A Survey on Access Control in Fog Computing

Peng Zhang, Joseph K. Liu, F. Richard Yu, Mehdi Sookhak, Man Ho Au, and Xiapu Luo

The authors provide a comprehensive survey of access control of users' data in the environment of fog computing with the aim of highlighting security problems and challenges. They discuss the definition, architecture, and characteristics of fog computing, based on which typical requirements and essential models of access control are addressed.

Abstract

Fog computing has emerged as an attractive solution to the distributed application of the Internet of Things, and could provide low-latency, highly mobile, and geo-distributed services distinguished from the cloud. While fog computing offers numerous benefits, it also faces various security challenges, of which access control is one of the fundamental requirements since it is concerned with authorizing users to access resources in the open fog environment. This article provides a comprehensive survey of access control of users' data in the environment of fog computing with the aim of highlighting security problems and challenges. It discusses the definition, architecture, and characteristics of fog computing, based on which typical requirements and essential models of access control are addressed. Finally, it highlights known access control schemes in the environment of fog computing, and identifies existing unresolved problems as future directions.

Introduction

The Internet of Things (IoT) links smart devices, such as wearable devices, connected vehicles, and wireless sensors, to the Internet, and makes everything connected and smarter. As predicted by Cisco, more than 50 billion smart devices will be connected to the Internet, and the average connected smart devices each person has will reach 6.58 in 2020. Obtaining deeper insight with analytics on applying IoT could raise productivity and produce new business models and revenue streams. As is known to all, smart devices usually face challenges including battery capacity, storage space, computing resource, and bandwidth limitation, which in turn impede user experience and quality of service (QoS). Considering these resource limitations on smart devices, cloud computing is viewed as one of the most useful computing paradigms, which provides services to users on the part of software, platform, and infrastructure, and offers elastic resources for applications at low cost [1].

In recent years, the expansion of cloud computing has been booming in the areas of information technology and computer networking, and its application has a well developing future. However, increasing challenges emerge in cloud computing to satisfy the IoT's application demands due to some intrinsic problems (e.g., lack of mobility support, unreliable latency, and local awareness). The latency of many IoT applications (e.g., vehicle-to-roadside communications, virtual reality) generally is below tens of milliseconds. On the other hand, the data is created by a large and growing amount of connected things at an exponential rate; thus, extremely high communication bandwidth would be needed to upload all the data. However, these demands extend far beyond the level of the attainment of cloud services. Thus, it is urgent to find a new platform to meet these demands.

Fog computing was introduced by Cisco [2], and defined as a paradigm that extends computing at the edge of the network, which could offer new services and applications, particularly for future networks such as IoT. Fog computing offers storage, computation, and network services for users by facilities or infrastructures called fog devices. The remarkable characteristics of fog computing focus on mobility support, proximity to users, and widespread geographical distribution. Figure 1 is a three-layer user-fog-cloud hierarchy. User devices are attached to fog devices, which could be interlinked with each other, and each fog device is connected to the cloud. Hence, the data from users could be preprocessed by fog devices, which could relieve the overloaded cloud data centers, reduce service latency and improve QoS to deliver outstanding user experience.

It is inevitable that fog computing will collect and process deeply personal information. As a result, without proper security and privacy-preserving mechanisms, fog computing cannot be adopted despite its usefulness. Clearly, fog computing will suffer from the classical security and privacy problems inherited from cloud computing. In addition, it suffers from a unique threat due to the adoption of fog devices, such as man-in-the-middle attack.

Therefore, access control is especially important to ensure security, which is a strategy that allows or restricts users' access to the system. Access control guarantees normal access for valid users, prevents attacks of unauthorized users, and solves security problems caused by fault operations of valid users.

In traditional access control, users store their data in trusted servers. Then the trusted servers check whether the requested user has the privilege of access data. In fog/cloud computing, as users and servers are in different trust domains, and servers are untrusted, this kind of model is disabled. At the same time, due to the salient features of fog computing, existing solutions for cloud computing cannot be applied directly.

In this article, we focus on:

- Discussing the definition, architecture, and characteristics of fog computing
- Providing requirements and taxonomies of access control

Digital Object Identifier:
10.1109/MCOM.2018.1700333

Peng Zhang is with Shenzhen University; Joseph K. Liu is Monash University; F. Richard Yu and Mehdi Sookhak are with Carleton University; Man Ho Au and Xiapu Luo are with Hong Kong Polytechnic University.

0163-6804/18/$25.00 © 2018 IEEE

- Highlighting the current state of access control in fog computing
- Identifying potential problems

The layout of this article is as follows. The definition, architecture, and characteristics of fog computing are introduced. We discuss requirements and the classification of access control. The existing work on access control in fog computing is surveyed. Open research problems are analyzed. This article is then concluded.

Fog Computing Overview

Here, we provide an overview on fog computing and the existing definitions of this new concept. We also describe the architecture and characteristics of fog computing.

