Digital Communications and Networks 6 (2020) 195–202

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

A survey of edge computing-based designs for IoT security

Kewei Sha *, T. Andrew Yang, Wei Wei, Sadegh Davari

Department of Computing Sciences, University of Houston-Clear Lake, Houston, 77058, USA

ARTICLE INFO

Keywords:
Edge computing
Internet of Things (IoT)
Security
Architecture
Secure protocols
Firewall
Intrusion detection
Authentication
Authorization
Privacy

ABSTRACT

Pervasive IoT applications enable us to perceive, analyze, control, and optimize the traditional physical systems. Recently, security breaches in many IoT applications have indicated that IoT applications may put the physical systems at risk. Severe resource constraints and insufficient security design are two major causes of many security problems in IoT applications. As an extension of the cloud, the emerging edge computing with rich resources provides us a new venue to design and deploy novel security solutions for IoT applications. Although there are some research efforts in this area, edge-based security designs for IoT applications are still in its infancy. This paper aims to present a comprehensive survey of existing IoT security solutions at the edge layer as well as to inspire more edge-based IoT security designs. We first present an edge-centric IoT architecture. Then, we extensively review the edge-based IoT security research efforts in the context of security architecture designs, firewalls, intrusion detection systems, authentication and authorization protocols, and privacy-preserving mechanisms. Finally, we propose our insight into future research directions and open research issues.

1. Introduction

With recent developments in sensing, communication, and micro-controller technologies, we have been witnessing a convergence of the physical world and the cyber world [1,2]. Connecting billions of smart objects and smart devices, the Internet of Things (IoT) aims to build a smart world that enables us to perceive, analyze, control, and optimize the traditional physical systems using modern cyber technologies. Many IoT applications have been developed and deployed in recent years, and they, in turn, make our life much more convenient than before. However, they also jeopardize the traditional physical systems with cyber threats [3]. Many security breaches have been reported recently. For example, a huge number of smart cameras are compromised and utilized to form a botnet and launch a large-scale Distributed Denial-of-Service (DDoS) attack against DNS servers managed by Dye. Inc [4,5]. Security issues have become a significant concern of IoT systems and applications. They can restrict the expansion of IoT application deployment, or cause huge property losses because of the security breaches [6].

Although IoT applications expect strong security protection, securing IoT systems is challenging, which is attributed to many factors. Among them, severe resource constraints and insufficient security design are two major causes of many security problems in current IoT applications [3]. Many existing security mechanisms, including advanced security algorithms such as attribute-based access control [7], group-signature based authentication [8], homomorphic encryption [9], and public-key based solutions, demand the device to have a high level of computation power and memory space to run them. They are mostly not applicable in many IoT end devices, such as smart meters, smart lockers, smart cameras, etc. The cloud usually has almost unlimited resources, but it is located far away from IoT end devices, and it is not effective for the cloud to provide quality services to IoT devices. As an extension of the cloud, the recently emerging edge computing migrates enormous computing and storage resources to the network edge [10], which forms an edge layer that is close to IoT end devices. Therefore, many computation-heavy and resource-demanding tasks can be offloaded to the resource-rich edge layer from the resource-constrained end devices. This new computing paradigm not only alleviates the resource constraints at IoT end devices [11–14], but also optimizes the system performance [15]. It also provides a new venue to design and deploy security solutions for IoT end devices.

Some research efforts have been made in designing novel edge-based IoT security solutions. These efforts include edge-based security architecture designs [11–14], firewalls [16], intrusion detection systems [17,18], authentication and authorization protocols [19], and privacy-preserving mechanisms [20–22]. However, the research on edge-based IoT security is still at its early age. There need to be continuous investigations for more sophisticated edge-based security designs for the IoT. There is also a lack of a comprehensive review that can present a clear picture of the state-of-the-art in this research area. This paper aims to fill the gap by delivering a

* Corresponding author.
E-mail address: sha@uhcl.edu (K. Sha).

https://doi.org/10.1016/j.dcan.2019.08.006
Received 5 December 2018; Received in revised form 16 April 2019; Accepted 29 August 2019
Available online 6 September 2019

systematic survey of the existing proposed edge-based IoT security solutions. Base on that, we also aim to outline future research directions in this field to foster future novel edge-based security designs.

The contributions of the paper are three folds. Firstly, we present a general edge-centric IoT architecture, which explains how the edge layer interacts with the IoT application users, the cloud and the IoT end devices. Secondly, a comprehensive survey of edge-based IoT security designs is presented, and these designs are classified into five major categories, including security architecture, firewall, intrusion detection, authentication and authorization, and privacy. We showcase the details of representative research work for each category. To the best of our knowledge, this is the first effort to conduct such a systematic survey for edge-based IoT security designs. Finally, we identify a set of challenges and future research directions. We hope our study can both pave the way for as well as encourage and inspire many novel designs in edge-based IoT security research.

The remainder of the survey paper is organized as follows. In Section 2, we define an edge-centric IoT architecture. Following that, we conduct a systematic survey of edge-based IoT security designs in Section 3. Future research directions are outlined in Section 4. Finally, we conclude the paper in Section 5.

2. Edge-centric IoT architecture

With the rapid spreading of IoT applications, it is predicted to have 77.44 billion IoT devices by 2025 [23]. For the enormous number of IoT devices, various IoT architectures [1,24,25] are proposed from different perspectives by different organizations, and edge computing has been recognized as an important support to IoT systems [24]. Yet there does not exist an edge-centric IoT architecture. In this section, we present an edge-centric computing architecture for IoT applications, as illustrated in Fig. 1.

The edge-centric IoT architecture contains four major parts: the cloud, the IoT end devices, the edge, and the users. The design of the architecture takes both the available resources and specific features of each party into consideration. The users use intelligent IoT applications to make their lives more convenient, while more often they communicate with IoT end devices through interactive interfaces provided by the cloud or the edge rather than directly interacting with IoT end devices. The IoT end devices are deeply embedded in the physical world. They sense the physical world and take actions to control the physical world, but they are not sophisticated in computation-heavy tasks. The cloud has almost unlimited resources, but they are usually located physically far away from the end devices. Therefore, a cloud-centric IoT system usually cannot perform efficiently [26], especially when the system has the real-time requirements. With the edge being a central component of the whole architecture, it can both coordinate the other three parties to work together and complement the cloud and IoT end devices for optimized performance.

