both the Bluetooth communication and cipher calculation are completed on the Raspberry Pi 3.

The parameters that we can control are the distance between the LED rings and camera, the aperture, the focal length, and the number of client devices.

C. Results

To minimize the impact of ambient light during the experiment, we set the camera's aperture to its minimum setting. Each box or dot in the figures represents the result of 100 trials.

1) Impact of the RF Channel (Bluetooth): Since the performance of A1 relies on both the VLC and RF channels, we first investigate the difference between using CamCom with a real (self-implemented) Bluetooth channel and with

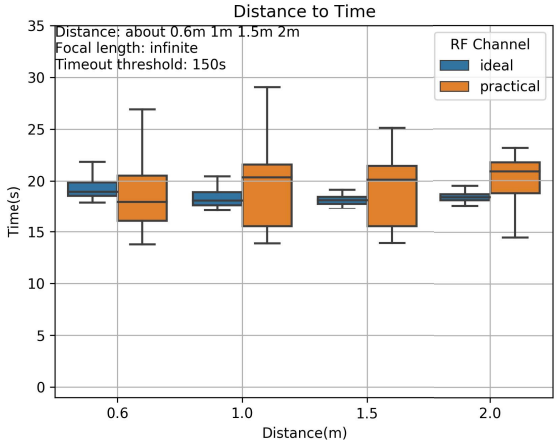


Figure 2: Pairing time using practical and ideal RF at different distances

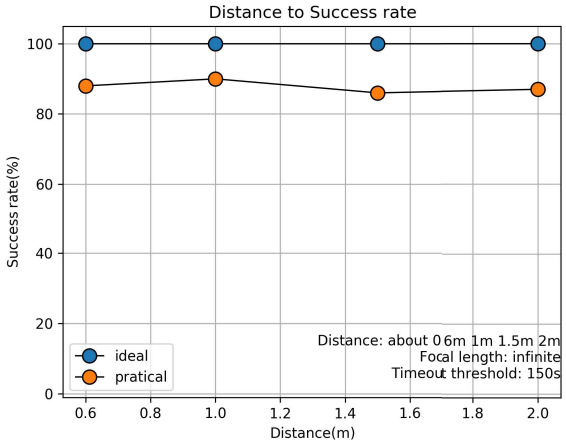


Figure 3: Pairing success rate using practical and ideal RF at different distances

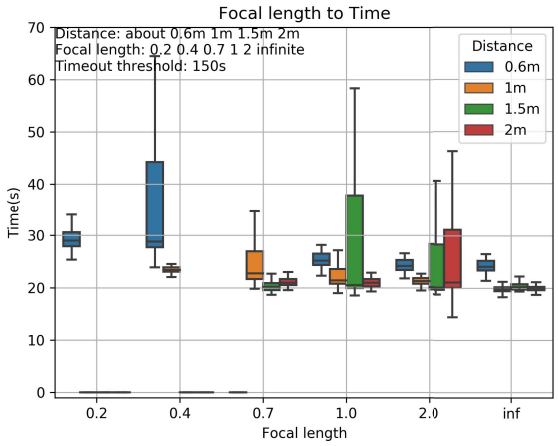


Figure 4: Four VLC devices' pairing time at different distances under different focal lengths

