Feasibility of Societal Model for Securing Internet of Things

Hiroshi Tsunoda[1], Rodrigo Roman[2], Javier Lopez[2], and Glenn Mansfield Keeni[3]
[1] Faculty of Engineering, Tohoku Institute of Technology
Sendai, Miyagi, 982-8577 - JAPAN
[e-mail: tsuno@m.ieice.org]
[2] Network, Information and Computer Security (NICS) Lab,
University of Malaga, 29701, Malaga - SPAIN
[e-mail: {roman,jlm}@lcc.uma.es]
[3] Cyber Solutions Inc. Sendai, Miyagi, 989-3204 - JAPAN
[e-mail: glenn@cysols.com]
*Corresponding author: Hiroshi Tsunoda

Received October 15, 2017; revised December 7, 2017; revised January 22, 2018; accepted February 20, 2018; published August 31, 2018

Abstract

In the Internet of Things (IoT) concept, devices communicate autonomously with applications in the Internet. A significant aspect of IoT that makes it stand apart from present-day networked devices and applications is a) the very large number of devices, produced by diverse makers and used by an even more diverse group of users; b) the applications residing and functioning in what were very private sanctums of life e.g. the car, home, and the people themselves. Since these diverse devices require high-level security, an operational model for an IoT system is required, which has built-in security. We have proposed the societal model as a simple operational model. The basic concept of the model is borrowed from human society – there will be infants, the weak and the handicapped who need to be protected by guardians. This natural security mechanism works very well for IoT networks which seem to have inherently weak security mechanisms. In this paper, we discuss the requirements of the societal model and examine its feasibility by doing a proof-of-concept implementation.

Keywords: Internet of Things, security, operational model, network architecture, SNMP

A preliminary version of this paper [1] appeared in the 13th International Wireless Communications and Mobile Computing Conference IEEE IWCMC 2017), June 26-30, Valencia, Spain. This version includes a comprehensive survey of the related works, thorough descriptions of the proposed model, details of the proof-of-concept implementation, and expanded considerations.

http://doi.org/10.3837/tiis.2018.08.003 **ISSN : 1976-7277**

1. Introduction

Internet of Things (IoT) has penetrated almost every sphere of society. In the IoT concept, various devices such as sensors and actuators possess computing capability and network connectivity. As a result, these devices are accessible for monitoring, control and information collection, via the literally ubiquitous Internet.

The IoT concept is bringing in an entirely new gamut of services and applications. At the consumer end, driver-less cars with automatic control and braking mechanisms are emerging, and smart homes with automatically controlled electrical appliances are maturing. In the industry, automated systems to monitor and control factory and plant processes are developing rapidly.

While the IoT paradigm will bring various attractive services and economic impact, security and privacy issues have been a major focus area for IoT [2], [3]. One of the reasons is that IoT devices will potentially be used in very private sanctums of life, e.g. in the car, inside the home and maybe even inside the human body. In addition, various critical infrastructures such as smart grid and energy plants are extensively deploying IoT devices for wide area monitoring and control. Consequently, if IoT systems are compromised, there is a serious risk that human life will be at risk and life-line services will be disrupted and social order will be breached.

In fact, there have been already a myriad of IoT related security incidents in the consumer area. For specialized IoT devices and Internet-connected systems, such as healthcare devices and connected vehicles, various problems have been reported [4]-[6]. Moreover, the attacks against consumer-level IoT products such as TVs, cameras, and even intelligent wearables and smart toys, are steadily increasing [7]. As for the industrial area, there are various attacks that can be performed against the Industrial Internet of Things (IIoT) [8]. Given the alarming amount of attacks that have targeted important infrastructures [9], it is expected that, unless properly protected, the IIoT will also be the target of such attacks.

The impact of such IoT related vulnerabilities is not limited to the exposed devices or their environment: IoT objects might become attackers themselves, targeting anyone, anywhere, anytime [10]. A clear example of this situation is the advent of IoT botnets, where the communication and processing resources of a huge number of vulnerable IoT devices with Internet access are used to mount massive Distributed Denial-of-Service (DDoS) attacks on targets [11], [12]. One of the largest DDoS attacks to date recorded a traffic of nearly 1.1 terabits from more than 150,000 vulnerable Internet-connected cameras and digital video recorders [13], and it is well-known that a botnet built from a malware called Mirai [14] was used for conducting these attacks. This is not an isolated situation: in [15], the authors used honeypot and sandbox systems to show that a significant number of IoT devices are compromised and are targets of malware infection.

The current insecure situation is caused by two major factors: immature IoT devices and IoT's operational model implicitly derived from the Internet. IoT devices generally have handicaps such as severe constraints on resources and functionalities due to cost and/or size limitations, and thus it is difficult for IoT devices themselves to have enough security countermeasures. And even if the objects themselves are not constrained, like various consumer devices, the race to connect any consumer device to the Internet without proper security development, vulnerability testing, and protection mechanisms is leading to an

“Internet of (vulnerable) devices” [16]. Since the development of legal systems to improve IoT security is making little progress **[17]**, we should expect that immature and vulnerable devices will be around at least for the time being. As for the existing IoT's operational model, it implicitly incorporates the concept of the Internet's operational model, namely end-to-end connectivity. Since end-to-end connectivity concept requires every entity in the network to seamlessly communicate each other, the primary focus in the design and development of IoT devices has been the smooth connection and communication between the devices and the Internet. Such design focus has exposed IoT devices to various threats.

To address this growing threat to immature IoT devices, we have re-examined the present IoT's operational model and have proposed the *societal model* [18], a simple but robust operational model which has built-in security. In this model, IoT devices are explicitly protected by their designated guardian devices and isolated from the Internet and potential threats. This concept is borrowed from human society; there will be infants, the weak, and the handicapped who will need the support and protection by guardians. This natural security mechanism in human society works very well for IoT networks which seem to have inherently weak security mechanisms.

In this paper, we investigate the requirements of the societal model and discuss its feasibility. The main contributions of this paper are summarized as follows:

- We detail the societal model and the appropriate network architecture for it.
- We clarify the core and subsidiary requirements of the model.
- We show that constructs of the Internet standard management framework can be used to develop practical solutions that fulfill the core requirements.
- We provide a proof-of-concept implementation of the model and establish the feasibility of the model through experiments using the proof-of-concept implementation.
- We discuss the characteristics and downside of the model.

2. Related Work

In the last years, there have been various researchers [2], [8], [19] that have explored the main security and privacy issues of the IoT. One of the conclusions of this research is that IoT objects are extremely vulnerable against attacks, not only due to factors such as the challenge of updating a myriad of things, or their overall lack of resources, but also due to their inherent Internet connectivity – which opens the avenue to subtle (e.g. vulnerability exploitation) and not-so-subtle (e.g. DoS) attacks from anywhere, anytime. Another conclusion refers to the need of developing a unified vision that could satisfy the security and privacy requirements of this very heterogeneous environment. Such vision should allow the integration of protection mechanisms (from secure communications to intrusion detection, trust, and others) that are independent of the exploited platform.

One clear example of the vulnerable nature of things and these security issues can be found in the consumer area. For example, the Open Web Application Security Projects (OWASP) has described several concerns about the insufficient security of consumer IoT devices, and enumerated the main attack surface areas and top 10 IoT vulnerabilities in [20]. Besides, there have been various works, including [7], [21], that have analyzed the security of various consumer-level IoT devices such as TVs, webcams, home thermostats, and door locks.

According to these surveys, the average number of vulnerabilities found per device was significantly high. In fact, the devices were found vulnerable to a wide range of attacks from Heartbleed to denial of service attacks, weak passwords, and cross-site scripting attacks.

There have been a multitude of research works whose main goal is to assure end-to-end security between IoT objects and Internet entities, such as the standardization works of the IETF Security working groups [3], [22], [23], plus other proof-of-concept implementations [24], [25]. On the other hand, other researchers are considering a different vision to solve this problem. In such vision, IoT objects should not be directly connected to the Internet due to security considerations. One of the earliest analyses of this vision is found in Alcaraz et al. [26], where after a theoretical analysis it is concluded that some Wireless Sensor Network (WSN) applications should not connect directly to the Internet. More recently, the need to deploy a security-oriented piece of hardware and software that should sit between an IoT device and the Internet – a "bump in the wire" – was defended by various authors, such as Alan Grau in IEEE Spectrum [27].