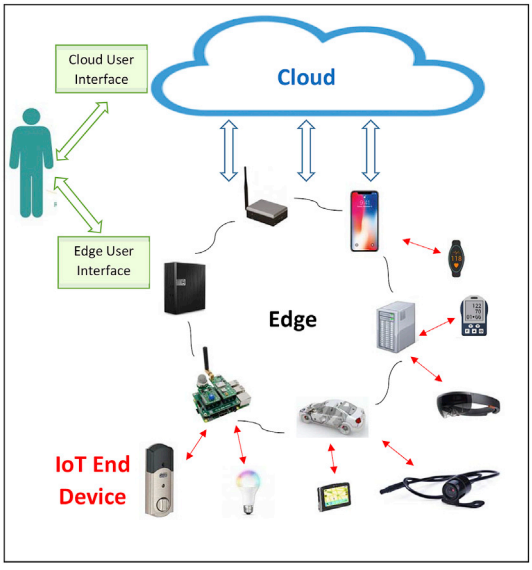

Fig. 1. Edge-centric IoT architecture.

In the edge-centric IoT architecture, IoT users submit queries to access IoT data or commands to control IoT devices. These queries and commands will eventually arrive at the edge layer through a web or mobile app-based interface provided by the cloud or the edge. Then they are handled by the edge layer, who will either forward them to IoT end devices or handle them at the edge layer on behalf of IoT end devices. Interacting with IoT end devices, the edge layer not only bridges them with the users and the cloud, but also can store data collected and uploaded from IoT end devices and offload substantial computational needs, such as big data analysis and comprehensive security algorithms from IoT end devices. In addition, many existing services for IoT end devices can be migrated from the cloud to the edge, and can be customized based on the needs of IoT end devices. In terms of the relationship between the edge and the cloud, the edge can work independently from the cloud, or the edge can work collaboratively with the cloud. In the first model, the edge is powerful enough to handle IoT application needs. For example, it can provide storage and computing services to fulfill all requests from IoT devices. In the second model, the edge gets supports from the cloud to manage the edge layer or to help handle IoT application needs. For instance, the cloud can perform deep learning based on the huge amount of collected data, and the learned model can be used by the edge to provide better services for the end devices.

The edge-centric IoT system architecture is an optimized design. Besides satisfying many real-time needs and offloading heavy computational tasks for end devices, the edge layer is an optimized venue to deploy IoT security solutions for the following reasons. First, the edge layer has more resources than IoT end devices so that many computation-intensive security operations, such as homomorphic encryption and attributes-based access control, can be deployed at the edge layer. Second, the edge layer is physically close to IoT end devices. It can satisfy real-time requirements needed in security design [27]. Third, the edge layer collects and stores data from many IoT end devices. Therefore, compared with end devices, the edge is a better place to make security decisions, for an optimal security decision depends both on the efficiency of the algorithm and the availability of sufficient information. For example, with more data, the edge layer can detect intrusion more efficiently [3–5]. With the popularity of software-defined networks and network virtualization, many security operations will be converted to routing policies; however, they may conflict with each other. Having an overview of the whole network connected through the edge, we can resolve these conflicts at the edge. Fourth, considering resource constrains, maintenance cost, and extremely large scale of end devices, it is mostly not feasible to deploy and manage firewalls at every IoT end device. Instead, deploying firewalls at the edge layer enables incoming attacks to be filtered and blocked more effectively. Fifth, considering the mobility of end devices, the edge layer can keep tracking the movement of these devices and provide a continuous secure connection for them. In addition, the relatively stable relationship between the end devices and the edge layer helps build a strong trust between them. This relieves the concerns of trust establishment among these devices. Last but not least, the edge usually has a high-speed connection with the cloud. Whenever it is necessary, the edge can contact the cloud layer for security supports. For example, the cloud can provide location and task verification for the edge, as indicated in Ref. [28], and the cloud can design powerful security mechanisms to protect the edge. Next, we investigate the edge-based solutions for IoT security.

3. Edge-based security designs for IoT

With the emergence of edge computing in recent years, many researchers have explored edge computing-based designs to tackle the IoT security challenges. These designs range from comprehensive security architecture designs to specific designs to achieve dedicated security goals, such as distributed firewalls, intrusion detection systems, authentication and authorization algorithms, and privacy-preserving mechanisms. In this section, we summarize the proposed designs and discuss their strengths and weaknesses, respectively. Because there are only a small number of edge-based IoT security solutions, we try our best to include all related and quality work in this survey.

3.1. Comprehensive security architectures at the edge layer

The edge provides a new venue to design and deploy novel and comprehensive security solutions for IoT applications. These designs target at satisfying most security needs of end devices by maximally offloading security protection from end devices to the edge layer. Placing security mechanisms at a trusted edge layer can alleviate the security challenges caused by resource constraints at the IoT device layer. Fig. 2 depicts three major categories of edge-based comprehensive security architectures, including user-centric [13,29], device-centric [12,14] and end-to-end security [30].

User-centric edge-based IoT security architecture. User satisfaction is one of the key factors in the success of IoT applications. With billions of IoT devices connected to an Internet-scale network, IoT applications provide users chances to access an enormous amount of resources in the system through various terminals, such as PCs, smartphones, smart TVs, smartwatches, etc. The convenience and pervasive resource accessibility are the most attractive features of IoT applications. However, when security is considered, there are two significant concerns. On the one hand, the user may not login infrom an always trusted and secured device. On the other, most regular users may not have enough knowledge of effectively managing security. Therefore, it is risky to rely on users for security. Having the edge layer managing the security for each specific user becomes an attractive idea, which results in security architecture designs such as offloading personal security to the network edge [13] and virtualized security at the network edge [29].

The main ideas of the user-centric security architecture designs are depicted in Fig. 3. In the figure, both designs [13,29] intend to establish a trusted domain at the edge layer. When users need to access resources in the IoT applications from various devices, they will first get connected to these Trusted Virtual Domains (TVD) that are deployed at the edge. The TVD manages the secure access to the IoT resources. The edge can have different formats. In Ref. [13], the edge layer consists of a set of secure gateways, while in Ref. [29] the edge layer consists of one or more Network Edge Devices (NED). Both designs adopt Network Function Virtualization (NFV) technology to construct the edge layer.