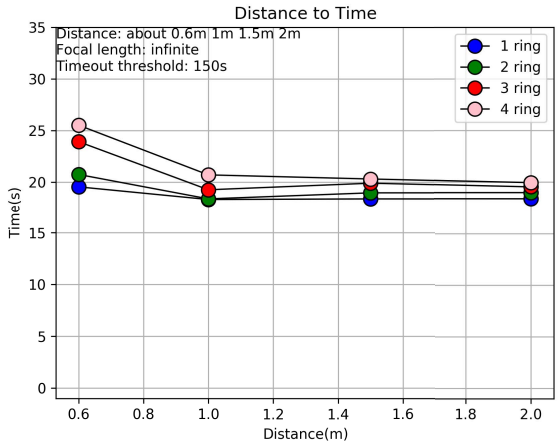


Figure 5: VLC devices' pairing time at different distances

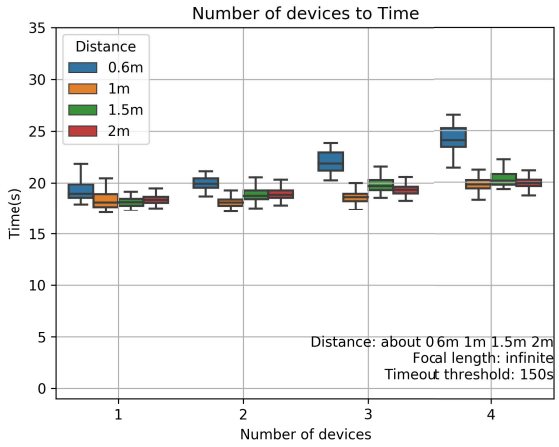


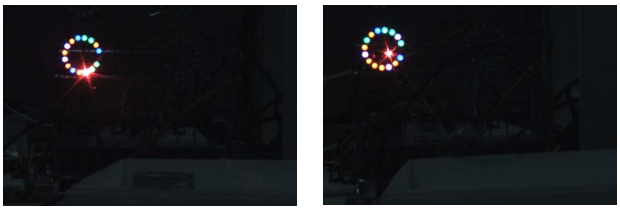
Figure 6: Pairing time at different distances from one to four VLC devices

an ideal one (which takes zero time to transmit). Figure 2 shows that when using a real Bluetooth channel, the pairing times have a high variance, possibly due to interference on the same frequency band and the non-optimized Bluetooth implementation. Using an ideal Bluetooth channel also leads to a higher pairing success rate than using a real one, as shown in Figure 3. In other words, though CamCom could scale to multiple devices, the RF channel may quickly become a bottleneck.

To investigate the impact of other factors, we consider the ideal Bluetooth channel for the rest of the experiments.

2) Impact of the Focal Length: Figure 4 shows the pairing times of four VLC devices at different distances under six focal lengths. Each focal length corresponds to four data boxes. A zero-value box means that the camera is unable to decode the LED rings' pictures under that circumstance.

We can see that the best focal length (leading to the lowest pairing time) varies with the distance between the receiver



(a) Emit laser at LED ring **(b)** Emit laser behind LED ring

Figure 7: Emitting laser

and the device. Although we could use an auto focus mode, this becomes complicated in practice, as the devices may be at different distances. Surprisingly, we found that the pairing time is shortest with an infinite focal length. Further investigation suggests that although an infinite focal length blurs the captured images, the blur increases the area of the colored pixels, which in turn helps the decoding process of our color-based CamCom scheme.

Hence, the focal length is set to infinity for the rest of the experiments.

3) Distance: Figure 5 shows the average pairing time (of 100 trials) at different distances for one to four LED rings, revealing a similar trend for different numbers of rings. Particularly, as the devices move away from the server, the average pairing times for the different numbers of rings are equally low. One reason could be that, as shown in Figure 4, an infinite focal length causes a higher pairing time at distance 0.6m than at further locations. Within our test range, the impact of distance is not as critical as we imagined.

Another observation about distance can be seen in Figure 3, which shows the pairing success rates of a single LED ring device using both real and ideal Bluetooth channels. With a real Bluetooth channel, the success rates is around 90% and does not significantly change as the distance increases, whereas the success rate remains 100% when considering an ideal Bluetooth channel regardless of distance. This also confirms the low impact of distance within our test range.

4) Number of Devices: Figure 6 shows that the pairing time increases with the number of client devices. This is reasonable, since as the number of LED rings increases, the receiver needs to decode additional images and thus slows down the decoding process.

5) External Light Source: To experimentally simulate what an adversary might do, we use a laser pen to emit red light as our external light source, and observe how it could affect the pairing process.

First, as shown in Figure 7b, we target the laser light at an LED bulb on the ring to override the color of the bulb, which effectively alters the message sent by the client, consequently corrupting the decoded message on the server side. This can be an effective denial-of-service attack if the

laser light goes undetected.

Second, as shown in Figure 7, we emit the laser from behind the ring to simulate a strong ambient light. This time, the decoded message is intact. Due to how our decoding algorithm works, the server could still differentiate whether the external light is on the ring or not.

An interesting future direction to explore is to evaluate whether an adversary can control the decoded message using such an ambient-light attack.

V. Practical Considerations

This section discusses CamCom's practical considerations, classified into three categories: hardware, software and configuration, and environment. We emphasize factors that affect the unique properties of CamCom (P4, P5, S4). When applicable, we describe how each factor affects the performance of our secure multi-device pairing protocol (A1).

Table I summarizes these factors and which part of the CamCom protocol (i.e., encoding, decoding, and communication) they affect.

A. Hardware

Some limitations are imposed by the transmitter/receiver hardware, such as the camera's frame rate and the maximum number of devices that can be connected via Bluetooth. Limitations in this category can only be overcome by replacing the hardware.

