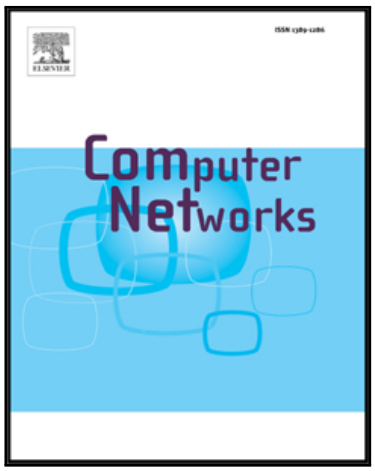

Cyber-Physical Systems Information Gathering: A Smart Home Case Study

Quang Do , Ben Martini , Kim-Kwang Raymond Choo

PII: S1389-1286(18)30144-0
DOI: 10.1016/j.comnet.2018.03.024
Reference: COMPNW 6451

To appear in: *Computer Networks*

Received date: 2 May 2017
Revised date: 15 March 2018
Accepted date: 22 March 2018

Please cite this article as: Quang Do , Ben Martini , Kim-Kwang Raymond Choo , Cyber-Physical Systems Information Gathering: A Smart Home Case Study, *Computer Networks* (2018), doi: 10.1016/j.comnet.2018.03.024

Highlights

- Cyber-Physical Systems Information Gathering
- Smart home forensics
- Internet of Things (IoT) forensics
- Forensically sound adversary model

Title page

Title: Cyber-Physical Systems Information Gathering: A Smart Home Case Study

Authors:

Quang Do
School of Information Technology and Mathematical Sciences, University of South Australia, Adelaide, SA 5095, Australia
quang.do@mymail.unisa.edu.au

Ben Martini
School of Information Technology and Mathematical Sciences, University of South Australia, Adelaide, SA 5095, Australia
ben.martini@unisa.edu.au

Kim-Kwang Raymond Choo [Corresponding author]
Department of Information Systems and Cyber Security and Department of Electrical and Computer Engineering, The University of Texas at San Antonio, San Antonio, TX 78249, USA
School of Information Technology and Mathematical Sciences, University of South Australia, Adelaide, SA 5095, Australia
raymond.choo@fulbrightmail.org

Cyber-Physical Systems Information Gathering: A Smart Home Case Study

Quang Do [a], Ben Martini [a] and Kim-Kwang Raymond Choo [b,a]

[a] School of Information Technology and Mathematical Sciences, University of South Australia, Adelaide, SA 5095, Australia

[b] Department of Information Systems and Cyber Security and Department of Electrical and Computer Engineering, The University of Texas at San Antonio, San Antonio, TX 78249, USA

quang.do@mymail.unisa.edu.au; ben.martini@unisa.edu.au; raymond.choo@fulbrightmail.org

Abstract

With the growth in the use of Cyber-Physical Systems, such as Internet of Things (IoT) devices, there is a corresponding increase in the potential attack footprint of personal and corporate users. In this paper, we explore the potential for exploiting information retrieved from two IoT devices which, seemingly, are unlikely to store substantial amounts of data. We specifically focus on prominent smart home devices for the purpose of obtaining compromising information. We undertake a collection and analysis process, constrained by the limitations placed upon three types of adversaries, namely: forensic passive, forensic active and real-time active. The former two adversaries aim to comply with the requirements of forensic soundness, whereas the real-time active adversary does not have these constraints and therefore more closely models a malicious real-world attacker. The findings show that a variety of device data is available to even the passive adversary, and this data can be used to determine the actions and/or presence of an individual at a given time based on their interactions with the IoT device. These interactions can be both user initiated (e.g. powering on or off a switch or light) and device initiated (e.g. background polling).

Keywords

Cyber-physical system forensics; Smart home forensics; Internet of Battlefield Things forensics; Internet of Things forensics; Forensic adversary

1 Introduction

Cyber-physical systems play an increasingly important role in society. These systems generally consist of a number of Internet of Things (IoT) devices or "things" (e.g. sensors, smartphones, etc.), which cooperate in such a way that they are able to achieve a common objective [1]. Cyber-physical systems (and IoT devices, also known as smart devices) are now used in power management, healthcare, supply management, public safety, smart homes and more. One of the most promising current cyber-physical systems applications is the smart home. For example, smart devices such as temperature sensors and smart lights can be coordinated in order to improve quality of life and even reduce power consumption.

The application of digital forensics in the context of smart home devices is an exceptionally new field of study. The motivating factors for this research are the rapidly rising popularity of these smart devices and the fact that these devices have already been used in illegal attacks with great impact [2-4]. In fact, one of the largest ever distributed denial-of-service attacks was performed with the help of smart devices [2].

Oriwoh et al. [5] noted that current cyber-physical systems and IoT research focuses on the potential benefits that IoT can provide, including the possible future applications of cyber-physical systems and IoT, and that there is a distinct lack of methodologies for digital forensic responses. Smart cities, an emerging cyber-physical systems and IoT infrastructure wherein a city's infrastructure consists of a number of highly integrated smart devices [6], are an area that would benefit greatly from specialized forensic methodologies. Bajramovic et al. [7] highlighted the importance of digital forensics knowledge and forensic readiness in smart buildings and smart cities in order to prepare for future security incidents.

In addition to the use of cyber-physical systems and IoT devices as part of cyber security attacks, evidence obtained from such systems and devices will likely see growing use in traditional criminal cases. For example, a recent high profile murder case heard in Australian courts presented evidence relating to when a mobile device was connected to a charger within a home as part of the trial (see *R v Baden-Clay* [8]). The evidence allowed prosecutors to demonstrate inconsistencies in the timeline of events as given in testimony by the defendant, specifically the charging events extracted from the device provided circumstantial evidence, which indicated that the defendant may have been awake contrary to their testimony.

In another case involving IoT devices (see *Commonwealth v. Risley* [9]), an individual's Fitbit (a smart fitness band) was able to contradict their sexual assault allegations, which were originally presented to court. Specifically, the Fitbit's activity logs were able to show that the individual was in fact awake and also mobile, as the device contained an accelerometer with continuous logging, when they had originally claimed that they were asleep. The court ruled that the individual had fabricated the incident based on this finding.

Smart home devices have the potential to be of even more importance in criminal investigations. For example, a number of recent smart home devices come with motion detectors (e.g. Belkin Switch + Motion) or microphones (e.g. Amazon Echo and Google Home), which may more conclusively prove the location of a suspect or even some of their actions. In fact, in the currently ongoing Arkansas murder case of *State v James Andrew Bates* [10], investigators have requested recordings from an Amazon Echo device which was present at the premises of the murder. However, Amazon did not provide these recordings. This highlights the importance of investigators being able to obtain their own evidential data without deferring to an external, possibly uncooperative, party. In addition to an Amazon Echo, other IoT devices including Belkin WeMo devices and a Nest Thermostat were found at the premises. The investigators also located two smartphones and deemed one of these smartphones infeasible to examine due to the use of hardware encryption. A smart meter on the premises also showed that 140 gallons of water had been used between 1:00 am and 3:00 am on the day of the incident.

These examples have demonstrated the importance of secondary sources of evidence, particularly IoT devices, in a criminal investigation and the subsequent legal proceedings.

In this paper, we present an adversary model constructed for use in IoT security and forensics. We construct two forensic adversaries (a passive adversary and an active adversary) and a single active malicious adversary from this adversary model. Using these three adversaries, we explore two smart home device case studies, namely: a smart light and a smart switch. We then determine the types of data

that could be obtained and the actions that might be performed by an increasingly more powerful adversary.