More specifically, in Ref. [29], each user specifies his security policy in a straightforward way. Then with the help of specification policy language [31] and policy transformation mechanisms [32], these policies are translated into a set of Personal Security Application (PSAs), such as antivirus, firewalls, and content inspection tools. A TVD works as a logical container that encapsulates user-specific PSAs, and it is deployed at a NED. Users leverage a system named SECURED [31] to configure the PSAs to a nearest compatible NED. Remote attestation and verification techniques [33] are employed to build trust between the user and the

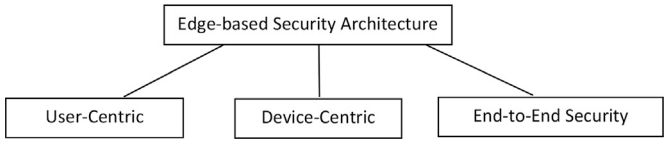

Fig. 2. Edge-based IoT security architecture.

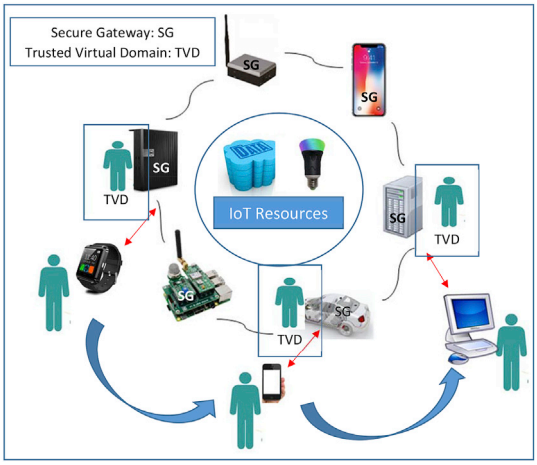

Fig. 3. User-centric edge-based IoT security architecture.

SECURED system. Finally, the NFV orchestration system helps manage and control NEDs. In this way, the majority of users' security needs are managed by the edge.

Similarly, in Ref. [13], a virtual mobile security architecture is presented. It consists of four major components, including Security Application Virtual Container (SAVC), network enforcer, resource migrator, and orchestrator. The SAVC contains a set of security tools such as firewalls, anti-phishing software, antivirus, etc., designated for each specific user based on the user's security requirements. The network enforcer instantiates a virtual private network for each user. With the coordination of an orchestrator, the resource migrator moves the state of a particular SAVC to a location that is close to the user. Thus, security issues caused by user mobility can be effectively addressed.

In sum, the user-centric edge-based IoT security designs customize security protection mechanisms at the edge layer for each particular user. Based on the virtualization technology, they can securely connect IoT users from different locations and use devices of various security protection levels.

Device-centric edge-based design for IoT security. Billions of IoT devices are deeply embedded in the physical world. They not only sense valuable data which enables many intelligent applications to be built upon them, but also actuate many vital decisions to control the physical world. Different from user-centric IoT security designs, the device-centric IoT security design customizes security solutions for each individual end device based on its available resources, the sensitiveness of the sensing data, and the impact of actuating tasks. It can also consider the security needs of a set of end devices together. EdgeSec [12] and ReSIoT [14] are two representative designs that leverage the edge layer to deploy device-specific IoT security solutions. The main ideas of these designs are to offload security functions from IoT devices to the edge layer, as

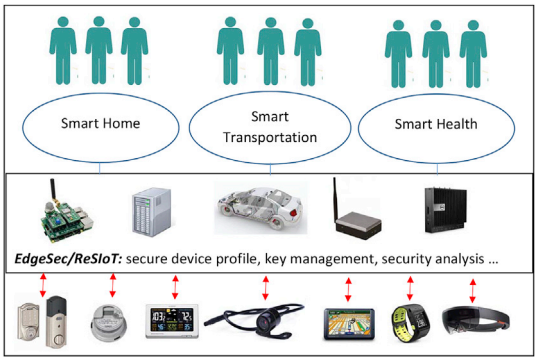

Fig. 4. Device-centric edge-based IoT security architecture.

indicated in Fig. 4. Most designs do not aim to change the existing network architecture and standard protocols. Instead, they complement end devices to satisfy the security requirements of IoT applications.

EdgeSec [12] designs a new security service that is deployed at the edge layer to enhance the security of IoT systems. EdgeSec consists of six major modules that work together to systematically handle specific security challenges in IoT systems. These modules include security profile manager, security analysis, protocol mapping, a security simulation, communication interface management, and request handling. First, each IoT device is registered to the security profile managing module so that the device-specific information is collected and the device-specific security requirements are identified. Then, a security analysis module oversees the security of an independent IoT subsystem by implementing two functions. One analyzes the security dependency of the registered devices in the IoT subsystem, and the other makes a decision on where to deploy the security functions. The protocol mapping the module chooses appropriate security protocols from the protocol library for each particular IoT device based on its available resources and established security profile. Moreover, the security simulation module simulates the consequences of critical instructions before they are actually executed in order to protect the safety of the physical system. Other components provide functions such as masking the heterogeneity in communication and coordinating different modules to work together.

ReSIoT [14] presents a reconfigurable security framework for IoT applications. The framework designs a Security Agent (SA), which can be a wireless router, a base station, or a gateway device, to offload the overhead of cryptographic computations at IoT devices. Therefore, the resource-constrained IoT devices will be protected by advanced security algorithms which require high computation power. In the ReSIoT architecture, the whole IoT system is organized into four major components, including a set of IoT application servers, IoT security domains, a global key management system, and a global Authentication, Authorization, and Accounting (AAA) system at the edge layer. The SAs work together to implement a set of Reconfigurable Security Functions (RSFs) protocols that realize the functions defined in the above four ReSIoT components. In this way, many computation-intensive and advanced cryptographic algorithms, such as group signature and attribute-based encryption, can be utilized to construct IoT security solutions.

In sum, device-centric edge-based IoT security designs consider the characteristics of each end device and satisfy its security needs by customizing an appropriate security solution for it.