There have been certain approaches that have explored the concept of a protected IoT environment. One example of this is Body Sensor Networks or Body Area Networks, where all the sensors interface with the outer world via a Local Processing Unit (LPU), which acts as a "router" [28]. Naturally, the security solutions and the functionality of such systems are still limited to the acquisition of data [29]. Another example is the existing IoT middleware platforms, which make use of several patterns to provide services to external entities (e.g. Service-oriented Architectures (SoA), event-based architectures) [30]. However, many of these platforms did not prioritize the protection of the devices in their design, focusing mostly on the establishment of a secure communication channel between authenticated entities. In fact, Tiburski et al. discussed in [31] the need of creating a standard security architecture for SoA-based IoT middleware.

There are also several recent security mechanisms that are exploring the inclusion of a "security bridge" between the IoT objects and the Internet, as Alan Grau noted. Still, many of such mechanisms consider the "bridge" as a helper: rather than shielding the objects, they mostly assist on the integration of security mechanisms such as key negotiation [32] and authentication [33]. Other approaches seek to deploy such "bridges" to fully protect one specific feature. One example of this is a personal gateway, which allows users to decide which data can be shared with the external world through the integration of embedded privacy mechanisms [34]. A more aggressive approach is a central *security manager* [35], which provides software update, traffic filtering, and strong authentication. This manager is expected to be built on "bridges." Finally, other proposals have sought to improve the security of existing "bridges" – such as middleware platforms. One example is the extension proposed by Chen et al., which extends the original ITU-T M2M gateway application with a secure gateway element that provides mutual authentication and key exchange procedure [36].

It is then clear that, despite the significant amount of work in the area of security for IoT networks, and the several solutions that have been proposed, investigated, implemented and deployed, it is still necessary to conceptually explore the existence of a comprehensive solution that addresses the generic needs of protected IoT devices. It is in this context where we have proposed our societal model for IoT security.

3. Societal model for securing IoT

3.1 Concept

Our societal model is a simple operational model where the basic concept is borrowed from human society. Considering the survival and growth of human race, it appears that some form of security that has ensured survival is built-in. While reproduction is a biological factor that has assured that new members join the society, there is also a built-in mechanism that ensures that babies and infants are protected. Otherwise, they would not reach the reproductive age, the number would dwindle, and the race would become extinct.

In an ensemble that is as diverse as the human race, the survival mechanism cannot depend solely on individual effort or awareness but is ensured by rules, traditions, and conventions. An important aspect seems to be the awareness that infants need protection. The protection is provided by a designated group of members, e.g. parents or guardians. An underlying principle seems to be that the guardian will be the interface between the infant and the rest of the world.

To reflect this natural principle into the IoT world, the societal model introduces a designated guardian device, called IoT Guardian (*I-Guardian*) to protect IoT devices. The I-Guardian must have enough resources and functionalities. Users/applications will access IoT devices over the Internet via an I-Guardian.

In the following subsections, we discuss the appropriate underlying network architecture for the societal model and the details of the proposed operational model.

3.2 Network Architecture

The architecture for the societal model envisages a collection of IoT devices, a collection of IoT Applications (*I-Applications*), and a relatively smaller collection of I-Guardians. IoT devices are designed for a dedicated purpose and are connected in what we will call a ThingNet (*T-Net*). An I-Application resides on the Internet. Users/administrators will access their IoT device over the Internet using some I-Application to obtain information or to give instructions. The access to an IoT device from an I-Application must be possible only via the designated I-Guardian which serves the IoT device (cf. **Fig. 1**).

There are several strategies to integrate the existing Internet and Internet of Things. In [37], the authors discuss three major integration approaches of the Internet and WSN, which is one of the most important elements of the IoT concept, from the security point of view. The integration approaches are the following: Front-End Proxy solution, Gateway solution, and TCP/IP Overlay solution. Each approach is characterized by the similarity of the protocol stack of a sensor node to that of Internet hosts and the degree of isolation of WSN from the Internet. Every approach assumes the existence of a “bump in the wire” device, i.e. a base station located at the border between the Internet and WSN. The role of a base station is different in each approach.

In the Front-End proxy solution, a sensor node has an entirely different protocol stack from an Internet host and the base station provides a kind of proxy service to both networks. Two networks are completely isolated and direct interaction across the border is not possible at any level of the protocol stack. In the Gateway solution, they have a common application layer protocol while they use different lower layers protocols. The base station has to serve as an application layer gateway. Although two networks are isolated from the view point of underlying network infrastructure, they can exchange information directly based on the same application protocol. In the TCP/IP Overlay solution, a sensor node has the TCP/IP protocol stack and can communicate with Internet hosts. The base station acts as a router in the Internet. This solution allows sensor nodes and Internet hosts to interact each other, and thus two networks are not isolated.

Our approach is very similar to the Front-End proxy solution because the core requirement of the societal model requires IoT devices to be entirely isolated from the Internet. Like the base station in the Front-End proxy solution, I-Guardians reside on the border between the T-Net and the rest of the Internet, and their role is for enabling secure interaction between an I-Applications and an IoT device while keeping the isolation of the IoT device. I-Guardians must be equipped with enough resources for providing appropriate security measures, such as data encryption, authentication, and data integrity check.

The most crucial point that differentiates the societal model from the Front-End proxy solution for WSNs is that I-Guardian must handle the huge diversity of IoT devices. It needs to provide an interface for any type of IoT devices irrespective of the protocol stack of the target IoT device. The next subsection discusses the operational model and functional requirements to take care of the IoT devices' diversity.

3.3 Operational Model

The societal model must provide a simple and secure access to a wide variety of IoT devices while isolating the IoT devices from direct Internet access. The detailed operational requirements are as follows.

1. The I-Guardian must explicitly recognize every IoT device in the T-Net
2. An IoT device will communicate only with its designated I-Guardian
    a. IoT devices must not communicate with any device on the Internet
    b. Only in some special cases, an IoT device may need to interact with another device in the same T-Net under the supervision of I-Guardians.
3. Legitimate entities (I-Applications) will be able to access the services of IoT devices via the I-Guardian over the Internet
    a. I-Guardian must allow legitimate users to have legitimate access to their devices seamlessly
4. It must be possible to add new devices to the realm of an I-Guardian with relative ease. Even in the case where the device is a new type.

To satisfy the above requirements, we have proposed an operational model as illustrated in **Fig. 1**.

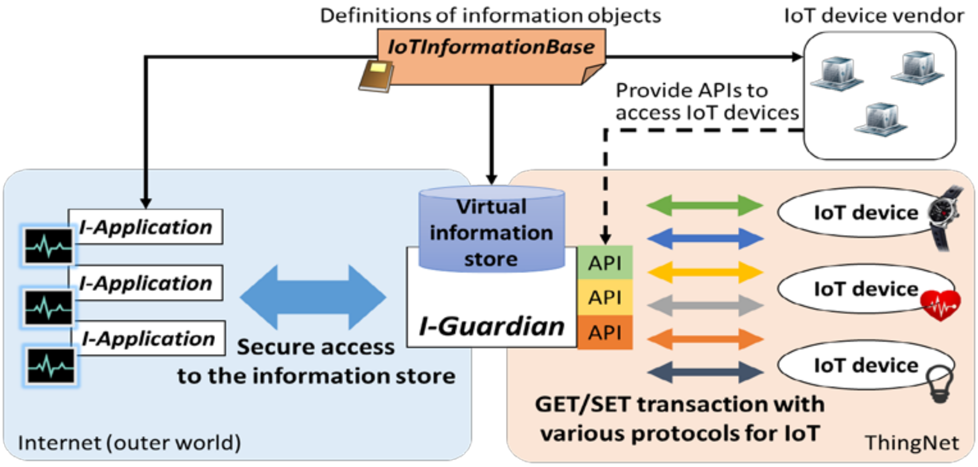

Definitions of information objects
IoTInformationBase
IoT device vendor
Provide APIs to access IoT devices
I-Application
I-Application
I-Application
Virtual information store
I-Guardian
API
API
API
IoT device
IoT device
IoT device
Secure access to the information store
GET/SET transaction with various protocols for IoT
Internet (outer world)
ThingNet