These types of consumer IoT devices are likely to be some of the most prevalent home automation devices currently deployed. The smart light is a LIFX Original 1000 and the smart switch is a Belkin WeMo switch. As is common with many smart home devices, these devices were initially paired using a smartphone to connect to the respective device's ad-hoc network and subsequently providing the smart device with the local (Wi-Fi) network's credentials (see "Initial Connection" in Figure 1). These smart devices are discussed in further detail below.

The following section examines related work in the fields of IoT security and IoT forensics. In Section 2, we present background information with regards to the smart home devices examined and a formal definition for forensic soundness, an important concept used throughout the remainder of this paper. Section 3 presents the adversary model and the three constructions. The smart home devices are examined as case studies in Sections 4 and 5. Lastly, Section 6 concludes this paper.

2 Related Work

In the following sections, we discuss research performed in both fields of IoT security and IoT forensics. IoT security refers to methods which researchers have proposed that may either violate the protections of an IoT device or improve (i.e. harden) the device's security. As opposed to IoT security, IoT forensics seeks methods to obtain evidential data from these devices. Furthermore, the processes are (generally) required to be forensically sound (see [11] and Section 2.3) to ensure that the evidence obtained is not considered tainted in a court of law (i.e. legally admissible).

2.1 IoT Security

A number of researchers have presented security analyses of specific smart home devices. Ho et al. [12] examined the security of five popular smart locks and developed two types of attacks with every device being susceptible to at the very least, one type of attack. One of these attacks made use of the fact that several of the tested locks did not have direct Internet connections. These locks would innately trust and utilize the Internet connection of the paired mobile device. This attack took advantage of the lock's trust model in order to allow an attacker with revoked access to avoid both the revocation check and logging of access on the server. The second attack relied on the proximity-based automatic unlocking capability available to several locks to allow an attacker to enter a home if the owner's mobile device was within a certain distance of the smart lock.

Ronen and Shamir [13] demonstrated a number of attacks on smart lights. They considered a covert channel comprised of a number of smart lights that were connected via an internal network. These smart lights would transmit data by changing the intensity such that the difference was imperceptible to the human eye but perceivable by a light sensor. The light sensor would use the changes in intensity in order to exfiltrate data from the network. The authors note that they were able to accurately transmit data over a distance of more than 100 meters.

In 2013, Crowley et al. [14] examined eight different home automation devices and noted that the majority of these devices were secure to Internet-based attacks. They found, instead, that manufacturers did not seem to consider local network attacks in the design of smart home devices. Furthermore, the authors note that due to these devices having direct control over the physical world, this lack of security could be detrimental to users or businesses. For example, a nearby attacker could potentially disable smart cameras and smart alarms of a business or home. Later, in 2014, Costin et al. [15] performed a large-scale security analysis of the firmware of embedded devices. As part of their study, they examined the firmware

of a number of home automation devices and found that one of these devices employed a backdoor which could be triggered by receiving a certain string.

Another 2014 study was that of Notra et al. [16] who examined the security of three smart home devices, namely: the Nest smoke alarm, the Philips Hue smart light and the Belkin WeMo motion switch. The authors found an attack for the Philips Hue smart light that allowed an attacker to take control of the smart hub which controlled the smart light. They also found that the Belkin WeMo motion switch did not utilize encryption in both the local network and communications with the Belkin servers. In a similar study, Sivaraman et al. [17] introduced a malicious app to the Apple App Store with an aim of detecting and compromising the user's local smart home devices. They demonstrated this attack on a number of smart home devices, namely: a smart camera and a Belkin WeMo switch. They were able to maliciously configure the devices such that the attacker could thereafter remotely control these devices via the Internet. Nobakht et al. [18] proposed the use of an intrusion detection system that could detect and mitigate similar types of attacks. They tested their intrusion detection system, known as IoT-IDM, with a Philips Hue smart light, and were successful in detecting unauthorized access to the device.

A number of security analyses on smart home devices was also performed by Dhanjani [19]. Dhanjani found a number of security vulnerabilities in smart lights, smart door locks, smart televisions and even smart baby monitors.

2.2 IoT Forensics

With regards to specific forensic analyses of smart home devices, there have been a number of studies. Sutherland et al. [20] investigated the future forensic requirements for the growing capabilities of smart televisions. The authors proposed that smart TVs can contain a wealth of digital evidence (e.g. viewing behaviors, browsing history and network information) which could be a valuable resource in an investigation. Boztas et al. [21] proposed a digital forensics investigation procedure for use on smart TVs. The authors analyzed a Samsung smart TV and categorized the potential artifacts into several categories, including: network information, app activity, web browsing history, media files and cloud artifacts.

As opposed to the relatively few security and forensics IoT device analyses, there have been a greater number of generalized IoT digital forensics methodologies proposed in the literature. One of the earliest IoT forensics processes was conceptualized by Oriwoh and Sant [22] in 2013. The researchers considered a smart home scenario in their Forensics Edge Management System (FEMS). FEMS allows for forensic investigations to be performed on smart home IoT networks and also provides a number of security features. Perumal et al. [23] developed a model for use by forensic practitioners in smart home investigations. In 2015, Zawoad and Hasan [24] proposed what they termed the "first working definition of IoT forensics" and presented a model for use in forensics investigations involving IoT infrastructures. They noted a number of challenges unique to IoT forensics, with the main problem being the lack of a standardized protocol for IoT. This means that many IoT devices use proprietary protocols, thereby reducing the applicability and efficacy of common digital forensics tools. Similarly, Kebande and Ray [25] suggested a framework for digital forensics investigations involving IoT technologies. The authors compared their framework with that of Oriwoh and Sant [22], Perumal et al. [23] and Zawoad and Hasan [24]. Kebande and Ray [25] concluded that their framework was stronger with regards to proactive forensics (e.g. identification of evidence and scenarios).

Although a number of works have examined the value of IoT devices in the information gathering process, there is little work focused on the contexts of these attacks or investigations. For example, researchers have previously demonstrated the insecurities present in IoT devices but have not considered the real-world impact such devices may have, such as applications in legal proceedings and espionage.

We seek to rectify this discrepancy in the literature by examining the potential information that can be obtained from smart home devices from the perspectives of both forensic and malicious attackers (see Section 3). We highlight the importance of classifying the information that can be obtained by adversaries of varying abilities and, as a result, provide sources of information leakage present on smart home devices.

2.3 Forensic Soundness

Forensic soundness is an important concept that we refer to throughout the remainder of this paper. In our previous research [11], we formally defined this concept based on the two seminal works examining this notion (see [26, 27]). Forensic soundness refers to the fact that evidence collection processes must follow a certain standard if they are to be considered admissible in a court of law. A particular process is considered to be forensically sound if it meets the following four criteria:

1. **Meaning** – evidence collected must retain the original meaning and interpretation but preservation without change may be impractical and in fact hinder a forensic investigation.
2. **Errors** – it is a requirement that errors be identified if they exist and sufficient justification should be provided to ensure that the validity of the evidence is not affected. A common method of error-checking is cryptographic hashes of obtained files of interest.
3. **Transparency and Trustworthiness** – These criteria highlight a requirement that the processes used should have independent verification through, for example, an external examination of the forensic procedures by a court of law.
4. **Experience** – the individuals undertaking the forensic investigation should have sufficient experience or knowledge such that their findings can be trusted.