End-to-end security for IoT. Many IoT applications expect end-to-end security among IoT devices and between IoT devices and the cloud. However, realizing end-to-end security in the IoT is challenging, mainly because of the heterogeneity of those devices. As the edge layer works as a bridge to connect heterogeneous IoT devices and the cloud, researchers have considered designing a secure middleware deployed at the edge layer for the secure end-to-end communications among IoT devices. In Ref. [30], the middleware manages security functions, such as MAC algorithms, encryption algorithms, authenticators, as well as secure session status for mobile devices.

3.2. Firewalls at the edge layer

Most IoT devices are resource-constrained, so they cannot support heavy security applications such as firewalls. In addition, considering the large scale of IoT devices, it will be extremely costly to manage a huge number of firewalls if each IoT device has a firewall. Edge-based firewalls are most cost-effective and efficient. Fig. 5 demonstrates an example of an edge-based firewall design. In the figure, IoT applications define firewall policies that are converted into a set of flow policies. After conflicts in the flow policies are detected and resolved, these policies become a set of distributed firewall rules which are deployed at the edge. Later on, all incoming and outgoing traffics are subject to the examination of these rules.

Deploying the firewall at the edge layer is an optimal choice for the following advantages. First, updating of firewalls will be more feasible and manageable as there is only one conceptually centralized firewall. Second, in many IoT applications, an edge device may manage an IoT subsystem. Thus, the firewall can be configured to accommodate the overall security needs of the subsystem. Third, it can support user mobility in the IoT system as the edge layer may track the movement and the credential of the user and end device. Next, we review two edge computing-based firewall designs, including FLOWGUARD [16] and a distributed firewall architecture at the network edge [34]. The first utilizes the Software-Defined Network (SDN) technology while the second makes use of the Virtual Network Function (VNF) technology.

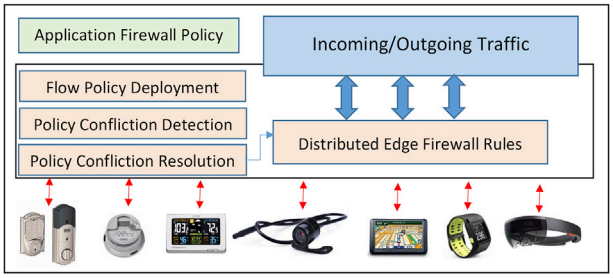

Fig. 5. Distributed edge-based firewalls.

FLOWGUARD [16] contains three major function units: network state and configuration update, violation detection and violation resolution. In FLOWGUARD, the violation detection not only checks the violation of each flow as in traditional techniques, but also tracks the flow path to identify the original source and the final destination of each flow in the network. The idea is to use Header Space Analysis (HSA) [35] as the flow tracking mechanism. They also introduce the concept of Firewall Authorization Space(FAS) to represent allowed or denied packets by the firewall rules, enabling the conversion of firewall rules into disjoint authorization sub-spaces, i.e., denied authorization space and allowed authorization space. Based on the flow path and firewall authorization space, violations are detected. In the process of violation resolution, when installing a new flow policy, a novel comprehensive violation resolution mechanism is designed. The new mechanism, instead of directly rejecting a new flow that may partially violate the flow policy, proposes flow rerouting and flow tagging to break flow dependency [36].

Markham and Payne have presented a distributed firewall architecture at the network edge [34]. The architecture adopts a master/slave architecture to provide centralized management at the edge layer of distributed policy enforcement points for many devices. A policy server provides functions, such as user interface, policy management, network connection group management, and audition. It also creates policies and pushes them to the Network Interface Cards (NICs), which filter packets that violate policies. The designed distributed firewall architecture is expected to be scalable, topology-independent, non-bypassable, and tamper-resistant.

3.3. Intrusion Detection Systems (IDS) at the edge layer

In 2016, attackers compromised a great number of IoT devices and used them to launch a DDoS attack on many DNS servers owned by Dye Inc [4,5]. The attack caused significant losses in that it interrupted Internet connection in a large area. If there was a distributed intrusion detection system, it might have been able to detect the DDoS attack at its early stage and limit the loss caused by the attack. With more information available at the edge layer, there are many advantages to designing intrusion detection mechanisms at the edge layer. For example, it can utilize advanced machine learning algorithms to relate data from multiple sources for better intrusion detection results. It can also be adaptive to the changes in the attack patterns. Below we discuss several alternative designs that aim to detect intrusions in IoT systems at the edge layer [17,18,37].

A conceptual edge-based intrusion detection system design is depicted in Fig. 6. In this design, the distributed traffic monitoring service collects real-time network traffics. It then runs intrusion detection algorithms at the individual edge device. Moreover, collaborative intrusion detection is performed by examining traffic data from multiple edge devices. Finally, the detection results are enforced by the network controllers deployed at the edge devices.

Roman et al. proposed a Virtual Immune System (VIS) to analyze the security and consistency of the underlying IoT infrastructure [17]. As depicted in Fig. 7, the VIS has two functional parts, the VIS kernel and the Virtual Immune Cells (VIC), and it contains a communication module, a reporting module, and a security operations agreement module. Within the VIS Kernel, there is a VIS orchestrator that configures and deploys VICs in the edge infrastructure based on the information collected from various sources, including the internal system administrators, external threat intelligence feeds, and information collected by VICs in the edge infrastructure. The VICs scans communication ports, analyzes traffic, and handles platform-specific tasks. They also manage credentials, store logs, and hold Security Operations Level Agreements (SOLA).

SIOTOME [18] illustrates an Edge-ISP collaborative architecture to detect and isolate IoT security attacks. It integrates the large-scale view from the ISP and the fine-grained view of each IoT device to build efficient and privacy-aware IoT security services. In SIOTOME, the edge data collector monitors the behavior of IoT devices based on the observation of network traffic. Then the edge analyzer analyzes the collected data to identify threats and attacks, as well as to notify the edge controller when threats and attacks are detected. The edge controller then configures the network gateway to modify the network traffic. SIOTOME also leverages defense mechanisms like network isolation [38] to limit the attack surface and the allowed network input and output, as well as to stop vulnerability scanning and DDoS attacks.