Fig. 1. Operational Model

An end-user of an IoT device must have an I-Guardian which has two types of interfaces: *O-Interfaces* and *T-Interfaces*. O-Interfaces are interfaces to the outer world and will be usually Ethernet, Wi-Fi, and/or LTE. T-Interfaces are interfaces to T-Net and have various options, such as Ethernet, Wi-Fi, RS-232C, Bluetooth, Zigbee. An IoT device will not be allowed to communicate with IoT devices in the basic design. All communications will happen via the corresponding I-Guardian, and all transactions are mediated by an I-Guardian. The I-Guardian device terminates every communication to and from a T-Net, and it vets the transaction and processes it only if the transaction satisfies the T-Net security requirements. For example, if the transaction originator cannot be authenticated, or the transaction is authenticated but is found to be harmful, it will be (silently) rejected.

Operations on an IoT device are modeled as inspections or modifications of some information components on the IoT device. This is a simple, device-independent, atomic model of IoT access on which complex operations may be based. The information component itself is any one of several pre-defined types. We will call this information component an *IoTInformationObject*. A collection of such objects will form a virtual information store, called *IoTInformationBase* (*IIB*). The definition of the IIB must be shared among concerned parties: the vendor of an IoT device, I-Application developers, and I-Guardian developers.

The vendor of an IoT device will develop a definition (name, type, etc.) of each IoTInformationObject for her device and make it available for I-Application developers and I-Guardian developers. The vendor also provides some application program interfaces (APIs) corresponding to the IIB module so that I-Guardian vendors can provide the instrumentation for accessing specific IoT devices available to their I-Guardians.

An I-Guardian instantiates all or part of an IIB. It collects information from IoT devices and stores them in the IIB. Applications may also manipulate some information components in the IIB. The I-Guardian controls IoT devices in accordance with the information components in the IIB. The collection of information and the control of IoT devices will be done through APIs provided by the vendor.

An I-Application will indirectly interact with an IoT device by accessing corresponding IoTInformationObjects in the IIB. Each IoTInformationObject in the IIB is an abstraction of some facet of an IoT device. In the proposed operational model, inspections and modifications of IoTInformationObjects by an I-Application are mapped on to monitoring and configuring of IoT devices.

An I-Guardian must provide I-Applications with secure universal access to IoTInformationObjects. The transaction between an I-Application and I-Guardian is done based on the name and type of an IoTInformationObject. Once the I-Guardian receives a request pertaining to an IoTInformationObject, the I-Guardian may access the corresponding IoT device by using the instrumentation if the IoTInformationObject instance is not in the IIB.

The proposed simple operational model accommodates the potential diversity inside a T-Net and can give a unified view of IoT devices to I-Applications. The actual method for a transaction between an I-Guardian and correspondent IoT devices will depend on the underlying networking technologies in T-Net and the protocol stack supported by target IoT devices. For example, an IoT device may support Constrained Application Protocol (CoAP) [38] over 6LoWPAN, another device may support MQ Telemetry Transport (MQTT) [39] over WiFi, yet another device may be running a complete TCP/IP protocol stack. Also, new protocols, technologies, and devices are likely to emerge. When a new type of device is connected to the T-Net, the I-Guardian can accommodate it only by obtaining

- definitions of IoTInformationObjects for the device
- corresponding APIs to access the device

from the vendor and preparing related instrumentation using the APIs. An I-Application does not need to bother about the wide variation of IoT devices that it may need to access.

3.4 Use case scenario

In this section, we describe a use case scenario of the proposed operational model by envisaging a smart home application.

The administrator of a smart home network, who may be a member of the family or a service engineer carrying out installation/maintenance work, must set up an I-Guardian device and connect it to the home network by using its O-Interface. The administrator also manages credentials that will be used by an I-Application to access the IoT via the I-Guardian. The credentials will be known to the I-Application (user) and will be registered with the I-Guardian.

When an end-user buys an IoT device, he/she registers the device's information with the I-Guardian and also registers the I-Guardian's information with the IoT device. After that the user connects the device to the I-Guardian through one of the T-Interfaces, the I-Guardian accesses the web site of the vendor of the connected IoT device and downloads relevant APIs and definitions of IoTInformationObjects for the IoT device. We assume that the vendor will properly authenticate itself through the use of trusted certificates, thus this exchange of information will be protected by existing communication standards such as TLS. An I-Application will interact with the I-Guardian to learn about newly available IoT devices and securely download the definitions of IoTInformationObjects for the IoT device from the vendor's web site.

When a user logs in and manipulates an I-Application in order to monitor and configure IoT devices, the I-Application sends requests to the I-Guardian to access the corresponding IIB objects. The I-Guardian validates the request in order to confirm that it is a valid request issued by a valid I-Application (user). The request is processed only when the request is valid. In this manner, a user will be able to carry out a desired operation such as knowing the current temperature of his/her living room, open/close a window in the bedroom.

3.5 Functional Requirements

We discuss the functional components required to realize the proposed societal model-based network. There are three core requirements (1, 2, 3) and two subsidiary requirements (4, 5). The core requirements are vital in any situation regardless of the size of a T-Net. The subsidiary requirements will be important when the size of a T-Net is large.

1) A Scalable Virtual Information Store: IIB, a virtual IoT information store that forms the core of the societal model, must be scalable, extensible, and maintainable in a multivendor, distributed environment. This is because IIB must accommodate a wide variety of information components handled by a potentially large number of diverse IoT devices. To handle various types of many information components, we will need an extensible and unique name space that scales globally from the operational and maintenance point of view.

Each IoTInformationObject will have a name, syntax and corresponding semantics. I-Applications will refer to an IoTInformationObject's name and will collect and/or set the corresponding value. Therefore, a language to define the corresponding value, its syntax and semantics, is required. This is currently a hot topic in the academia, as IoT interoperability is one of the most important factors for the advancement of this area [40].

2) A Universal Secure Access for the Virtual Information Store: A protocol will be required for communication of IoT-related information in the IIB between an I-Application and an I-Guardian. Since operations on IoT devices are modeled as inspections or modifications of some "value" of the corresponding IoTInformationObject, the protocol operations would also be modeled as simple GET and/or SET functions.

Besides, the access protocol must be equipped with appropriate security measures, such as strong data encryption, flexible authentication, and access control.

The protocol should be widely available for I-Applications and I-Guardians. The wide availability of the protocol is an important requirement for the development of I-Applications.

3) An Alert Mechanism: It is common for everyday applications to raise alarms that may be monitored by management applications and/or administrators/users. IoT devices must have a mechanism to alert an I-Guardian. The I-Guardian device will then use appropriate mechanisms to alert the designated Network Monitoring Systems or administrators.

The alert mechanism must meet the basic security requirements, namely, confidentiality, integrity, availability, accountability, authenticity, and non-repudiation. It must also have provisions for describing the alert in terms of IoTInformationObjects corresponding to the IoT device. Besides, the protocol for sending alerts should be widely available for I-Applications and I-Guardians.

4) A Membership Management Mechanism for ThingNet: Membership management is required in order to prevent rogue devices from accessing the T-Net and to allow valid devices to access I-Guardians.

When a new IoT device joins a T-Net, it will be explicitly registered with the corresponding I-Guardian by an administrator/user. A similar process will be done when a member leaves the T-Net; the IoT device will be explicitly de-registered. For a small-scale and relatively static T-Net, such registration and de-registration processes for membership management can be done manually. We consider this requirement to be a subsidiary one.

As more IoT devices join the T-Net, the demand for efficient membership management mechanism will grow. The registration and de-registration processes should employ membership verification based on strict authentication; it must not be based on easily spoofable identities like IP address and/or MAC address. The mechanism must be strong and robust enough to ensure that a non-member will not have any access to members in the T-Net.