1) Frame per second (fps): Since CamCom uses a camera as the receiver, the channel throughput is bound by the camera's frame rate (how many images the camera can capture per second). If the transmitter (LEDs) changes color or blinks faster than the camera's fps, the receiver will be unable to decode the messages.

2) RF hardware: When a CamCom-based protocol leverages a RF channel for high-throughput transmission, its performance will be limited by the stability and scalability of the RF channel. For instance, Bluetooth is a common choice for implementing the primary RF channel. However, a Bluetooth master can connect to seven Bluetooth peripherals at most. Hence, even if CamCom can establish visible light channels with multiple devices simultaneously, when incorporating Bluetooth, the maximum number of devices is bound by the hard limit imposed by Bluetooth's protocol. Moreover, the range of the Bluetooth signal is limited. Class 1 Bluetooth devices can only transceive data within about 100 meters.

B. Software and Configuration

Factors in this category may be changed by updating the software code or configuration.

1) Aperture: The camera's aperture controls the amount of light that enters the lens. If the captured image by the camera is exposed to a high amount of ambient light, the receiver may fail to decode it. This is why we set the aperture to its minimum setting. It is worth investigating how to enhance resilience against strong lighting sources, e.g., sunlight and laser.

2) VLC modulation: Depending on modulation techniques and receiver types, the throughput of VLC channels can range from 10-13 bytes per second at the range of a few meters [6] to full Ethernet speed (10 Mbit/s) [1]. In fact, low channel throughput is a major downside of CamCom when compared to using a photodiode as the receiver.

3) Number of devices per camera view: In our VLC setting, each camera has a fixed dimension of view for capturing images. Thus, if we increase the number of devices that can fit into the camera's view without adjusting the distance, the space between the devices must be decreased. If the lighting sources of the devices are too close to each other, the receiver may have trouble decoding individual transmitted messages due to scattered light interference.

In other words, while CamCom is considered interference-free (P3) and thus suitable for communicating with multiple devices in parallel (P4), these parallel visible light channels may interfere with each other due to the way CamCom's decoding algorithm works.

C. RF selection

As mentioned above, the performance of a CamCom-based protocol will be limited by the RF channel, if exists. Besides using Bluetooth, one can also consider other RF channels such as WiFi, 4G, and ZigBee for different data rates, ranges, and power consumptions.

1) Cryptographic algorithms: Compared to photodiode-based VLC, CamCom has relatively low throughput (e.g., 10 bytes per second). Hence, as previously mentioned, CamCom is often reserved for sending critical or sensitive data, such as cryptographic tokens, that require a higher level of protection. Even so, sending a full 256-bit SHA256 hash would take more than 3 seconds, which may significantly increase latency, particularly for real-time protocols.

2) Frequency of message transmission: While frame-per-second (fps) defines the speed of data reception, the frequency of transmission focuses on the speed of data transmission on the transmitter side. Since usually a light bulb can blink or change colors faster than the fps rate, we need to programmably control the rate of the lighting change, such that the receiver can reliably capture all encoded data.

D. Environment

Factors in this category are subject to the surrounding environment and may be unable to control by users.

1) Obstacles: Since visible light travels in straight lines, any obstacles between the transmitter and the receiver would interfere with the data transmission.

Table I: The three categories of practical considerations and the aspects affected by them.

	Hardware	**Software & Configuration**	**Environment**
Transmitter (encoding)		VLC modulation Message transmission frequency	
Receiver (decoding)	Frame per second	Aperture Number of devices per camera view	Distance Ambient light
Communication	RF hardware	RF selection Cryptographic algorithms	Obstacle

2) Distance: Because data is extracted from the captured image in CamCom, we suspected that as the distance between the transmitter and receiver gets longer, the contribution of the lighting source becomes lower and harder to decode. However, our experiments disproved this: the change in distance is not significant within the range of 1 to 2 meters. One reason for this could be that using an infinite focal length blurs the image and increases the area of the colored pixels, making up for the distance.

3) Ambient light: Ambient light, such as the sun or unintended light sources in the same indoor environment, is one of the main challenges to visible light communication. Since the ambient light source is usually weaker than the transmitter light source (e.g., LEDs), in CamCom, we could limit the interference of ambient light by controlling the aperture.

VI. Related Work

Arfaoui *et al*. [3] present an overview of physical-layer security and visible light communication (VLC). Using a unified mathematical model, they consider factors such as the characteristics of the VLC channel, input distribution, the architectures of the transmitter/receiver, etc. They also discuss the secrecy level that can be achieved by different types of VLC systems (SISO, MISO, MIMO, and hybrid) under various assumptions. Blinowski [4] proposes a method to quantify the security risks of in-door VLC. A risk value can be calculated for jamming, snooping and data modification based on factors such as data transmission range and power. Rohner *et al*. [9] summarize the challenges and opportunities of VLC security. Compared to these surveys on VLC security, our work focuses on CamCom, a special type of VLC, and investigates its security properties and practical considerations through experiments.