Similarly, an edge computing-based anomaly detection scheme is proposed in Ref. [37]. The researchers devised an edge computing-based system to detect a specific yet important attack, the selective forwarding in mobile IoT systems. In Ref. [37], IoT devices work as watchdogs that measure the dropping rates of their neighboring devices. Each edge server collects, aggregates, and shares the information from the watchdogs with other edge servers. A voting method is applied to detect the malicious activity of selective forwarding behaviours at end devices.

3.4. Edge-based authentication and authorization mechanisms

Recently, Trend Micro showed that unauthorized access is the top type of attacks towards a control system [39]. Authentication and authorization are crucial security mechanisms to stop many types of attacks, including unauthorized access and DDoS attacks [40]. In the IoT system architecture, end-to-end security is also expected to be based on authentication and authorization mechanisms, but it is extremely hard to realize due to many reasons. Using mutual authentication as an example, first, it is very difficult to establish an end-to-end direct communication between two heteromerous peers. Second, many traditional authentication mechanisms like those based on digital signature are not applicable in IoT end devices.

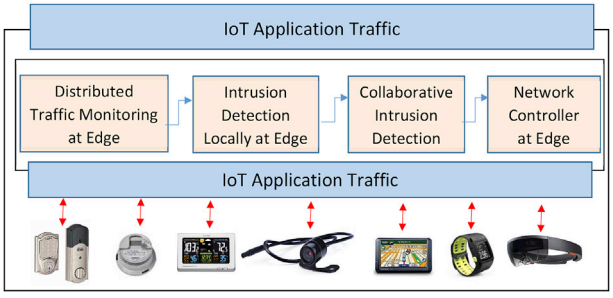

Fig. 6. Distributed edge-based intrusion detection systems.

Fig. 7. Virtual immune system.

With the support of the edge layer, many researchers have designed edge-based authentication protocols that utilize the multiple-phase authentication, as shown in Fig. 8. As shown in the figure, the authentication process is divided into multiple segments, including the authentication between the end device and the edge layer, as well as the authentication between the edge and another party, which can be an IoT user, the cloud, or other end devices. Based on the characteristics of communication peers, customized authentication protocols can be adopted for different segments. In this way, the edge works as a benign man-in-the-middle that helps set up mutual authentication for heterogenous devices. Besides, there is another way in which the edge represents end devices to complete the authentication and authorization process, i.e., the end devices outsource the authentication and authorization functions to the edge. Furthermore, because the edge has resources to support multiple authentication interfaces, the multi-factor authentication [41] becomes possible in IoT systems.

As an example design in which the edge works as a proxy and represents the IoT end devices for authentication and authorization purposes, in Ref. [19], control system gateways are developed to implement multi-factor authentication and authorization, as well as to deploy a real-time identity monitoring service that assures the identity validity. The multi-factor authentication usually involves more than two types of different authentication mechanisms, which aim to verify what you know (such as username/password and security questions), and/or what you have (such as a token, a key, a certificate, and/or a smart fob), and/or what you are (like biometrics). In the authentication and authorization process, the IoT user sends a request together with a biometrics and identity information to the gateway. The gateway authenticates the user and determines its authorization level.

The research work by Sha et al. [28,42] shows an example of employing an edge device as a bridge to mutually authenticate an IoT user and the IoT device. The solution is a two-phase authentication protocol. In the first phase, the edge device authenticates the user using a digital signature based protocol and gets a credential from the user. Based on the received credential, the edge device further reaches a mutual authentication with the IoT end device using a symmetric key--based authentication protocol. Similarly, several other edge-supported authentication protocols are proposed to authenticate RFIDs, including [43,44].

At the resource-rich edge layer, many powerful authentication and

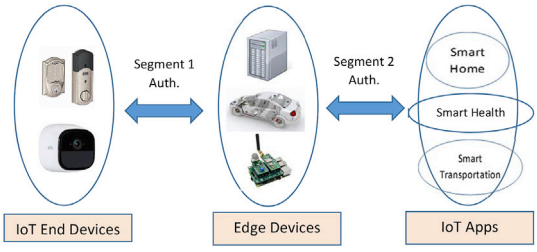

Fig. 8. Multi-segment authentication based on edge-computing.

authorization algorithms can be supported. For example, in Ref. [7], attribute-based access control at the edge layer enforces a fine-grained access policy to access IoT data.

3.5. Edge-based privacy-preserving designs

IoT applications collect enormous valuable and sensitive data from pervasive IoT devices. Because many IoT applications, such as smart home and smart city, are deeply embedded into everyone's daily life, IoT users expect stringent privacy protection. With more data available at the edge layer than at the end devices, it is possible to achieve various privacy goals such as differential privacy [45], k-anonymity [46–48], and privacy-preserving aggregation [20]. In other words, when IoT applications submit queries for IoT data, the edge can first process the data and then respond to these queries by supplying the IoT applications with privacy-preserved data, as illustrated in Fig. 9.

A Lightweight Privacy-preserving Data Aggregation (LPDA) scheme is proposed at the edge layer [20]. In this scheme, the IoT devices report their locally processed sensing data together with a Message Authentication Code (MAC) to the edge nodes. After the edge nodes receive the reports, they first authenticate the IoT end devices by comparing the MAC values and then produce an aggregated value for the IoT applications. By using homomorphic Paillier encryption [49], the Chinese Remainder Theorem(CRT) [50], and one-way hash chain techniques, the proposed scheme can address issues of aggregating hybrid IoT data, reduce the volume of communication, and filter false data from the IoT end device reports. Moreover, LPDA utilizes differential privacy techniques [51] to realize the goal of privacy preservation.

Similarly, Du et al. also proposed an Output Perturbation (OPP) method by adding Laplacian random noise to the output value [21]. The mechanism achieves differential privacy while not significantly impacting data utility. They also introduce ObJective Perturbation (OJP). Unlike OPP, OJP adds noise to the blocked data rather than the output value at the edge. Then the output value is calculated based on the modified data at the edge layer. Compared with OPP, OJP achieves better privacy-preserving results.

A privacy-aware scheduling algorithm was proposed based on edge computing [22]. The main idea of the work is to execute tasks from different applications that have different privacy requirements at different servers. For example, private application tasks execute only at a local or private cloudlet/micro data center, semi-private application tasks can be executed at a local or private cloudlet/micro data center that communicates with a cloud data center, while public application tasks can be scheduled to any data center. The proposed scheduling algorithm also aims to satisfy the real-time requirements of applications.