We also previously defined adversary capabilities as having one of two increasingly weaker definitions of forensic soundness: a **strict** level of forensic soundness and a **standard** level of forensic soundness [11]. Strict forensic soundness refers to techniques which result in no changes being made to data stored by target evidential devices, whereas standard level forensic soundness permits changes where the exact extent of the change is known and documented by the forensic practitioner and the change does not violate any of the above four criteria.

3 Proposed Adversary Model

We now propose an adversary model for use in this research based on the query-based adversary model of Bellare and Rogaway [28, 29], and Pointcheval [30]. In our previous research [31], we demonstrated that adversary models were a valuable resource in representing an attacker and their potential actions even without the context of a cryptographic protocol.

The adversary model capabilities, derived from our previous research [31], are as follows:

1. **`Listen`** (*Channel or Target*): Allows an adversary to (passively) monitor a communications channel or a target device.
2. **`Transmit`** (*Message*, *Target*): Allows an adversary to transmit a message to a target destination. The adversary must utilize another capability in order to obtain any replies (e.g. `Listen`).
3. **`Modify`** (*Message*, *Target*): Allows an adversary to modify a particular (unencrypted) message (e.g. a file, configuration, etc.) on a device. This capability may be triggered as a result of another adversarial capability (e.g. `Transmit`).
4. **`Intercept`** (Target): Allows the adversary to obtain all messages targeted at target device (e.g. via the use of ARP spoofing). The adversary can then choose to forward, drop or modify such

messages. This adversary capability can be considered a combination of the `Listen` (e.g. listening to redirected messages), `Transmit` (transmission of poisoned ARP packets and forwarding of messages) and `Modify` (e.g. modifying ARP tables on devices) capabilities. An important point to note is that this capability is not considered applicable in a forensics context.

5. **`Corrupt`** (*Target*): Allows the adversary to obtain control of a target (e.g. a device). Adversaries that are able to employ this capability are considered some of the most powerful attackers.

As the **`Listen`** capability typically does not make any changes to an evidential source, it is considered a forensically sound adversarial capability of the strict level. **`Transmit`** and **`Modify`**, are classified at the weaker "standard" level of forensic soundness as they have the potential to modify the evidence source.

The **`Intercept`** capability is usually used as part of a real-time attack and is not likely to be used as part of a post-event digital forensic investigation. The final capability, **`Corrupt`**, has been included for the sake of completeness and for use in future research where, for example, the firmware of the device has been modified (e.g. where state actors could install backdoors as part of the supply chain [32]). One example of an extremely powerful active adversary that utilizes the **`Corrupt`** capability is an adversary with possession of the manufacturer's cryptographic signing key. This adversary would be able to send arbitrary data via over-the-air firmware updates to the relevant devices. As a result, the adversary would be able to, theoretically, add backdoors and "corrupt" these devices. This powerful but infeasible attacker is not considered in our research.

Based on the proposed adversary model, we construct three specific adversaries, namely: a weaker passive adversary constrained by the strict principals of forensic soundness, and two stronger adversaries that are able to utilize a number of active capabilities. The first active adversary is constrained by the standard level of forensic soundness (see Section 2.3) whilst the latter is not. These adversaries are summarized in Table 1 and described in detail further below.

	Forensic Passive Adversary (see Section 3.1)	**Forensic Active Adversary (see Section 3.2.1)**	**Real-time Active Adversary (see Section 3.2.2)**
Adversary Capabilities			
• `Listen`	•	•	•
• `Transmit`		•	•
• `Modify`		•	•
• `Intercept`			•
Adversary Environment	Post-event	Post-event	Real-time
Forensic Soundness Level	Strict	Standard	None

Table 1: A summary of the three adversaries considered in this paper.

3.1 Forensic Passive Adversary

This construction of the adversary model represents a forensic practitioner seeking to uphold the highest level of forensic soundness in their investigation. The aim of this model is to preserve the integrity of the evidential data obtained such that the findings are suitable for presentation in a court of law. There are many similarities with this model and the passive (i.e. Eve) adversary model used in computer security research with the main difference being the inclusion of a forensic soundness restriction. The sole capability of this adversary is **`Listen`**. Due to the strict restriction of forensic soundness on this

adversary, the **`Listen`** capability is the only applicable adversarial power as it typically does not modify any potential evidential sources.

3.2 Active Adversaries

The active adversary constructions, on the other hand, are able to transmit messages in the network and receive replies. We consider two types of active adversaries in this research, namely: an active forensic adversary and a real-time active adversary.

3.2.1 Forensic Active Adversary

The active forensic adversary represents a forensic practitioner who has undertaken forensic examinations at the strict level without obtaining a sufficient amount of digital evidence. Even if the practitioner may have adequate evidential data, they are likely to seek more context and details relating to the evidence, which would require a more in-depth and hands-on approach. For example, the forensic practitioner could have the detail that a particular device was enabled or switched on during a certain timeframe but may wish to obtain the specific times. This information may be present on the device but require the practitioner to perform activities on the evidential sources directly (e.g. the device may be encrypted or the data may be contained within volatile memory and therefore, may not be obtainable from an image of the device). Such a situation requires the use of a less strict forensically sound adversary model.

In our research, this adversary has access to the following adversarial capabilities: **`Listen`**, **`Transmit`** and **`Modify`**.

3.2.2 Real-time Active Adversary

The real-time active adversary is the most powerful adversary considered in this research. This adversary emulates an attacker performing an active attack on a smart home network with the intent of obtaining compromising information or causing harm to the victims. This compromising information could, for example, be the times that a user is in their premises. Such information could be collected by planting a device in the premises and exfiltrating data from the smart devices on the network to a remote location. As opposed to the forensic adversaries considered, the real-time adversary is not only interested in information but may also wish to cause harm to the user.

In addition to the adversarial capabilities that the active forensic adversary (i.e. the standard-level forensic adversary) has access to, the real-time adversary can **`Intercept`** messages in the network and choose whether to forward, drop or modify these messages. Although this adversary shares the majority of its capabilities with the active forensic adversary, this particular adversary performs attacks in real-time. For example, the adversary may be compromising the system when a user is present or not a suspect of an ongoing investigation. As a result, the adversary is able to obtain further information and have a level of influence on the target users at a degree much greater than in forensic examinations.

In addition to the standard malicious (e.g. Byzantine) adversary, this adversary could also be used to represent an intelligence-gathering entity (i.e. espionage). As the entity is, typically, not interested in conforming to the strict principles of forensic soundness, they can obtain further information by querying or intercepting messages from devices on the network. Information obtained by this adversary could be considered “tainted” in a court of law due to the potential modifications made to evidential sources [33], and are often used for intelligence rather than evidential purposes.

We now perform two smart home case studies from the perspective of the three previously constructed adversaries.

4 Case Study: LIFX Original 1000

In this section, we consider adversaries that are present on the network after a device has been initialized (i.e. the "Initial Connection" stage in Figure 1 has been completed by the user of the device). In the case of the LIFX Original 1000, this temporary unsecured infrastructure network is not available once initialization is complete. A summary of the findings is provided at the end of this section in Table 3.

4.1 Background: LIFX Original 1000

The LIFX Original 1000 is a smart light able to produce up to 16 million different colors, and capable of connecting to a local Wi-Fi network and receive commands from any compatible device. In the initial setup, the LIFX Original 1000 broadcasts an unsecured infrastructure network with which a smartphone running the LIFX app can connect. The smartphone app proceeds to request the password for the user's Wi-Fi network which it sends (together with the SSID) to the smart light. At this point, the smart light can be fully controlled via the smartphone app connected to the user's regular access point. This sequence is described in Figure 1.