One of the techniques of interest in this area is the "resurrecting duckling" scheme proposed in [41].

5) Group Security Mechanisms inside ThingNet: A T-Net must have the following mechanisms to provide group security for T-Net members.

- Ensuring that IoT devices communicate only with the designated I-Guardian.
- Detecting and notifying attempts of IoT devices to communicate with devices other than the designated I-Guardian.
- Ensuring that only known (member) devices are present in the T-Net.
- Ensuring that the designated I-Guardian is authentic.

I-Guardian and IoT devices must collaborate to realize above mechanisms. IoT devices and the designated I-Guardian will be made aware of each other through some registration process and member IoT devices will communicate only with the designated I-Guardian. If a member

IoT device notices that another IoT device is attempting to communicate with a device other than the I-Guardian, the IoT device should log that event and/or alert the I-Guardian. An advanced IoT device with appropriate capability may attempt to block such illegal communication. Moreover, if IoT devices have enough resources, it might be possible to integrate intrusion detection systems to monitor the behaviour of the T-Net and to detect anomalous behaviour [42]. As technology develops and the related laws mature, we will expect more and more IoT devices with the functionalities described here.

On the other hand, for IoT devices with insufficient resource and functionalities, some existing networking technologies may be readily employed. For example, control of packet-flow by utilizing layer 2 mechanisms, such as Address Resolution Protocol in IPv4 and Neighbour Discovery Protocol in IPv6, may be used to support group security inside the T-Net.

Last but not least, the I-Guardian itself might also implement various security services that continuously look after the security and resilience of the network during its existence. Including the integrated intrusion detection system mentioned above, examples of such services include continuous vulnerability assessment tools (e.g. “pentesting”), device integrity mechanisms (e.g. attestation), automatic patching tools, fault tolerance assistants (e.g. identification of IoT devices that can replace the functionality of a failing device), and many others. Such services will strengthen the overall group security and robustness of the whole T-Net.

We consider this requirement to be a subsidiary one.

4. Feasibility of the Societal Model

In this section, we discuss the feasibility of our proposal through a proof-of-concept implementation based on the Internet standard management framework.

4.1 Internet Standard Management Framework

Since the early days of the Internet, researchers and engineers have been working on the challenging issue of a management architecture where in all networked (and other) devices could be managed in an open, extensible, scalable, and secure framework. The problem has great similarities with the issues related to the societal model for IoT systems described above. We can use all the features of the network management framework to securely access and manipulate IoT devices. This is the reason why we choose the Internet standard management framework as the first option for the proof-of-concept implementation.

In the Internet standard network management framework [43], a Managed Object (MO), an information component representing a specific aspect of a managed device, is accessed via a virtual information store called the Management Information Base or MIB. In the MIB, MOs are represented as nodes in a tree-structure (MIB tree), and a MIB module (a related set of MOs) forms a sub-tree of the MIB tree. We can uniquely identify a particular MO with the object identifier, which is the sequence of the integer numbers assigned to every node on the path from the root to the node representing the target MO. New MOs can be defined using the Structure of Management Information (SMI) [44] and are added as the sub-tree to the MIB tree. The MIB has a distributed scalable and flexible framework that allows vendors to possess and maintain their own name space. Consequently, the MIB provides an extensible information store with the globally unique naming scheme.

The MOs in the MIB are accessed using the Simple Network Management Protocol (SNMP) [45], [46]. The protocol provides simple operations, GET, SET, NOTIFY and a few variations of these operations to access the MOs. A side effect of this scheme is that, all management tasks such as monitoring and configuration on the managed devices are carried out via *SNMP agents* on the devices. Only the agent has access to the MIB, and management applications do not have "direct" access to the managed devices/entities. The managed devices are shielded by the agent which is expected to carry out the security procedures before acting on a request from a management application. This aspect serves the core requirement of protection for the IoT devices in the societal model.

The latest version of the Framework (the SNMPv3 Framework) [43] has robust security mechanisms. The framework defines a User-based Security Model (USM) [47] and a View-based Access Control Model (VACM) [48]. These mechanisms provide important security services: user-based authentication, data integrity and confidentiality, and fine-grained access control. These are useful building blocks that make the societal model feasible.

4.2 Overview of Proof-of-Concept Implementation

Here, we explain that how each of the following core requirements of the societal model is realized in the proof-of-concept implementation.

- A scalable virtual information store
- A universal secure access for the virtual information store
- An alert mechanism

1) A Scalable Virtual Information Store: The IoTInformationBase (IIB), a virtual information store for modeling IoT devices, is defined as a MIB module. An IoT device is modeled as a set of IoTInformationObjects in the MIB. The SNMP agent, which serves as the I-Guardian in the societal model context, provides access to IoT devices via the IIB. Each IoTInformationObject is named and defined using SMI constructs in a globally unique name space.

We have designed the IIB which is capable to handle a wide variety information components of various devices. The detailed design of the IIB is discussed in the next section.

2) A Universal Secure Access for the Virtual Information Store: We assume that an I-Application in the Internet possesses SNMP APIs and accesses the Information Objects in the virtual information store by interacting with an I-Guardian using the SNMP protocol constructs such as GET, SET, and NOTIFY.

Security mechanisms, authenticity, confidentiality, integrity, and access control for the outer world are handled by the I-Guardian using mechanisms made available in the USM and VACM in the SNMPv3 framework.

3) An Alert Mechanism: An alert mechanism is realized using the SNMPv3 INFORM mechanism, the asynchronous notification mechanism available in SNMP. The SNMPv3 INFORM mechanism supports acknowledgements from the receiver. It also offers authentication and encryption. Therefore, secure and reliable alerting is possible.

When an I-Guardian receives an alert from an IoT device, the I-Guardian generates an SNMPv3 INFORM message including the received alert information and sends it to the corresponding I-Applications and/or the designated Network Management System of the T-Net. The I-Application and NMS need to be equipped with the receiver of an INFORM message.

4.3 Design of the IoTInformationBase module

Fig. 2 illustrates the tree structure of the IIB module designed in this research. The module consists of following three groups. Objects in each group are enumerated in **Tables 1-3**. The usage example of each group is illustrated in **Fig. 3**.

- *ConfObjects* group representing I-Guardian's configuration information (**Table 1**)
- *DeviceTable* group abstracting IoT devices (**Table 2**)
- *DataTable* group abstracting data on IoT devices (**Table 3**)