4. Open research issues

In previous sections, we have reviewed a set of existing research efforts which focus on designing edge-based IoT security solutions. We have observed that this research is still in its infancy. There are still many challenging issues to be addressed. In this section, we outline a set of open research issues, including securing the edge layer, dealing with untrusted edge layer, data quality for security, distributed and cross-domain machine learning algorithms for IoT security, safety simulation and response mechanisms, lightweight protocols for end device-edge communications, as well as secure operating systems and lightweight virtual machines.

While the edge layer provides a new venue for deploying novel IoT security solutions, it also increases the attack surfaces because the edge layer itself needs security protection. Securing the edge layer is not a trivial task because, compared with data centers in the cloud, most edge nodes may not be administrated by a strong team of security professionals and may not be located in a physically secured location. This calls for additional research efforts to design security solutions for edge devices. In addition, the edge-centric IoT architecture introduces many

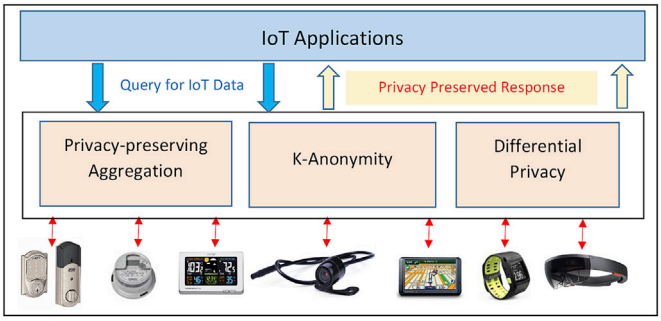

Fig. 9. Edge-based privacy-preserving design.

extra communications between the edge and the cloud as well as between the edge and IoT end systems. It is also necessary to secure these communications.

Although there are several edge-based privacy-preserving techniques for IoT applications, in many current designs, IoT end devices choose to trust the edge. However, the edge nodes may be compromised, or they can be curious [52], i.e., they may try to track the activities of IoT devices for their own interests as what the cloud has done [53]. In this case, IoT end devices need novel solutions to protect their privacy. Isolation technologies have been explored, but how to support isolation for resource-constrained IoT devices or how to effectively implement isolation at the edge layer are interesting future research topics. On the other hand, we need algorithms that can establish a strong trust between IoT end devices and the edge. In addition, an efficient third-party audition [54] can also be considered to protect privacy.

Sommer and Paxson proposed using machine learning for network intrusion detection in 2010 [55]. Later on, many efforts have been made to advance this research area. Buczak and Guven presented a comprehensive survey of data mining and machine learning methods for intrusion detection [56]. Recently, with the popularity of deep learning, these methods are also used in intrusion detection. For example, Recurrent Neural Networks (RNN) is used to detect intrusion in Ref. [57] and Shone et al. proposed to detect intrusion by constructing a deep learning classification model based on nonsymmetric deep autoencoder [58]. However, most existing deep learning and machine learning methods are centralized algorithms and need a large amount of data. They are suitable to be deployed at the cloud. Having seen the benefits of deploying intrusion detection and firewall at the edge, we observe that most of these developments are still in their early stage. How to efficiently customize these algorithms at the edge and effectively detect intrusion based on a small set of data with limited information remains a research challenge. Cross-domain reasoning is important for future intelligent intrusion detection algorithms. This also requires effective collaborations among many edge nodes, which may be deployed and managed by different administration domains. How to encourage edge nodes to participate, collaborate, and achieve the same security goal at an affordable low cost could be investigated. Machine learning technologies should be customized for the edge-centric IoT architecture for better accuracy and performance based on a small set of data with limited information. Conflict resolution mechanisms are also needed to consolidate multiple policies from different administration domains.

We usually make security decisions based on data collected from devices in the IoT system and the system environment. For example, machine learning algorithms learn attack models from the data for attack detection. The quality and trustworthiness of these data are crucial to the correctness of the decisions. Thus it is not trivial to design reliable protocols to collect a set of high-quality data [59]. In this direction, cross-verification algorithms and deceptive data detection and filtering technologies [60] are of interest.

Although the safety of the physical system is extremely important, there is not sufficient research in this field. Security and safety

simulations could be of great importance, as indicated in Ref. [12]. However, how to model and conduct the safety simulation that can produce a reliable evaluation of the safety risk is a significant challenge. Moreover, many safety-related decisions have real-time requirements. This further complicates the simulation modeling and design. Consequently, isolation techniques and first response mechanisms [61] to respond to potential safety risks need a lot more studies to limit the physical system loss. Both edge-based and end device-based solutions are expected.

Even though we can offload many security operations from IoT end devices to the edge layer, in many IoT applications, there are still the needs for a good level of security protection for the communication channels connecting end devices and the edge. However, it is difficult because of severe resource limitations at the end devices. Another area worth exploring is the lightweight secure communication protocols between the end devices and the edge layer. Physical-features-based approaches, such as [20,62], are of great interest.

Finally, virtual machines are widely used to construct the edge layer, and sometimes there may be multiple virtual machines at one edge node as the edge node serves multiple users or end devices. The security of these virtual machines and hypervisors [63] are critical for the security of the whole edge layer. Considering the scale of IoT applications, it is also required for these virtual machines to be lightweight. In consideration of those needs, we need more research for secure and lightweight virtual machines, secure operating systems like SeL4 [64], as well as their applications in edge computing.

5. Conclusion

In recent years, the challenge of securing IoT systems has sparked tremendous research interests. However, it remains a significant challenge. Emerging edge computing has resulted in many novel edge-based security designs for IoT security. This paper presents a systematic and in-depth review of existing edge-based IoT security solutions in the context of a formally defined edge-centric IoT architecture. These solutions cover the most important topics in IoT security, including comprehensive security architecture, firewalls, intrusion detection systems, authentication and authorization mechanisms, as well as privacy-preserving designs. Furthermore, we have identified a set of challenges in the field and outlined a list of research directions.

Declarations of interest

None.