Mostafa and Lampe [8], [7] propose physical-layer techniques to enhance the secrecy rates of VLC in the presence of eavesdroppers. An interesting future direction is to explore physical-layer techniques that can improve other security properties of VLC and CamCom (as presented in Section II-B).

Saxena and Ekberg [10] propose secure pairing protocols that require only a unidirectional OOB channel. Chen et al. [5] suggest a secure key exchange scheme between multiple devices using a trusted server, which distributes public keys via Wi-Fi and transmits data commitments over VLC. Our work discusses the practical considerations of CamCom-enabled security protocols and uses secure multi-device pairing as an example.

VII. Conclusion

Camera Communication (CamCom) presents new opportunities for security applications. CamCom supports precise locationing and the establishment of multiple visible light channels simultaneously; it also strengthens physical-layer authenticity by allowing visually verifiable association, binding transmitted data to the precise physical location of the transmitter. Due to these properties, we demonstrate that CamCom enables location-bound authentication and simplifies multi-device secure pairing protocols. However, when implementing such security applications with CamCom, we observe certain fundamental limitations and tradeoffs that could potentially degrade expected performance and/or security.

For future work, we plan to study CamCom under adversarial settings in which an adversary attacks CamCom directly on the physical layer, and explore CamCom schemes that are robust against such attacks. One physical-layer attack is using laser light to deliberately manipulate the message. As presented in our laser-attack experiment, while the attacker can successfully interfere with CamCom decoding, the attack is obvious to human observers checking the ring or the screen. Can the attacker stealthily alter the decoded message by changing the radius and power of the laser light?

In conclusion, we hope our results can shed light on the realistic behaviors of CamCom, helping researchers to make realistic assumptions when adopting it for security applications.

Acknowledgments

This research was supported in part by the Ministry of Science and Technology of Taiwan (MOST 108-2633-E-002-001, 109-2636-E-002-021), National Taiwan University (NTU-108L104039), Intel Corporation, and Delta Electronics. We gratefully thank Matthias Schunter and Hsin-Mu Tsai for their constructive comments. We also thank Wei-Chen Yeh for contributing to the VLC design and implementation.

References

[1] "RONJA," http://ronja.twibright.com/.

[2] R. M. Amadio and W. Charatonik, "On name generation and set-based analysis in the dolev-yao model," in *International Conference on Concurrency Theory (CONCUR)*, 2002.

[3] M. A. Arfaoui, M. D. Soltani, I. Tavakkolnia, A. Ghrayeb, C. Assi, M. Safari, and H. Haas, "Physical layer security for visible light communication systems: A survey," 2019.

[4] G. Blinowski, "Security issues in visible light communication systems," *IFAC-PapersOnLine*, vol. 28, no. 4, pp. 234–239.

[5] Y.-S. Chen, C.-Y. Lin, H.-C. Hsiao, Y.-H. Lin, and H.-M. Tsai, "Poster: VLC-based Authenticated Key Exchange," in *IEEE Symposium on Security and Privacy, 2016*.

[6] H.-Y. Lee, H.-M. Lin, Y.-L. Wei, H.-I. Wu, H.-M. Tsai, and K. C.-J. Lin, "Rollinglight: Enabling line-of-sight light-to-camera communications," in *ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2015.

[7] A. Mostafa and L. Lampe, "Physical-layer security for indoor visible light communications," in *IEEE International Conference on Communications (ICC)*, 2014.

[8] ——, "Enhancing the security of vlc links: Physical-layer approaches," in *IEEE Summer Topicals Meeting Series (SUM)*, 2015.

[9] C. Rohner, S. Raza, D. Puccinelli, and T. Voigt, "Security in visible light communication: Novel challenges and opportunities," *Sensors and Transducers*, vol. 192, pp. 9–15, 09 2015.

[10] N. Saxena, J.-E. Ekberg, K. Kostiainen, and N. Asokan, "Secure device pairing based on a visual channel: Design and usability study," *IEEE Transactions on Information Forensics and Security*, vol. 6, pp. 28 – 38, 04 2011.

[11] Y.-L. Wei, C.-J. Huang, H.-M. Tsai, and K. C.-J. Lin, "Celli: Indoor positioning using polarized sweeping light beams," in *ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2017, pp. 136–147.