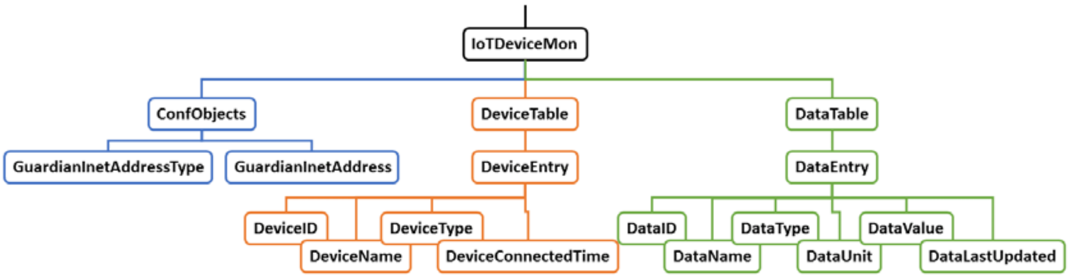


Fig. 2. IoTInformationBase Sub Tree

Table 1. Configuration Objects of I-Guardian

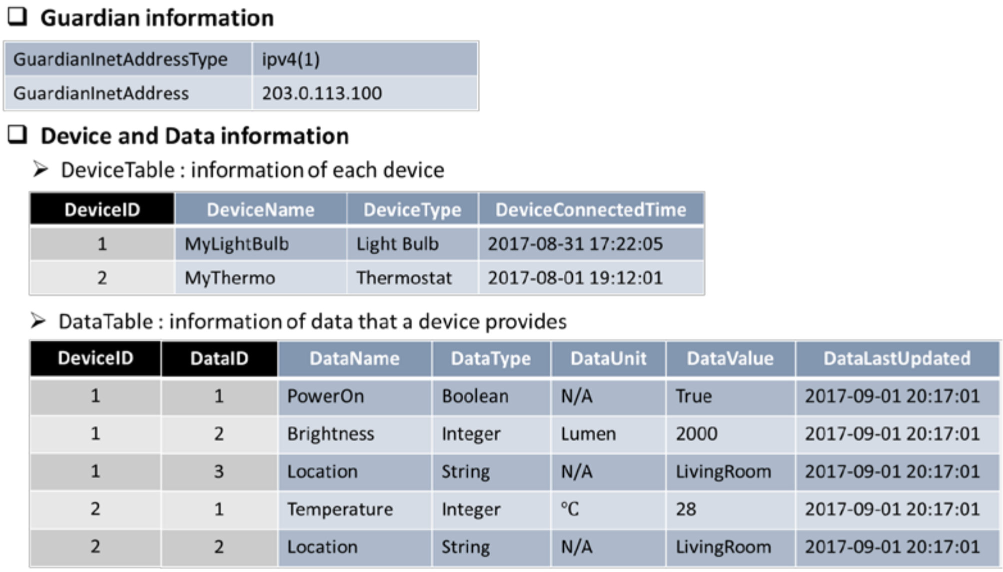
Object	Type	Description
GuardianInetAddressType	Address type	The type of address of IoT devices' guardian
GuardianInetAddress	IP address	IP address of IoT devices' guardian

Table 2. Objects in DeviceTable

Object	Type	Description
DeviceID	Integer	An index to uniquely identify each IoT device
DeviceName	String	The name of the monitored IoT device
DeviceType	String	The type of the monitored IoT device
DeviceConnectedTime	Time	The time at which the monitored IoT device is connected.

Table 3. Objects in DataTable

Object	Type	Description
DataID	Integer	An index to uniquely identify each data component on a given IoT device
DataName	String	The name of data to be monitored or configure
DataType	String	The type of data to be monitored or configure
DataUnit	String	The unit of data to be monitored or configure. E.g., celsius, lumen, second, etc.
DataValue	String	The actual data to be monitored and/or configured
DataLastUpdated	Time	The time at which the data is last updated



❑ **Guardian information**

GuardianInetAddressType	ipv4(1)
GuardianInetAddress	203.0.113.100

❑ **Device and Data information**

➢ DeviceTable : information of each device

DeviceID	DeviceName	DeviceType	DeviceConnectedTime
1	MyLightBulb	Light Bulb	2017-08-31 17:22:05
2	MyThermo	Thermostat	2017-08-01 19:12:01

➢ DataTable : information of data that a device provides

DeviceID	DataID	DataName	DataType	DataUnit	DataValue	DataLastUpdated
1	1	PowerOn	Boolean	N/A	True	2017-09-01 20:17:01
1	2	Brightness	Integer	Lumen	2000	2017-09-01 20:17:01
1	3	Location	String	N/A	LivingRoom	2017-09-01 20:17:01
2	1	Temperature	Integer	℃	28	2017-09-01 20:17:01
2	2	Location	String	N/A	LivingRoom	2017-09-01 20:17:01

Fig. 3. Usage Example of the IoTInformationBase module

Each entry in *DeviceTable* corresponds to the attribute information of an IoT device protected by an I-Guardian. Each entry in *DataTable* corresponds to the attribute information of each data element to be monitored or configured on a specific IoT device. The index of the DataTable is composed of *DeviceID* and *DataID*. An I-Application can access a specific data element on a specific IoT device by referring to its *DeviceID* and *DataID*.

Each data element is represented by a tuple of four string objects: *DataName*, *DataType*, *DataUnit*, and *DataValue*. Since the type of *DataValue* is String, this object can hold any value that can be represented with a string of ASCII characters. By this design, we believe that this IIB module can be utilized for various types of IoT devices and IoT device vendors usually do not have to design their own IIB module.

4.4 Experimental Results

Fig. 4 shows an overview of our proof-of-concept implementation.

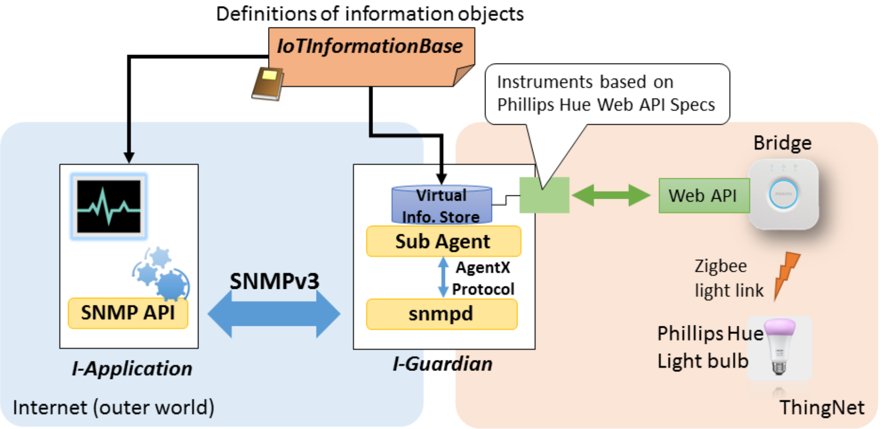


Fig. 4. Implementation Overview

We implemented the IIB module and an I-Guardian's functionalities using NetSNMP [49], a widely available reference implementation of SNMP. We implemented a sub agent accessing the virtual information store based on the definition given in Sec. 4.3 using NetSNMP Perl Module [50]. The sub agent communicates with the master agent (snmpd) using Agent Extensibility (AgentX) protocol [51]. These two agents together play the role of an I-Guardian.

We used a Phillips Hue Light bulb [52] as a target IoT device. This LED light bulb can be connected to a network via a bridge device. The bridge device is equipped with a Web API. A user can monitor and control the bulb via HTTP. Since Phillips has published the specifications of the Web API [53], we can implement the instrumentation on an I-Guardian for accessing the light bulb. In this experiment, we assumed that the addition and removal of members from a T-Net group are manually handled by a T-Net administrator. We also assumed that some simple connection management mechanisms to prevent rogue devices from connecting the T-Net, are provided as a group security mechanism.

We envisaged that a user will monitor and control the light bulb in her smart home using an I-Application. The I-Application sends appropriate SNMP requests using SNMP API along with the user's authentication information to the I-Guardian. The I-Guardian confirms the authenticity and corresponding authorization of the user. If the authenticated user has appropriate authority, the I-Guardian will attempt to service the request. Otherwise, the request is silently ignored. We have confirmed that an authentic user can monitor the status of IoT devices seamlessly and securely and unauthorized attempts are blocked by the security mechanisms. We have also confirmed that the status change of the light bulb is alerted to the I-Application as an SNMPv3 INFORM message.

5. Considerations

5.1 Pros and Cons of the Societal Model

1) Pros: One of the major implications of the societal model is that the focus of security shifts from the numerous, security-wise mostly immature, IoT devices to a small number of security-wise mature I-Guardians. As IoT devices cannot be accessed directly through the Internet, they cannot be remotely queried (e.g. using search engines like SHODAN [54]) and exploited. This way, the attack surface is reduced, as I-Guardians become the only remote target – IoT devices can only be directly attacked if the adversary is located in the T-Net. Therefore, developers, manufacturers, and network administrators can focus their time and effort to patch, update and upgrade I-Guardians’ software and hardware. Moreover, the I-Guardian can facilitate the automatic management and dissemination of the security patches for the IoT devices located in their T-Net.

Besides, the societal model aids the traceability of IoT devices in IoT networks. All IoT devices must be registered with an I-Guardian, to be accessible. An IoT device is not allowed to directly interact with any other entity. The traceability will contribute to reducing blind spots and to improving the security of IoT systems.