Acknowledgements

This research has been supported by the National Science Foundation (under grant #1723596) and by the National Security Agency (under grant #H98230-17-1-0355).

References

[1] J. Gubbi, et al., Internet of things (iot): a vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (7) (2013) 1645–1660.
[2] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, W. Zhao, A survey on internet of things: architecture, enabling technologies, security and privacy, and applications, IIEEE Internet Things (IoT) J. (2017) 99, 1–1.
[3] K. Sha, et al., On security challenges and open issues in internet of things, Future Gener. Comput. Syst. 83 (2018) 326–337.
[4] T. Brewster, How hacked cameras are helping launch the biggest attacks the internet has ever seen. https://www.forbes.com/sites/thomasbrewster/2016/09/25/brian-krebs-overwatch-ovh-smashed-by-largest-ddos-attacks-ever/#705007235899, September 2016.
[5] M.-A. Russon, Hackers turning millions of smart cctv cameras into botnets for ddos attacks. http://www.ibtimes.co.uk/hackers-turning-millions-smart-cctv-cameras-into-botnets-ddos-attacks-1525736. (Accessed September 2016).
[6] K. Sha, W. Wei, A. Yang, W. Shi, Security in internet of things: opportunities and challenges, in: Proceedings of International Conference on Identification, Information & Knowledge in the Internet of Things (IIKI 2016), 2016.
[7] M. Alramadhan, K. Sha, An overview of access control mechanisms for internet of things, in: Proceedings of the 26th International Conference on Computer Communications and Networks (ICCCN 2017), 2017.
[8] S. Chen, P. Zeng, K.R. Choo, X. Dong, Efficient ring signature and group signature schemes based on q-ary identification protocols, Comput. J. 61 (4) (2018) 545–560.
[9] Z. Wang, K. Sha, W. Lv, Slight homomorphic signature for access controlling in cloud computing, Wirel. Pers. Commun. 73 (1) (2013) 51–61.
[10] W. Shi, J. Cao, Q. Zhang, Y. Li, L. Xu, Edge computing: vision and challenges, IEEE Internet. Things J. 3 (5) (2016) 637–646.
[11] P. Mach, Z. Becvar, Mobile Edge Computing: A Survey on Architecture and Computation Offloading, arXiv preprint arXiv:1702.05309.
[12] R. Errabelly, K. Sha, W. Wei, T.A. Yang, Z. Wang, Edgesec: design of an edge layer security service to enhance internet of things security, in: Proceedings of the First IEEE International Conference on Fog and Edge Computing (ICFEC 2017), 2017.
[13] D. Montero, R. Serral-Gracia, Offloading personal security applications to the network edge: a mobile user case scenario, in: Proceedings of IEEE 2016 International Conference on Wireless Communications and Mobile Computing, 2016.
[14] R. Hsu, J. Lee, T. Quek, J. Chen, Reconfigurable security: edge-computing-based framework for iot, IEEE Network 32 (5) (2018) 92–99.
[15] X. Tao, K. Ota, M. Dong, H. Qi, K. Li, Performance guaranteed computation offloading for mobile-edge cloud computing, IEEE Wireless Commun. Lett. 6 (6) (2017) 774–777.
[16] H. Hu, W. Han, G. Ahn, Z. Zhao, Flowguard: building robust firewalls for software-defined networks, in: Proceedings of the Third Workshop on Hot Topics in Software Defined Networking, 2014.
[17] R. Roman, R. Rios, J. Onieva, J. Lopez, Immune system for the internet of things using edge technologies, IEEE Internet. Things J. (2018) 1–8.
[18] H. Haddadi, V. Christophides, R. Teixeira, K. Cho, S. Suzuki, A. Perrig, Siotome: an edge-isp collaborative architecture for iot security, in: Proceedings of 1st International Workshop on Security and Privacy for the Internet-Of-Things (IoTSec), 2018.
[19] Z. Ali, M.S. Hossain, G. Muhammad, I. Ullah, H. Abachi, A. Alamri, Edge-centric multimodal authentication system using encrypted biometric templates, Future Gener. Comput. Syst. 85 (2018) 76–87.
[20] R. Lu, K. Heung, A. Lashkari, A.A. Ghorbani, A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced iot, IEEE Access 5 (2017) 3302–3312.
[21] M. Du, et al., Big data privacy preserving in multi-access edge computing for heterogeneous internet of things, IEEE Commun. Mag. 56 (8) (2018) 62–67.
[22] A. Singh, et al., Rt-sane: real time security aware scheduling on the network edge, in: Proceedings of the 10th International Conference on Utility and Cloud Computing, 2017.
[23] T.S. Portal, Internet of things (iot) connected devices installed base worldwide from 2015 to 2025. https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/, 2018.
[24] J. Lin, W. Yu, N. Zhang, X. Yang, L. Ge, On data integrity attacks against route guidance in transportation-based cyber-physical systems, in: Proceedings of the 14th IEEE Annual Confernece in Consumer Communications and Networking Conference (CCNC 2017), 2017.
[25] D. Singh, G. Tripathi, A.J. Jara, A survey of internet-of-things: future vision, architecture, challenges and services, in: Proceedings of 2014 IEEE World Forum on Internet of Things (WF-IoT), 2014.
[26] X. Chen, L. Jiao, W. Li, X. Fu, Efficient multi-user computation offloading for mobile-edge cloud computing, IEEE/ACM Trans. Netw. (5) (2016) 2795–2808.
[27] I. Lee, K. Lee, The internet of things (iot): applications, investments, and challenges for enterprises, Bus. Horiz. 58 (4) (2015) 431–440.
[28] K. Sha, N. Alatrash, Z. Wang, A secure and efficient framework to read isolated smart grid devices, IEEE Trans. on Smart Grid 8 (6) (2017) 2519–2531.
[29] D. Montero, et al., Virtualized security at the network edge: a user-centric approach, IEEE Commun. Mag. 53 (4) (2015) 176–186.
[30] B. Mukherjee, R. Neupane, P., Calyam End-to-end iot security middleware for cloud-fog communication, in: Proceedings of the IEEE 4th International Conference on Cyber Security and Cloud Computing, 2017.
[31] T.S. project, Security at the network edge. https://www.secured-fp7.eu/,online. (Accessed 1 November 2018).
[32] C. Basile, A. Lioy, S. Scozzi, M. Vallini, Ontology-based security policy translation, J. Inf. Assur. Secur. 5 (1) (2010) 437–445.
[33] K. Goldman, R. Perez, R. Sailer, Linking remote attestation to secure tunnel endpoints, in: Proceedings of the First ACM Workshop on Scalable Trusted Computing, 2006.
[34] T. Markham, C. Payne, Security at the network edge: a distributed firewall architecture, in: Proceedings of DARPA Information Survivability Conference & Exposition II, 2001, 2001.
[35] P. Kazemian, G. Varghese, N. McKeown, Header space analysis: static checking for networks, in: Proceedings of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI'12), 2012.
[36] M. Reitblatt, N. Foster, J. Rexford, C. Schlesinger, D. Walker, Abstractions for network update, Comput. Commun. Rev. 42 (4) (2012) 323–334.
[37] Q. Yaseen, F. AlBalas, Y. Jararweh, M. Al-Ayyoub, A fog computing based system for selective forwarding detection in mobile wireless sensor networks, in: Proceedings of IEEE International Workshops on Foundations and Applications of Self-* Systems, 2016.
[38] R. Nunes, R. Pontes, D. Guedes, Virtualized network isolation using software defined networks, in: Proceedings of 2013 IEEE 38th Conference on Local Computer Networks (LCN 2013), 2013.