The LIFX Original 1000 considered in this study was running firmware version 2.1 and communicating with a smartphone running the LIFX app version 3.4.4.

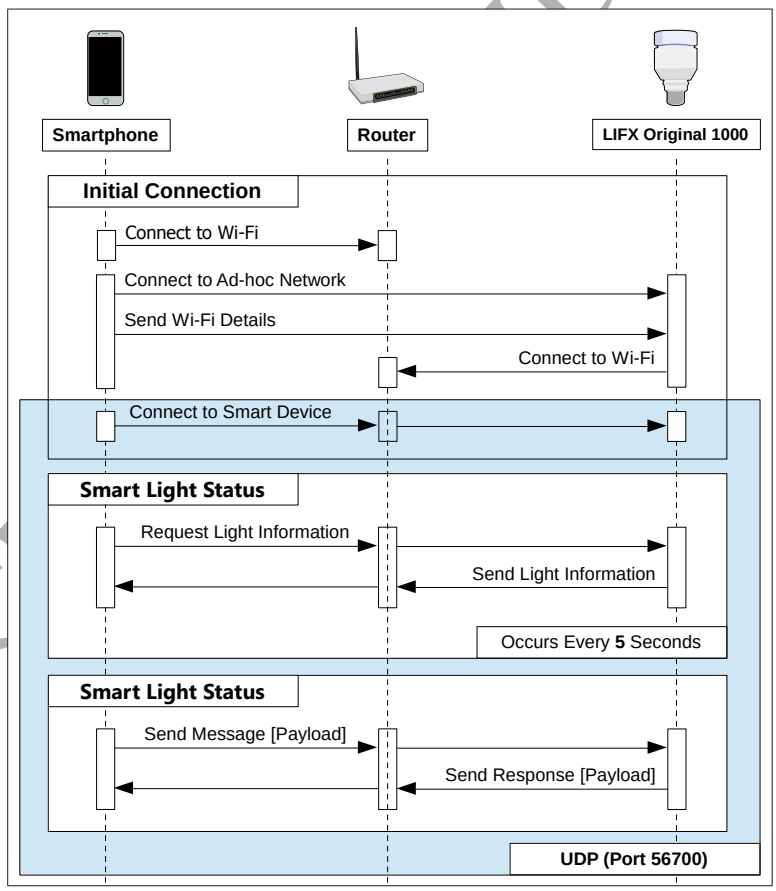

Figure 1: LIFX Original 1000 Communication Protocol.

4.2 Forensic Passive Adversary

A passive adversary connected to the network can eavesdrop (e.g. **`Listen`**) to some information of interest that is sent by the smart light. Under the default configuration, the smartphone app requests data from the LIFX smart light approximately every five seconds. Specifically, the app requests the device's Wi-Fi information (*GetWifiInfo*) and light state (*Get*) every five seconds (see Table 2 and "Smart Light Status" in Figure 1). The light replies to these requests with the respective response messages: "*StateWifiInfo*" and "*State*".

These messages are often broadcasted even when a device is in a standby state (e.g. the phone screen is off but the phone is still on standby). As a result, the adversary is able to obtain the approximate location of the smart light (based on Wi-Fi signal strength), the amount of data the smart light has received or transmitted through the network, the current state of the light (e.g. color and whether the light is on or off), and any custom identifiers the user has utilized for the smart light (e.g. the light may be labelled "Bedroom Light" or "Light #2"). These identifiers may aid an investigator in inferring the existence of other evidence sources or relevant information.

The LIFX app also allows the user to set a schedule with regards to when a device should turn on or off, or modify the light color. These schedules are stored in the cloud via the user's LIFX Cloud account and, as such, collection of this data is not considered in this paper. However, when the LIFX Original 1000 is configured with a cloud-based schedule, via this system, these commands can be collected by the adversary when they are triggered. As these commands may continue to run (e.g. continuous schedules such as the changing of a light's state every weekday) even when a user is not present or interacting with the device, such data could be collected by the passive adversary.

The commands are sent directly to the LIFX Original 1000 in a format that is vastly different to the local communications protocol used by the device. Messages are transmitted using TLS version 1.2, which would present a significant challenge for forensic practitioners wishing to examine these packets. Nonetheless, upon receiving a command from the cloud server, the LIFX Original 1000 sends a number of directed broadcasts (i.e. subnet-only broadcasts) which indicate to local devices that the light state has changed. If this broadcast is received by a local device running the LIFX app, the app then requests the smart light's current state information. The broadcast message itself does not contain specific information relating to the changes made to the smart light. Both the directed broadcasts and state requests are performed using UDP on port 56700, which is the primary local communications standard for LIFX smart devices, at the time of research.

4.3 Forensic Active Adversary

The first and most crucial step is to determine if any smart devices are connected and enabled on the network. If passively listening (i.e. the passive adversary) does not detect any devices, such as times when the smartphone app is not running either as a service or in the foreground, we require active methods in order to find these devices. The LIFX Original 1000 did not respond to ping requests during our testing, meaning basic tools which utilize ICMP-based host discovery will not detect these devices. Other, more in-depth, host discovery techniques such as ARP scanners will, however, discover the device.

We determined that the most reliable and swiftest method of determining the presence of LIFX smart lights was to make use of the adversary's **`Transmit`** capability and broadcast the "*GetService*" message through the network under the UDP protocol on the appropriate port (56700 in our research). Upon receiving this message, even without any authentication or prior messages, the LIFX smart light responds with a message containing the communication protocol (UDP) and port (56700). Upon confirming that

the device is present on the network, the adversary can now communicate with the device via device specific messages (see Table 2) as it does utilize any form of local authentication.

Device Request Message	Device Response Message
GetService	*StateService:* • Service: 1 (UDP) • Port: 56700 (typically)
GetHostInfo	*StateHostInfo:* • Signal strength • Bytes transmitted since power on • Bytes received since power on
GetHostFirmware	*StateHostFirmware:* • Build time (since epoch) • Version number
GetWifiInfo	*StateWifiInfo:* • Signal strength • Bytes transmitted since power on • Bytes received since power on
GetPower	*StatePower:* • Power draw level (either 0 or 65535)
SetPower: • Power draw level (either 0 or 65535)	No Response
GetVersion	*StateVersion:* • Vendor ID • Product ID • Hardware version number
GetInfo	*StateInfo:* • Current time (since epoch) • Device uptime • Time of last power down
Get	*State:* • Color • Power • Device label
SetColor: • Power level • Duration	No Response

Table 2: A number of LIFX Original 1000 messages and responses (see also the LIFX API [34]).

With regards to the potential data that can be obtained from the device, Table 2 lists the items of most interest to an adversary. An active adversary is able to obtain a wealth of information from these devices using the `Transmit` capability.

In addition to the data that a passive adversary listening to the smartphone app and smart light's communication can collect, the active forensic adversary is also able to obtain the specific device information (e.g. model number). In Section 4.2, we noted that upon receiving a cloud-based command, the LIFX Original 1000 would send a directed broadcast to the local network. As these specific commands were transmitted in an encrypted format, the adversary should instead request the light's state information (e.g. via the "*Get*" device request message) manually upon receiving this directed broadcast message. A small embedded device installed in the local network, such as a Raspberry Pi, could be used to perform this task.