Moreover, due to this holistic point of view that I-Guardians have of the IoT networks they manage, not only they can actively and passively monitor and manage the security of all IoT devices, but they can also implement additional services that facilitate the orchestration of the functionality of the whole T-Net, enhancing its resilience and robustness.

2) Cons: Since all transactions must be checked and validated by the I-Guardian, the “realtimeness” of the transactions will be impacted. In order to minimize the delay incurred by the validation, an I-Guardian must be carefully designed to have enough resources (CPU, memory, etc.) for handling a required number of transactions. In addition, the size of T-Net should be determined based on the required level of the realtimeness.

Another important element to consider is that an I-Guardian is a single point of failure. The entire T-Net will be unavailable if the I-Guardian fails. There are several challenges here. Without appropriate authentication mechanisms in place, a phoney I-Guardian may assume control over the T-Net with potentially grave consequences. If an I-Guardian succumbs to a DDoS attack, then the T-Net managed by the I-Guardian will be inaccessible. External adversaries will attempt to exploit I-Guardian’s vulnerabilities to gain access to the T-Net. If attackers can gain physical access to an I-Guardian, they may be able to gain control of the I-Guardian or at the least make it unavailable.

I-Guardians will be the focus of security and will require utmost care and consideration. Its role in the network will be just as important as other Internet-facing devices with critical roles such as servers, routers. Therefore, depending on the criticality of the T-Net, an I-Guardian may need external protection services like DDoS mitigation service. As for the I-Guardian itself, it must be properly configured and hardened (i.e. there must be no redundant services), and it must include appropriate resources and security functionalities; for example, secure software/firmware update mechanisms, packet filtering, antivirus, application firewall, and intrusion detection and prevention mechanisms, amongst others.

However, any security mechanism at a network level will become useless without implementing proper physical security for the I-Guardian. I-Guardians must be protected against unauthorised physical access. Locks, barriers, biometrics-based access control are examples of physical protection mechanisms. I-Guardians (either as physical machines or as virtualized components) can also be deployed in safeguarded environments (e.g. cloud servers, data centers, fog and mobile edge computing nodes). Besides, more advanced protection mechanisms, such as secure boot and secure attestation using secure elements and trusted platform modules (TPMs), might also be used to control the integrity of the I-Guardian [55]. It is also important that the I-Guardian be manageable by some network management system (NMS). The NMS will monitor the availability and operation of the I-Guardian. Also, the I-Guardian will notify the NMS in case it detects some event.

Furthermore, it is vital that every communication to and from the I-Guardian is authenticated and secured. For example, the I-Guardian must interact with IoT device vendors through protected channels equipped with appropriate authentication of the origin and validation of the integrity of the received information in order to prevent an I-Guardian from accepting malicious information from a bogus and/or compromised vendor.

5.2 Differences from Traditional “bump in the wire” Implementation

An I-Guardian is a “bump in the wire (BITW)” device, but its role is completely different from the role of the traditional one.

The BITW concept was originally introduced in the IPsec specification [56]. A dedicated in-line device, which is called a security gateway, provides transparent IPsec-protection to legacy devices that cannot possess IPsec functionalities inside themselves. As explained in [27], this concept has been imported into the IoT security area, and some vendors propose and develop IoT security gateways (IoT-SGs) that provide typical security mechanisms such as firewall and intrusion detection/prevention. IoT-SGs check a packet to the target IoT device whether the packet comes from trusted entities and does not include malicious contents. They transparently forward the packet to the target device only when the packet fulfills the above-mentioned conditions.

While the traditional BITW concept puts emphasis on the transparent operation, the societal model does not. Traditional firewalls and IDS/IPS intervene only if and when they find something unappropriate in the communication. Otherwise, the communication is untouched. In contrast, in the societal model, the I-Guardian is in charge playing an active role in all interactions between IoT devices and I-Applications. IoT devices are allowed to communicate only with their I-Guardian. I-Applications have to communicate with the corresponding I-Guardian of the target IoT devices. All transactions are always terminated at an I-Guardian. Packets are not allowed to pass through beyond the I-Guardian to an IoT device under any circumstances. Therefore, the security scheme provided by the societal model is not transparent to both IoT devices and I-Applications.

5.3 Deployment Scenario of I-Guardians

There are several choices for the location of an I-Guardian as shown in **Fig. 5**.

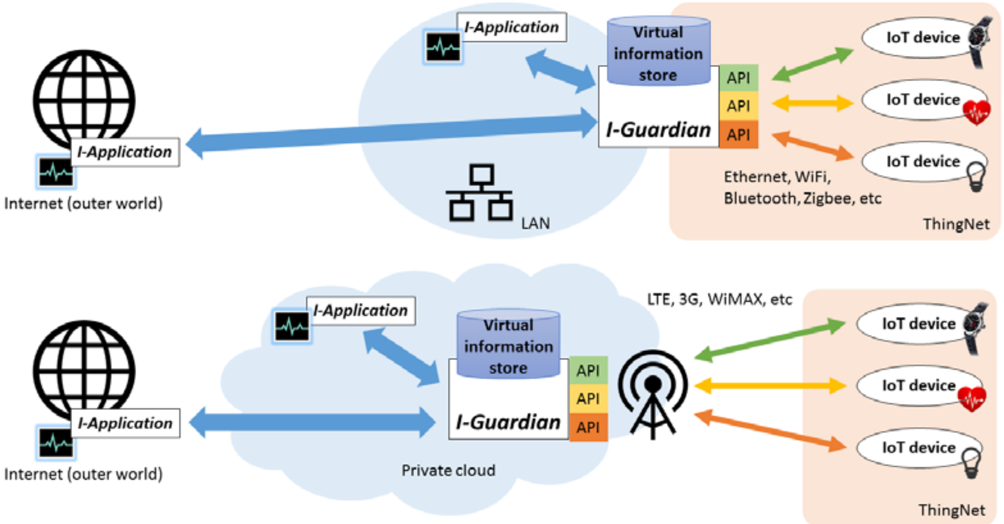


Fig. 5. Deployment Scenarios

In the simplest case, IoT devices connect to existing LANs and communicate with entities on the Internet. In this case, an I-Guardian is deployed on a physical device and must be the entry point for the existing LAN that connects the IoT devices. The I-Guardian will act as the default gateway and/or the wireless access point for IoT devices.

Recently, certain advanced services have emerged, wherein IoT devices connect to either a private cloud or a local cloud computing environment (i.e. Fog Computing) directly via LTE/3G connections without wading through the public Internet [57], [58]. In such the advanced cases, an I-Guardian will be deployed as a service on the cloud and must be at the cloud entry point for the IoT devices.

5.4 Advanced I-Applications

The societal model eases the development of advanced I-Applications which need to interact with multiple T-Net and/or IoT devices in a seamless and inter-operable manner. As a simple example, imagine an advanced I-Application in a personalized meal recommendation service. The application will recommend its user the healthiest and reasonable meal taking into consideration various factors, such as user's physical condition, food availability, consumption history of the user. In order to obtain related information, the application may need to interact with a weighing scale, wearable activity monitors, continuous glucose monitors, refrigerator, etc. In the societal model, the designated I-Guardian of each of above devices provides the application a unified way to access relevant information. The application can seamlessly obtain various data.

Another type of advanced I-Applications may need efficient mechanisms to access large volumes of data, such as various sets of long-time measurement data. The issue of efficient and accurate data collection has been examined in the network management arena [59]. For efficient data acquisition using SNMP, the authors have shown a managed object aggregation MIB [60], a technique to build complex aggregate MOs from simple MOs. This technique may be conveniently used to improve performance in cases where multiple instances of multiple objects need to be accessed periodically.

6. Conclusion

In this paper, we have discussed the security aspects of Internet of Things (IoT), proposed a societal model as a simple operational model that provides enhanced security, and assessed the feasibility of the proposal. The societal model does look attractive with security risks greatly reduced by moving the onus of handling security related matters from the potentially resource-constrained IoT device to a security proficient guardian device. The core requirements of the societal model can be fulfilled using existing technologies available in the Internet standard network management framework. We have given the actual design of the management information base that is used to monitor and control IoT devices, presented a proof-of-concept implementation, and confirmed that an actual off-the-shelf IoT device can be securely monitored and controlled within the proposed operational framework of the model.