[39] K. Wilhoit, Who's really attacking your ics equipment?. http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-whosreally-attacking-your-ics-equipment.pdf, June 2017.
[40] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, Ddos in the iot: mirai and other botnets, Computer 50 (7) (2017) 80–84.
[41] D. Dasgupta, A. Roy, A. Nag, Multi-factor authentication, in: Advances in User Authentication, Springer, 2017, pp. 185–233.
[42] K. Sha, C. Xu, Z. Wang, One-time symmetric key based cloud supported secure smart meter reading, in: Proceedings of the 23rd International Conference on Computer Communications and Networks (ICCCN 2014), 2014.
[43] K. Fan, et al., Eslras: a lightweight rfid authentication scheme with high efficiency and strong security for internet of things, in: Proceedings of 2012 4th International Conference on Intelligent Networking and Collaborative Systems (INCoS), 2012.
[44] P. Gope, R. Amin, S.H. Islam, N. Kumar, V.K. Bhalla, Lightweight and privacy-preserving rfid authentication scheme for distributed iot infrastructure with secure localization services for smart city environment, Future Gener. Comput. Syst. 83 (2018) 629–637.
[45] C. Dwork, et al., The algorithmic foundations of differential privacy, Found. Trends® Theor. Comput. Sci. 9 (3–4) (2014) 211–407.
[46] L. Sweeney, k anonymity, A model for protecting privacy, Int. J. Uncertain. Fuzziness Knowledge-Based Syst. 10 (5) (2002) 557–570.
[47] Y. Xi, K. Sha, W. Shi, L. Schwiebert, T. Zhang, Enforcing privacy using symmetric key-set in vehicular networks, in: Proceedings of the 8th International Symposium on Autonoumous Decentralized Systesms, 2007.
[48] K. Sha, Y. Xi, W. Shi, L. Schwiebert, T. Zhang, Adaptive privacy-preserving authentication in vehicular networks, in: Proceedings of the International Workshop on Vehicle Communication and Appliations, 2006.
[49] G. Gentry, D. Boneh, A Fully Homomorphic Encryption Scheme, Stanford University Stanford, 2009.
[50] D. Pei, A. Salomaa, C. Ding, Chinese Remainder Theorem: Applications in Computing, Coding, Cryptography, World Scientific, 1996.
[51] F. McSherry, K. Talwar, Mechanism design via differential privacy, in: Proceedings of 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07), 2007.
[52] B. Razeghi, S. Voloshynovskiy, Privacy-preserving outsourced media search using secure sparse ternary codes, in: Proceedings of 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018.
[53] S. Sharma, K. Chen, Privategraph: a cloud-centric system for spectral analysis of large encrypted graphs, in: Proceedings of 2017 IEEE 37th International Conference on Distributed Computing Systems, 2017.
[54] K. Loheswaran, J. Premalatha, Renaissance system model improving security and third party auditing in cloud computing, Wirel. Pers. Commun. 90 (2) (2016) 1051–1066.
[55] R. Sommer, V. Paxson, Outside the closed world: on using machine learning for network intrusion detection, in: Proceedings of 2010 IEEE Symposium on Security and Privacy, 2010, pp. 305–316.
[56] A.L. Buczak, E. Guven, A survey of data mining and machine learning methods for cyber security intrusion detection, IEEE Commun. Surv. Tutor. 18 (2) (2016) 1153–1176.
[57] C. Yin, Y. Zhu, J. Fei, X. He, A deep learning approach for intrusion detection using recurrent neural networks, Ieee Access 5 (2017) 21954–21961.
[58] N. Shone, T.N. Ngoc, V.D. Phai, Q. Shi, A deep learning approach to network intrusion detection, IEEE Trans. Emerg. Top. Comput. Intell. 2 (1) (2018) 41–50.
[59] K. Sha, S. Zeadally, Data quality challenges in cyber-physical systems, J. Data Inf. Qual. (JDIQ) 6 (2–3) (2015) 8.
[60] K. Sha, S. Wang, W. Shi, Rd4: role-differentiated cooperative deceptive data detection and filtering in vanets, IEEE Trans. Veh. Technol. 59 (3) (2010) 1183–1190.
[61] R.H. Weber, E. Studer, Cybersecurity in the internet of things: legal aspects, Comput. Law Secur. Rep. 32 (5) (2016) 715–728.
[62] K. Sha, M. Kumari, Patient identification based on wrist activity data, in: Proceedings of 3rd IEEE/ACM Conference on Connected Health: Apprications, Systems and Engineering Technologies, 2017.
[63] H. Tsai, et al., Threat as a service?: virtualization's impact on cloud security, IT professional 14 (1) (2012) 32–37.
[64] K. Eldefrawy, N. Rattanavipanon, G. Tsudik, Fusing hybrid remote attestation with a formally verified microkernel: lessons learned, in: Proceedings of 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop, 2017.