4.4 Real-Time Active Adversary

In addition to the capabilities of the passive and active forensic adversaries, the real-time adversary is able to intercept and modify messages in order to obtain further information or for other purposes. In order to demonstrate this capability, we designed an application that would perform ARP spoofing targeting the smartphone (after detection of the smart light and smartphone via host discovery). As a result, the application was able to intercept the messages intended for the smart light and either modify, drop or forward these messages. This is indicative of the fact that the LIFX protocol does not inherently authenticate client messages. The adversary could then covertly extract the collected data or obtain it by sending it through the Internet. This information, such as times when the lights are switched off or dimmed, could allow an adversary to determine the times that an individual is present on the premises or asleep. The **`Intercept`** capability unique to this adversary is described in Figure 2.

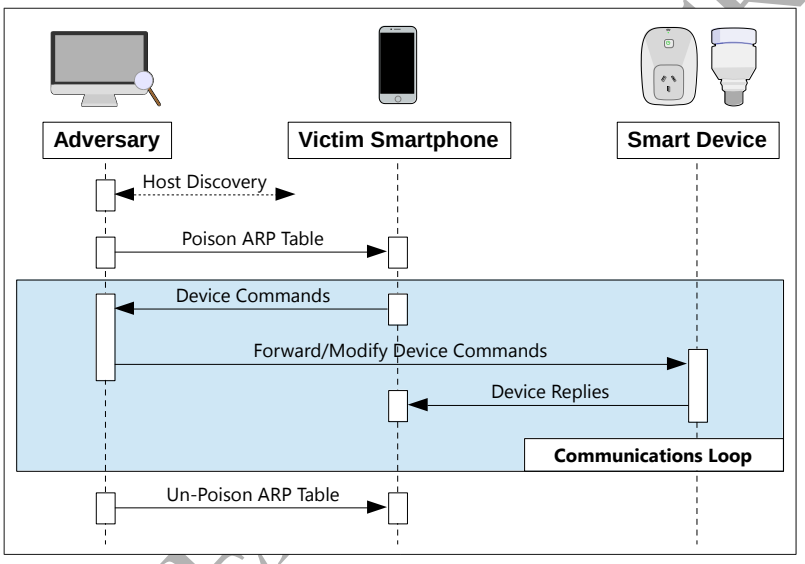

Figure 2: The real-time adversary's `Intercept` capability.

A nefarious adversary would be able to continuously change the brightness and color of the smart light, to potentially trigger epilepsy in photosensitive individuals, as seen in the work of Ronen and Shamir [13]. For example, in December 2016, an individual was accused of sending a tweet (a strobing GIF with the message: "You deserve a seizure.") to a journalist that triggered his epilepsy [35]. Ronen and Shamir [13] also demonstrated that compromised smart LEDs can be used to covertly exfiltrate data using changes in the light color which is unnoticeable to the human eye. Such attacks should be possible under the active real-time adversary proposed in our research. Although these attacks simply require the adversary to transmit and receive messages in the network (a capability that is shared with the active forensic adversary), the forensic adversary is a post-event adversary. As a result, these attacks are considered infeasible and counter-productive (i.e. these attacks may taint the evidential data) for the aforementioned forensic adversary.

As the active real-time adversary undertakes the attack when a user is present or in an active home, the amount of information the adversary can obtain by eavesdropping on the network communication changes as a direct result. An adversary can determine if a user is actively using the smartphone app to

change the smart light by checking for the "*SetColor*" messages and can determine when a user is opening the app by the broadcast of three specific messages in the following order: "*GetVersion*", "*GetHostFirmware*" and "*GetWifiFirmware*".

Adversary Capability	Potential Actions	Information Acquired	Forensic Passive Adversary	Forensic Active Adversary	Real-time Active Adversary
`Listen`	• Eavesdrop local communications between the smartphone (app) and the smart light.	• Wi-Fi signal strength and bytes received/transmitted. • User customizable label for the device. • The smart light state (i.e. on or off and color/brightness). • Commands received by the device from the cloud (e.g. scheduled or manual changes to the smart light).	•	•	•
`Transmit`	• Transmit host discovery messages in order to determine the presence of smart devices. • Request specific responses from the smart light by transmitting the appropriate commands.	• IP addresses and MAC addresses, of devices on the network (and related information). • Device location, firmware, uptime, downtime, state, etc.		•	•
	• Send spoofing packets (e.g. ARP poisoning).	N/A			•
`Modify`	• The actions performed by **`Transmit`** can cause a number of modifications to occur on the smart light. For example, the number of bytes transmitted or received would be modified.	N/A		•	•
	• Change the color, state and brightness of the light. • Exfiltrate data by causing minute changes in the smart light's colors [13].	N/A			•
`Intercept`	• A combination of the above capabilities allows the adversary to intercept and modify, drop or forward messages intended for a target.	• Specific messages sent by the user via the smartphone app to the smart light.			•

Table 3: A summary of the adversaries described in the LIFX Original 1000 case study.

5 Case Study: Belkin WeMo Switch

Similar to the above LIFX Original 1000 case study, we consider the scenario where a single Belkin WeMo switch has already been initialized by the user. The initial connection protocol is identical to that of the former device (see also "Initial Connection" in Figure 1).

5.1 Background: Belkin WeMo Switch

Belkin's WeMo smart switch allows a user to control the power state of a connected device. Akin to the LIFX Original 1000, initial setup requires the user to connect to the Belkin WeMo's temporary unsecured infrastructure network in order to provide credentials to the user's Wi-Fi network. After provisioning, the device can be controlled using a smartphone with the Belkin WeMo app.

The Belkin WeMo switch can be customized to a certain extent via the use of a rules system. The switch itself can be configured with instructions, such as when the connected device should be on or off based on the time of the day. For example, a bed lamp could be configured to automatically turn on at 10:00pm every day and turn off at 7:00am. Such configurations are constructed by the user through the use of the smartphone app.

The Belkin WeMo switch was running firmware version 2.00.10626 and communicated with a smartphone running the Belkin WeMo app version 1.17.1.

5.2 Forensic Passive Adversary

As with the LIFX Original 1000 case study, the first step is host discovery. Unlike the LIFX Original 1000, the Belkin WeMo utilizes Universal Plug and Play (UPnP). This means that it uses Simple Service Discovery Protocol (SSDP), a type of multicast advertisement and discovery protocol that is broadcasted with an IP address of 239.255.255.250 on port 1900. As a direct result, all devices on the network are able to regularly receive notification that a UPnP device is present [36]. Furthermore, this results in the host discovery process being feasible with a passive forensic adversary without requiring an investigator to examine all communications between all devices on the network.

Communications between the device and smartphone app utilize the Simple Object Access Protocol (SOAP), which is transported via HTTP. Due to the use of SOAP, the message body is stored and transmitted as human-readable XML data. As the smartphone app must be explicitly controlled by a user, the collection of communications between the smartphone and Belkin WeMo is considered infeasible for the forensic passive adversary. This is in contrast to the LIFX Original 1000, where the smartphone service would repeatedly communicate with the smart device, which resulted in more data (e.g. Wi-Fi and device information) being available to the adversary. In the case of the Belkin WeMo, the forensic passive adversary is only able to determine the existence of one or more devices based on the LAN-based information.