Security is a moving goal. At no point of time can we expect all the aspects of security to be fully understood and corresponding countermeasures to be in place. In this context, our proposal improves the protection of IoT devices. Guardian devices will handle security matters and services (e.g. intrusion detection) and, as such, security patches, fixes and updates will be carried out on the guardian(s). The IoT devices, some of which may be hidden out of sight and out of mind, will not be expected and/or required to be patched/secured/upgraded frequently. We believe this will be a significant advantage of delegating the security to the guardian(s).

We believe that security management of IoT devices based on the societal model will make society safer. In this paper, we limited ourselves to discussing only the simplest model where an IoT device can only communicate with its guardian in order to build the most secure environment. More advanced and useful designs where IoT devices will communicate with each other within a ThingNet will bring more security threats and will need further considerations. Moreover, the integration of additional security services and the applicability of the societal model to existing IoT platforms and standards will also be explored in future works.

Acknowledgements

The authors express our thanks to Dr. Ruben Rios and Dr. Ana Nieto for their valuable and constructive suggestions.

References

[1] Hiroshi Tsunoda and Glenn Mansfield Keeni, “Feasibility of societal model for securing Internet of Things,” in *Proc. of 13th International Wireless Communications and Mobile Computing Conference (IWCMC2017)*, pages 541–546, Valencia, 2017. Article (CrossRef Link)

[2] Sabrina Sicari, Alessandra Rizzardi, Luigi Alfredo Grieco, and Alberto Coen-Porisini, “Security, privacy and trust in Internet of Things: The road ahead,” *Computer Networks*, 76:146–164, 2015. Article (CrossRef Link)

[3] Sye Loong Keoh, Sandeep S. Kumar, and Hannes Tschofenig, “Securing the internet of things: A standardization perspective,” *IEEE Internet of Things Journal*, 1(3):265–275, 2014. Article (CrossRef Link)

[4] Mandeep Khera, “Think Like a Hacker,” *Journal of Diabetes Science and Technology*, 11(2):207–212, 2017. Article (CrossRef Link)

[5] Andy Greenberg, “This Gadget Hacks GM Cars to Locate, Unlock, and Start Them (UPDATED),” 2015, Article (CrossRef Link).

[6] Troy Hunt, “Controlling vehicle features of Nissan LEAFs across the globe via vulnerable APIs,” Article (CrossRef Link), 2016.

[7] Nan Zhang, Soteris Demetriou, Xianghang Mi, Wenrui Diao, Kan Yuan, Peiyuan Zong, Feng Qian, XiaoFeng Wang, Kai Chen, Yuan Tian, Carl A. Gunter, Kehuan Zhang, Patrick Tague, and Yue-Hsun Lin, “Understanding IoT Security Through the Data Crystal Ball: Where we are Now and Where we are Going to Be,” 2017. Available online arXiv:1703.09809.

[8] Ahmad-Reza Sadeghi, Chrstian Wachsmann, and Michael Waidner, “Security and privacy challenges in industrial Internet of Things,” in *Proc. of 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, pages 1–6, June 2015. Article (CrossRef Link)

[9] Lorena *Cazorla*, Cristina Alcaraz, and Javier Lopez, “Cyber Stealth Attacks in Critical Information Infrastructures,” *IEEE Systems Journal*, pages 1–15, 2016. Article (CrossRef Link)

[10] David De Cremer, Bang Nguyen, and Lyndon Simkin, “The integrity challenge of the Internet-of-Things (IoT): on understanding its dark side,” *Journal of Marketing Management*, 33(1-2):145–158, 2017. Article (CrossRef Link)

[11] DANIEL CID, “Large CCTV Botnet Leveraged in DDoS Attacks,” 2016. Article (CrossRef Link).

[12] DANIEL CID, “IoT Home Router Botnet Leveraged in Large DDoS Attack,” 2016. Article (CrossRef Link).

[13] Pierluigi Paganini, “150,000 IoT Devices behind the 1Tbps DDoS attack on OVH,” 2016. Article (CrossRef Link)

[14] Roger Hallman, Josiah Bryan, Geancarlo Palavicini, Joseph Divita, and Jose Romero-Mariona, “IoDDoS The Internet of Distributed Denial of Sevice Attacks A Case Study of the Mirai Malware and IoT - Based Botnets,” in *Proc. of the 2nd International Conference on Internet of Things, Big Data and Security* , April, 2017. Article (CrossRef Link)

[15] Yin Minn Pa Pa, Shogo Suzuki, Katsunari Yoshioka, Tsutomu Matsumoto, Takahiro Kasama, and Christian Rossow, "IoTPOT: Analysing the Rise of IoT Compromises," in *Proc. Of 9th USENIX Workshop on Offensive Technologies (WOOT 15)*. USENIX Association, 2015.

[16] Hypponen Mikko and Linus Nyman, "The Internet of (Vulnerable) Things: On Hypponen's Law, Security Engineering, and IoT Legislation," *Technology Innovation Management Review*, 7(4):5–11, 2017.

[17] Bruce Schneier, "IoT Security: What's Plan B?" *IEEE Security & Privacy*, 15(5):96, 2017. Article (CrossRef Link)

[18] Hiroshi Tsunoda and Glenn Mansfield Keeni, "Societal Model for Securing Internet of Things," in *Proc. of International Conference on Business and Industrial Research*, pages 220–225. Thai-Nichi Institute of Technology, 2016.

[19] Ke Xu, Yi Qu, and Kun Yang, "A Tutorial on the Internet of Things: From a Heterogeneous Network Integration Perspective," *IEEE Network*, 30(2):102–108, 2016. Article (CrossRef Link)

[20] OWASP. OWASP Internet of Things Project. Article (CrossRef Link).

[21] Internet of things research study 2015 Report. Technical report, Hewlett Packard Enterprise, 2015. Article (CrossRef Link).

[22] Ari Keranen and Bormann Carsten, "Internet of Things: Standards and Guidance from the IETF," *IETF Journal*, 11(3), 2016.

[23] Zhengguo Sheng, Shusen Yang, Yifan Yu, Athanasios Vasilakos, Julie McCann, and Kin Leung, "A Survey on the IETF Protocol Suite for the Internet of Things: Standards, Challenges, and Opportunities," *IEEE Wireless Communications*, 20(6):91–98, 2013. Article (CrossRef Link)

[24] Giulio Peretti, Vishwas. Lakkundi, and Michele Zorzi, "BlinkToSCoAP: An End-to-end Security Framework for the Internet of Things," in *Proc. of 7th International Conference on Communication Systems and Networks (COMSNETS'15)*, pages 1–6, 2015. Article (CrossRef Link)

[25] ShahidRaza, Tómas Helgason, Panos Papadimitratos, and ThiemoVoig, "Securesense: End-to-end Secure Communication Architecture for the Cloud-connected Internet of Things," *Future Generation Computer Systems*, 77:40–51, 2017. Article (CrossRef Link)

[26] Cristina Alcaraz, Pablo Najera, Javier Lopez, and Rodrigo Roman, "Wireless Sensor Networks and the Internet of Things: Do We Need a Complete Integration?" in *Proc. of 1st International Workshop on the Security of the Internet of Things (SecIoT10)*, 2010.