Akin to the LIFX Original 1000, the Belkin WeMo allows remote cloud-based access to the device and remote management of device rules and schedules. These communications between the cloud server and device could be captured by a forensic investigator in order to obtain further details. Such communications data could be automated on/off rules based on the time or day. The Belkin WeMo cloud service initially communicates with the device using the "Traversal Using Relays around NAT" (TURN) protocol, which is transported over TCP, and the cloud service itself is located on Amazon's servers and appears to be running TurnServer 0.4. The initial TURN messages sent and received by the Belkin WeMo do not appear to be encrypted but are unlikely to contain any data of significant interest.

After a direct connection between the device and server is established, the server transmits the command(s) previously set by the user. Similar to the LIFX Original 1000 cloud-based commands, these messages appear to be encrypted. Also resembling the LIFX Original 1000, the Belkin WeMo notifies local devices of a change in the state of the device. However, the Belkin WeMo directly communicates with local devices via the use of the HTTP "Notify" method, which includes information concerning the changes made to the device's state. As these messages are transmitted unencrypted, the passive adversary is able to directly read these packets and ascertain the state of the device.

5.3 Forensic Active Adversary

Due to the relative simplicity of the Belkin WeMo Switch (as it is simply a smart power switch), the amount of information an adversary is able to obtain from the device is reduced compared to the LIFX Original 1000 examined above. Nonetheless an active adversary with forensic capabilities is able to obtain a range of sensitive information (see also Table 4).

Perhaps the most important information a forensic adversary can obtain from a smart switch is the state of the device. Unlike the LIFX Original 1000, the Belkin WeMo does not provide the device's uptime, which may be of interest to a forensic practitioner (e.g. a user may have recently changed the state of the device). Of secondary importance is the rules system—users can set custom rules for each device in the household (see Figure 3 in Section 5.4). These rules include turning a device on or off: after a countdown (e.g. a ten minute countdown), for a period of time each day (e.g. between 10pm and 7am) or at certain times on certain days (e.g. between 10pm and 7am on weekdays). Such information can aid an investigator in determining an individual's daily pattern or lifestyle. A similar automated system was employed by the LIFX Original 1000, known as "schedules", which was instead cloud-based.

These rules are stored directly on the Belkin WeMo and can be accessed by an active forensic adversary by performing an HTTP GET on the "rules.db" file from the root of the Belkin WeMo. Our findings showed that the file obtained was in fact a compressed ".zip" containing a single file: the SQLite 3 format rules database. The compressed file was not password-protected.

The Belkin WeMo app allows a user to customize the photograph that is displayed when a device is detected by the app. This image, when configured by the user, is stored directly on the Belkin WeMo device. As such, an adversary (even without access to the smartphone or app) is able to access this information. These images are stored as unencrypted files with the ".jpg" extension directly on the Belkin WeMo. Our further examination determined that these files were, in fact, PNG files. As these images can be any photograph taken by the user, they may contain details of device's physical location or other photographic details. As the images are converted to a reduced resolution PNG when a user configures a new device image, any metadata (e.g. EXIF data) present in previous image is lost.

Device Method	Device Response
GetExtMetaInfo	*GetExtMetaInfo:*
GetInformation	*GetInformationResponse:* • Product name, MAC address, port number, current state (on or off), "friendly" name, current time on device, device firmware version, etc.
SetBinaryState: • Binary State: 1 or 0 (on or off)	*SetBinaryStateResponse:* • Binary State: 1 or 0 (on or off) • Current device time (milliseconds since epoch)
StoreRules: • Rules database in Base64 format.	*StoreRulesResponse*

GET /setup.xml (HTTP)	An XML file containing the types of services supported by the Belkin WeMo including: the device's friendly name (configurable by the user), device serial number, firmware version, MAC address and state (i.e. on or off).
GET /icon.jpg (HTTP)	Icon image (stored on the device).
GET /rules.db (HTTP)	A .zip file containing an SQLite 3 database of the device's rules.
POST /icon/jpg (HTTP)	HTTP response (e.g. 200 OK).

Table 4: Belkin WeMo device methods and responses (including data of interest).

5.4 Real-Time Active Adversary

The most powerful adversary in this study, the real-time active adversary, is able to obtain similar amounts of information to the active forensic adversary but has greater capabilities relating to information manipulation. For example, an active real-time adversary is able to influence the Belkin WeMo by changing or adding their own rules. As previously described, an adversary would be able to override or add rules of their own without requiring access to the user's smartphone by using the "GET /rules.db" and "*StoreRules*" device methods. Furthermore, no forms of authentication are required to read or modify these rules.

As a direct result of the comparatively fewer capabilities of the Belkin WeMo (compared to the LIFX Original 1000), the interception capabilities of the real-time active adversary are not entirely necessary for information gathering. Furthermore, the Belkin WeMo is not prone to multiple operations within a short timespan (e.g. compared to the changing light colors of the LIFX Original 1000, the Belkin WeMo can either accept rule changes or change its on/off state). As a result, the adversary can simply query the device directly for, what is most likely, up-to-date state or rule information instead of performing an interception attack.

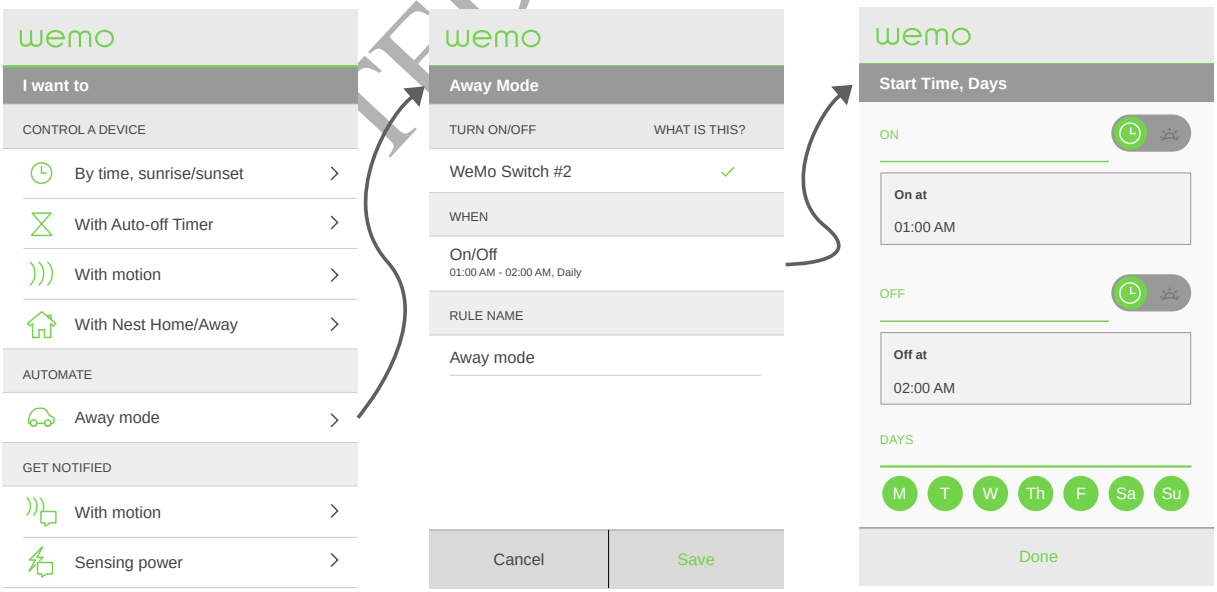

Figure 3: Belkin WeMo rules setup screen (left), "Away Mode" setup screen (center) and "Away Mode" time settings screen (right).

The "Away Mode" feature, available in the Belkin WeMo smartphone app (see Figure 3), is designed to simulate an occupied household by changing the state of devices at random times during the day. An adversary would be able to determine that this particular setting is active by querying the device and thus conclude that the user may not present for an extended period of time. The "Away Mode" is a part of the Belkin WeMo rule system and, as such, this configuration is stored within the rules database on the device.

Compared to the LIFX Original 1000, the real-time adversary in this case study is not able to obtain significantly more information when contrasted to the forensic active adversary (see Table 3 and Table 5, respectively).

Adversary Capability	Potential Actions	Information Acquired	Forensic Passive Adversary	Forensic Active Adversary	Real-time Active Adversary
`Listen`	• Eavesdrop communications on the network between devices (e.g. the smart switch and router) and between the smartphone and device (e.g. the Belkin WeMo app).	• Determination of presence of smart devices and their model numbers. • Cloud-based commands: e.g. scheduled or manual state changes.	•	•	•
`Transmit`	• Transmit device methods.	• Device meta-info: e.g. device state, custom labels, custom rules and icon. • Deduce a user's lifestyle or daily pattern based on the rules set.		•	•
	• Transmit spoofing packets.	N/A			•
`Modify`	• Actions performed by `Transmit` may cause a number of modifications to occur on the smart switch.	N/A		•	•
	• Replace, modify or add device rules. • Activate or deactivate "away" mode.	N/A			•
`Intercept`	• A combination of the above capabilities allows the adversary to intercept and modify, drop or forward messages intended for a target.	• Specific messages sent by the user via the smartphone app to the smart switch (e.g. rules or state changes).			•

Table 5: A summary of the adversaries described in the Belkin WeMo case study.

6 Conclusion

With an increasing number of cyber-physical systems being used in evidence as part of legal investigations, the literature in this area is currently still lacking. We have found that one of the most

essential pieces of evidence obtainable from cyber-physical systems, and in the context of this paper smart devices (particularly smart home devices), is temporal information which may corroborate with other important findings in these legal investigations.

In 2013, Crowley et al. [14] found that the majority of smart home devices did not adequately protect against LAN-based attacks, and based on the findings of our case studies in this paper, we believe that this still remains the case in 2017. For example, the lack of client authentication (in the case of both devices—e.g. the ability to change client messages in transit for the LIFX device and the ability to change the ruleset for the Belkin WeMo) for LAN-based communications provides adversaries with a significant attack vector.

In this paper, we defined and demonstrated the capabilities of three different types of adversaries using two smart home case studies. The case studies involved a smart light and a smart switch, two seemingly innocuous IoT devices.

One interesting finding is that passive adversaries (i.e. eavesdroppers) are able to obtain a significant amount of evidential data. Examples include: determination of devices present on the network and smart device status messages which may contain information pertaining to the locations of these devices. This result implies that an adversary should always take on the role of a passive adversary in the first instance in order to minimize modifications to potential evidence sources and to reduce the potential for detection.

We also found that there exists a substantial amount of information that may be of forensic interest stored on smart home devices. For example, the Belkin WeMo's rules feature could allow an investigator to determine the schedule of an individual or household. The fact that such a seemingly innocent feature could impact the results of a forensic investigation suggests that such features may exist in other (currently under-examined) smart devices. Similarly, we found that the LIFX Original 1000 stored the amount of time that the device had been powered on—this information could provide investigators with a specific time-frame in which an individual may have been present or active.

The third and most powerful adversary is able to actively obtain and modify messages sent by the user (and device) in real-time in order to obtain even more information or perform a malicious task. As noted in the paper, this adversary is the sole non-forensic adversary considered in this paper (i.e. the adversary resembles a typical malicious adversary). As a direct result, this adversary's activities included malicious activities outside of direct information gathering. For example, in the LIFX Original 1000 case study, the adversary could potentially subtly change the color of the smart light in order to exfiltrate data to an external location (e.g. see [13]).

These three adversaries have served to highlight the utility of constructing specific adversaries from an adversary model. Furthermore, these adversaries can be catered to fit an attacker's profile and this has been demonstrated in a meaningful manner in this paper. Such a method of initially designating one or more potential attackers and then proceeding to determine the potential impact that these adversaries may have on a system appears to be under-utilized in the literature.

6.1 Recommendations

As a direct result of the two case studies, we located a number of vulnerabilities present on both the LIFX Original 1000 and the Belkin WeMo. Both of these devices do not utilize any form of security for local communications. Consequently, these packets are unsecured and decipherable by any device on the network.

The Belkin WeMo makes use of the SOAP specification that is delivered using HTTP. Therefore, a change to HTTPS would dramatically improve the security of these communications. It is not uncommon for smart devices to secure local network communications. For example, MakerBot's 5th Generation 3D printers utilize HTTPS for part of their local communications [31].

In a different vein, the LIFX Original 1000 uses a custom protocol that communicates over the stateless UDP specification. This means that in order to improve the security of this unique protocol, a more in-depth approach is required (e.g. DTLS [37]).

6.2 Future Work

Future work includes the examination of other smart devices including IoT devices outside of those used in smart homes and those used in adversarial settings such as battlefields (e.g. Internet of Battlefield Things and Internet of Military Things). Extensions to the adversary model and adversary constructions could also be considered in these future studies.

Such extensions could include adversary capabilities specifically designed to cater for IoT, g. Internet of Battlefield Things and Internet of Military Things-based adversaries or adversaries that utilize the `Corrupt` capability proposed as part of the adversary model in Section 3.

Another potential extension is to integrate the forensic adversary model in a proof of concept that can be used to automate the identification and exploitation of vulnerabilities in a broader range of smart devices.

Acknowledgements

We thank the editor-in-chief, guest editors and anonymous reviewers for their constructive feedback. The last author is supported by the Cloud Technology Endowed Professorship.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787-2805, 2010.

[2] Z. Whittaker. "After massive cyberattack, shoddy smart device security comes back to haunt," Accessed 9th November 2016; http://www.zdnet.com/article/blame-the-internet-of-things-for-causing-massive-web-outage/.

[3] I. Zeifman, D. Bekerman, and B. Herzberg. "Breaking Down Mirai: An IoT DDoS Botnet Analysis," Accessed 9th November 2016; https://www.incapsula.com/blog/malware-analysis-mirai-ddos-botnet.html.

[4] Level 3 Threat Research Labs. "Attack of Things!," Accessed 9th November 2016; http://blog.level3.com/security/attack-of-things/.

[5] E. Oriwoh, D. Jazani, G. Epiphaniou, and P. Sant, "Internet of Things Forensics: Challenges and Approaches," in Proceedings of the 9th International Conference Conference on Collaborative Computing: Networking, Applications and Worksharing, 2013, pp. 608-615.

[6] T. Nam, and T. A. Pardo, "Conceptualizing Smart City with Dimensions of Technology, People, and Institutions," in Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times, College Park, Maryland, USA, 2011, pp. 282-291.

[7] E. Bajramovic, K. Waedt, A. Ciriello, and D. Gupta, "Forensic Readiness of Smart Buildings: Preconditions for Subsequent Cybersecurity Tests," in Proceedings of the 2016 IEEE International Smart Cities Conference, 2016, pp. 1-6.

[8] R v Baden-Clay, QSC 156, 2014 p. 1 (Supreme Court of Queensland).

[9] Commonwealth v. Risley, Criminal Docket: CP-36-CR-0002937-2015, 2016.

[10] State v James Andrew Bates, Case ID: 04CR-16-370.