[27] Alan Grau, "Can You Trust Your Fridge?" *IEEE Spectrum*, 52(3):51–56, 2015. Article (CrossRef Link)

[28] Riccardo Cavallari, Flavia Martelli, Ramona. Rosini, Chiara Buratti, and Roberto Verdone, "A Survey on Wireless Body Area Networks: Technologies and Design Challenges," *IEEE Communications Surveys and Tutorials*, 16(3):1635–1657, 2014. Article (CrossRef Link)

[29] Prosanta Gope and Tzonelih Hwang, "BSN-Care: A Secure IoT-Based Modern Healthcare System Using Body Sensor Network," *IEEE Sensors Journal*, 16(5):1368–1376, 2016. Article (CrossRef Link)

[30] Mohammad Abdur Razzaque, Marija Milojevic-Jevric, Andrei. Palade, and Siobhán Clarke, "Middleware for Internet of Things: A Survey," *IEEE Internet of Things Journal*, 3(1):70–95, 2016. Article (CrossRef Link)

[31] Ramo Tiago Tiburski, Leonardo Albernaz Amaral, Everton De Matos, and Fabiano Hessel, "The Importance of a Standard Security Architecture for SOA-based IoT Middleware," *IEEE Communications Magazine*, 53(12):20–26, 2015. Article (CrossRef Link)

[32] Pawani Porambage, An Braeken, Pardeep Kumar, Andrei Gurtov, and Mika Ylianttila, "Proxy-based End-to-end Key Establishment Protocol for the Internet of Things," in *Proc. of IEEE International Conference on Communication Workshop (ICCW'15)*, pages 2677–2682, June 2015. Article (CrossRef Link)

[33] Tobias Markmann, Thomas C. Schmidt, and Matthias Wählisch, “Federated End-to-End Authentication for the Constrained Internet of Things Using IBC and ECC,” *SIGCOMM Computer Commununication Review*, 45(4):603–604, 2015. Article (CrossRef Link)

[34] Martin Henze, Lars Hermerschmidt, Daniel Kerpen, Roger Huling, Bernhard Rumpe, and Klaus Wehrle, “A Comprehensive Approach to Privacy in the Cloud-based Internet of Things,” *Future Generation Computer Systems*, 56:701–718, 2016. Article (CrossRef Link)

[35] A. K. Simpson, F. Roesner, and T. Kohno, “Securing Vulnerable Home IoT Devices with an In-hub Security Manager,” The First International Workshop on Pervasive Smart Living Spaces (PerLS 2017) — in conjunction with IEEE PerCom 2017, 2017. Article (CrossRef Link)

[36] Hsing-Chung Chen, Ilsun You, Chien-Erh Weng, Chia-Hsin Cheng, and Yung-Fa Huang, “A Security Gateway Application for End-to-End M2M Communications,” *Computer Standards and Interfaces*, 44:85–93, 2016. Article (CrossRef Link)

[37] Rodrigo Roman and Javier Lopez, “Integrating Wireless Sensor Networks and the Internet: a Security Analysis,” Internet Research, 19(2):246–259, 2009. Article (CrossRef Link)

[38] Zach Shelby, Klaus Hartke, and Carsten Bormann, “The Constrained Application Protocol (CoAP),” RFC 7252, 2014. Article (CrossRef Link)

[39] Andrew Banks and Gupta Rahul, “MQTT Version 3.1.1 Plus Errata 01,” Article (CrossRef Link), 2015.

[40] Maria Ganzha, Marcin Paprzycki, Wiesław Pawłowski, Paweł Szmeja, and Katarzyna Wasielewska, “Towards Common Vocabulary for IoT Ecosystems—preliminary Considerations,” *Intelligent Information and Database Systems*, pages 35–45, 2017. Article (CrossRef Link)

[41] Frank Stajano and Ross Anderson, “The Resurrecting Duckling: Security Issues for Ubiquitous Computing,” *Computer*, 35(4):supl22–supl26, 4 2002. Article (CrossRef Link)

[42] Bruno Bogaz Zarpelo, Rodrigo Sanches Miani, Cludio Toshio Kawakani, and Sean Carlisto de Alvarenga, “A Survey of Intrusion Detection in Internet of Things,” *Journal of Network and Computer Applications*, 84:25–37, 2017. Article (CrossRef Link)

[43] Jeffrey Case, Russ Mundy, David Partain, and Bob Stewart, “Introduction and Applicability Statements for Internet-Standard Management Framework,” RFC 3410, 2002.
Article (CrossRef Link)

[44] Keith McCloghrie, Jrgen Schönwälder, David T Perkins, and Keith McCloghrie, “Structure of Management Information Version 2 (SMIv2),” RFC 2578, 1999. Article (CrossRef Link)

[45] Jeffrey D. Case, Mark Fedor, Martin L. Schoffstall, and James Davin, “Simple Network Management Protocol (SNMP),” RFC 1157, 5 1990. Article (CrossRef Link)

[46] Randy Presuhn, “Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)” RFC 3416, 2002. Article (CrossRef Link)

[47] Uri Blumenthal and Bert Wijnen, “User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3),” RFC 3414, 2002. Article (CrossRef Link)

[48] Bert Wijnen, Randy Presuhn, and Keith McCloghrie, “View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP),” RFC 3415, 2002.
Article (CrossRef Link)

[49] Net-SNMP. Article (CrossRef Link).

[50] Tut:Extending snmpd using perl. Article (CrossRef Link)

[51] Dale Francisco, Bert Wijnen, Mark Ellison, and Michael Daniele, “Agent Extensibility (AgentX) Protocol Version 1,” RFC 2741, 2000.

[52] Phillips, “Wireless and smart lighting by Phillips — Meet Hue,” Article (CrossRef Link).

[53] Phillips, “Phillips hue API — Phillips Hue API,” Article (CrossRef Link).

[54] Kai Simon, Cornelius Moucha, and Jörg Keller, "Contactless Vulnerability Analysis using Google and Shodan," *Journal of Universal Computer Science*, vol. 23, no. 4, 2017.

[55] Will Arthur, David Challener, "A Practical Guide to TPM 2.0: Using the Trusted Platform Module in the New Age of Security," *APress*, ISBN 978-1-4302-6583-2, 2015. Article (CrossRef Link)

[56] Karen Seo and Stephen Kent, "Security Architecture for the Internet Protocol," RFC 4301, 2005. Article (CrossRef Link)

[57] SORACOM, "SORACOM Overview" Article (CrossRef Link).

[58] Shanhe Yi, Cheng Li, and Qun Li, "A Survey of Fog Computing: Concepts, Applications and Issues," *in Proc. of the 2015 Workshop on Mobile Big Data, Mobidata '15*, pages 37–42, 2015. Article (CrossRef Link)

[59] Glenn Mansfield, Sandeep Karakala, Takeo Saito, and Norio Shiratori, "High Resolution Traffic Measurement," in *Proc. of Workshop on Passive And Active Measurements on the Internet (PAM2001)*, pages 67–73, 2001.

[60] Glenn Mansfield, "The Managed Object Aggregation MIB," RFC 4498, 2006. Article (CrossRef Link)

Hiroshi Tsunoda received his M.S. and Ph.D. degrees from the Graduate School of Information Sciences at Tohoku University in 2002 and 2005, respectively. From April 2005 to March 2008 he was an assistant professor in the Graduate School of Information Sciences, Tohoku University. He was a Visiting Researcher from Apri to December 2017 with the NICS Laboratory, University of Malaga. He is now an Asssociate Professor in the Department of Information and Communication Engineering, Tohoku Institute of Technology. His research interests include wireless networking, network management, and network security. He is a member of the IEICE, IPSJ, and IEEE.

Rodrigo Roman-Castro is a security researcher working at the University of Malaga (Spain), where he obtained his Ph.D. and M.Sc. degrees in Computer Engineering and Computer Science, respectively, in 2008 and 2003. Previously, he worked for the Institute of Infocomm Research (I2R) in Singapore in the areas of sensor network security and cloud security. Pursuing to make security simple and usable, his research is focused on the development of protection mechanisms for the Internet of Things and related paradigms like Edge Computing.

Javier Lopez is Full Professor in the Computer Science Department at the University of Malaga, and Head of the NICS Laboratory. His research activities are mainly focused on information security, future Internet security, and critical infrastructure protection, and has lead several international research projects in those areas. Prof. Lopez is Co-Editor in Chief of IJIS journal and the Spanish representative in the IFIP TC-11 on Security and Privacy Protection in Information Systems.

Glenn Mansfield Keeni received his Ph.D. degree in Information Engineering from Tohoku University, Japan. He is currently President/CEO of Cyber Solutions Inc. Sendai, Japan. His research interests include expert systems, computer networks, network management and network security. He is a member of the ACM, IEEE Computer Society and is an active member of the IETF.