[11] Q. Do, B. Martini, and K.-K. R. Choo, "A Forensically Sound Adversary Model for Mobile Devices," PLoS ONE, vol. 10, no. 9, pp. 1-15, 2015.

[12] G. Ho, D. Leung, P. Mishra, A. Hosseini, D. Song, and D. Wagner, "Smart Locks: Lessons for Securing Commodity Internet of Things Devices," in Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, Xi'an, China, 2016, pp. 461-472.

[13] E. Ronen, and A. Shamir, "Extended Functionality Attacks on IoT Devices: The Case of Smart Lights," in Proceedings of the 2016 IEEE European Symposium on Security and Privacy, 2016, pp. 3-12.

[14] D. Crowley, D. Bryan, and J. Savage, "Home Invasion v2.0 - Attacking Network-Controlled Hardware," in Black Hat USA 2013, 2013, pp. 1-15.

[15] A. Costin, J. Zaddach, A. Francillon, and D. Balzarotti, "A Large-Scale Analysis of the Security of Embedded Firmwares," in Proceedings of the 23rd USENIX Security Symposium, 2014, pp. 95-110.

[16] S. Notra, M. Siddiqi, H. H. Gharakheili, V. Sivaraman, and R. Boreli, "An Experimental Study of Security and Privacy Risks with Emerging Household Appliances," in Proceedings of the 2014 IEEE Conference on Communications and Network Security, 2014, pp. 79-84.

[17] V. Sivaraman, D. Chan, D. Earl, and R. Boreli, “Smart-Phones Attacking Smart-Homes,” in Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks, Darmstadt, Germany, 2016, pp. 195-200.

[18] M. Nobakht, V. Sivaraman, and R. Boreli, “A Host-Based Intrusion Detection and Mitigation Framework for Smart Home IoT Using OpenFlow,” in Proceedings of the 11th International Conference on Availability, Reliability and Security, 2016, pp. 147-156.

[19] N. Dhanjani, Abusing the Internet of Things Blackouts, Freakouts, and Stakeouts, Sebastopol: O'Reilly Media, 2015.

[20] I. Sutherland, H. Read, and K. Xynos, “Forensic analysis of smart TV: A current issue and call to arms,” Digital Investigation, vol. 11, no. 3, pp. 175-178, 2014.

[21] A. Boztas, A. R. J. Riethoven, and M. Roeloffs, “Smart TV forensics: Digital traces on televisions,” Digital Investigation, vol. 12, Supplement 1, pp. S72-S80, 2015.

[22] E. Oriwoh, and P. Sant, “The Forensics Edge Management System: A Concept and Design,” in Proceedings of the 10th International Conference on Ubiquitous Intelligence and Computing and 10th International Conference on Autonomic and Trusted Computing, 2013, pp. 544-550.

[23] S. Perumal, N. M. Norwawi, and V. Raman, “Internet of Things(IoT) digital forensic investigation model: Top-down forensic approach methodology,” in Proceedings of the 5th International Conference on Digital Information Processing and Communications, 2015, pp. 19-23.

[24] S. Zawoad, and R. Hasan, “FAIoT: Towards Building a Forensics Aware Eco System for the Internet of Things,” in Proceedings of the 2015 IEEE International Conference on Services Computing, 2015, pp. 279-284.

[25] V. R. Kebande, and I. Ray, “A Generic Digital Forensic Investigation Framework for Internet of Things (IoT),” in Proceedings of the 4th International Conference on Future Internet of Things and Cloud, 2016, pp. 356-362.

[26] E. Casey, “What does “forensically sound” really mean?,” Digital Investigation, vol. 4, no. 2, pp. 49-50, 2007.

[27] R. McKemmish, "When is Digital Evidence Forensically Sound?," Advances in Digital Forensics IV, IFIP — The International Federation for Information Processing I. Ray and S. Shenoi, eds., pp. 3-15: Springer US, 2008.

[28] M. Bellare, and P. Rogaway, "Entity Authentication and Key Distribution," Advances in Cryptology — CRYPTO’ 93, Lecture Notes in Computer Science D. R. Stinson, ed., pp. 232-249: Springer Berlin Heidelberg, 1993.

[29] M. Bellare, and P. Rogaway, “Provably Secure Session Key Distribution: The Three Party Case,” in Proceedings of the 27th Annual ACM Symposium on Theory of Computing, Las Vegas, Nevada, USA, 1995, pp. 57-66.

[30] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated Key Exchange Secure against Dictionary Attacks," Advances in Cryptology — EUROCRYPT 2000, Lecture Notes in Computer Science B. Preneel, ed., pp. 139-155: Springer Berlin Heidelberg, 2000.

[31] Q. Do, B. Martini, and K. K. R. Choo, “A Data Exfiltration and Remote Exploitation Attack on Consumer 3D Printers,” IEEE Transactions on Information Forensics and Security, vol. 11, no. 10, pp. 2174-2186 2016.

[32] C. Macdonald. "WikiLeaks 'Dark Matter' dump claims CIA has targeted Apple devices for at least a decade with software used to bug 'factory fresh iPhones'," Accessed 29th March 2017; http://www.dailymail.co.uk/sciencetech/article-4343102/WikiLeaks-claims-CIA-bug-factory-fresh-iPhones.html.

[33] R. McKemmish, "What is Forensic Computing?," Trends & Issues in Crime and Criminal Justice, vol. 118, pp. 1-6, 1999.

[34] LIFX. "LIFX LAN Protocol," Accessed 10th February 2017; https://lan.developer.lifx.com/docs.

[35] T. Collins. "Reporter goes after Twitter user for allegedly causing seizure," Accessed 24th April 2017; https://www.cnet.com/au/news/kurt-eichenwald-reporter-twitter-epilepsy-seizure/.

[36] A. Donoho, B. Roe, M. Bodlaender, J. Gildred, A. Messer, Y. Kim, B. Fairman, and J. Tourzan. "UPnP Device Architecture 2.0," Accessed 29th March 2017; http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v2.0.pdf.

[37] E. Rescorla, and N. Modadugu. "[RFC 6347] Datagram Transport Layer Security Version 1.2," Accessed 3rd April 2017; https://tools.ietf.org/html/rfc6347.

Quang Do received the Bachelor's (Hons.) degree in computer science from the University of South Australia. He is the recipient of the prestigious University of South Australia Vice Chancellor and President's Scholarship, and is an active Android security researcher. His research interests include Android user privacy, forensics, and security.

Ben Martini is a Lecturer in Cybersecurity at the University of South Australia. His research focus is digital forensics and information security, particularly relating to the cloud and mobile nexus. He has published a book entitled "Cloud Storage Forensics", and a number of refereed conference and journal articles. Ben has worked actively in the South Australian IT industry sectors including government departments, education and electronics across various organizations and continues to deliver occasional invited presentations to industry organizations in his area of expertise.

Kim-Kwang Raymond Choo holds the Cloud Technology Endowed Professorship at the University of Texas at San Antonio, and has a courtesy appointment at the University of South Australia. He is the recipient of ESORICS 2015 Best Paper Award, Winning Team of the Germany's University of Erlangen-Nuremberg Digital Forensics Research Challenge 2015, 2014 Highly Commended Award by the Australia New Zealand Policing Advisory Agency, Fulbright Scholarship in 2009, 2008 Australia Day Achievement Medallion, British Computer Society's Wilkes Award in 2008, etc. He is also a Fellow of the Australian Computer Society, an IEEE Senior Member, and the Honorary Commander of 502nd Air Base Wing, Joint Base San Antonio-Fort Sam Houston.