The Definition of Fog Computing

Fog computing was introduced by Cisco in 2012 [2], and *it is a highly virtualized platform, which provides storage, computation, and network services between smart devices and cloud servers, typically but not exclusively deployed at the edge of the network*. Rather than cannibalizing cloud computing, fog computing is viewed as one of cloud computing's extended versions, which helps to create a hierarchical infrastructure, where the local and global data analysis are executed at the fog and cloud, respectively.

Later, the definition of fog computing was revised by [3], and *it is a scenario where massive heterogeneous and distributed fog devices communicate and cooperate with each other to carry out data storage and computation tasks without the help of any third party. These tasks support basic network functions and new services and applications that run in sandboxed environments.*

Vaquero *et al.* [3] introduced a comprehensive view of fog computing but did not put forward the unique connection with the cloud. Yi *et al.* [4] studied all the similar concepts of fog computing and gave a more general definition as: *it is a distributed computing platform with a resource pool, which consists of one or more ubiquitous and heterogeneous fog devices at the edge of the network and not exclusively backed by cloud servers, to provide closely flexible storage, computation, and network services to large-scale users.*

These extended definitions are compared in Table 1. Although these definitions are debatable, they all reveal advantages that fog computing may have. Now it is not just an extended version of cloud computing, but it has its own paradigm in which centralized cloud computing coexists with distributed fog computing but is not essential.

The Architecture of Fog Devices

Fog devices are defined as heterogeneous virtualized components deployed in various environments, ranging from resource-rich machines (e.g., Cloudlet, IOx) to resource-poor equipment (e.g., access points, smartphones), and range from high-speed links to wireless access. Efforts to create a set of standardized architectures for fog devices have just started. Bonomi *et al.* [5] presented a sketch of the software architecture and highlighted its technological components. Based on the commercial products of fog devices, the architecture supporting future IoT applications is present, as shown in Fig. 2.

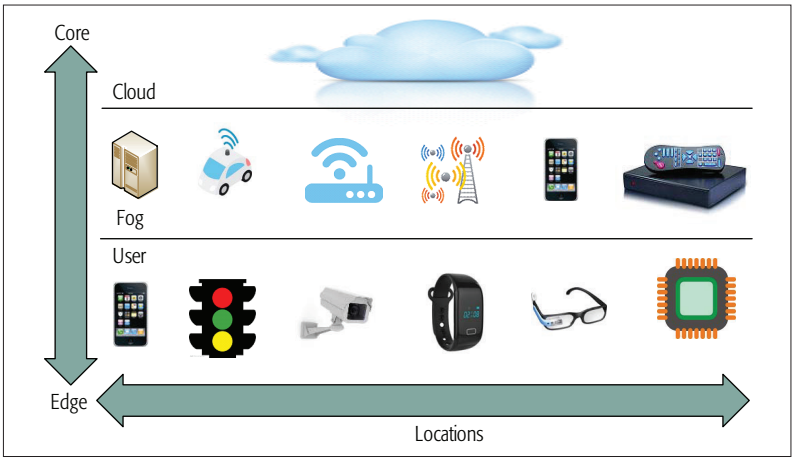

Figure 1. A three-layer user-fog-cloud hierarchy.

	Bonomi *et al.* [2]	Vaquero *et al.* [3]	Yi *et al.* [4]
Heterogeneity	N/A	Yes	Yes
Ubiquity	N/A	Yes	Yes
Distribution	Yes	Yes	Yes
Cloud interaction	Yes	N/A	Yes

Table 1. Comparisons of fog computing definitions.

Fog devices contain three key components: the application layer, policy management layer, and abstraction layer. The application layer consists of products or applications that could be rented for use. Multiple clients are enabled to host their applications on a single fog computing instance. The policy management layer is composed of a policy sublayer and a fog sublayer, which are in charge of policy management, and task and resource management, respectively. The abstraction layer relies on virtualization technologies and discloses universal application programming interfaces (APIs).

Characteristics of Fog Computing

Table 2 compares fog computing's characteristics with those of cloud computing to highlight the advantages, and the excellent characteristics of fog computing are listed below.

Proximity to users: According to the locations of mobile users, their service demands are predictable. For instance, a mobile user in a shopping mall is often more interested in local restaurants, sales, and so on. Fog computing solves this problem by arranging fog devices in the shopping mall and pre-caching local contents to offer high-rate local services.

Geographical distribution: Compared to centralized cloud computing, widespread deployments are needed for applications and services of IoT in fog computing. For example, it can play an active role in offering data streaming with high quality to high-speed vehicles, via access points or proxies located along tracks and highways.

Support for mobility and IoT: Because of the wide geographical distribution, a very large number of mobile devices (e.g., vehicles, smartphones) are connected to a network. Thus, communicating with mobile devices directly is essential for many fog applications, especially for IoT.

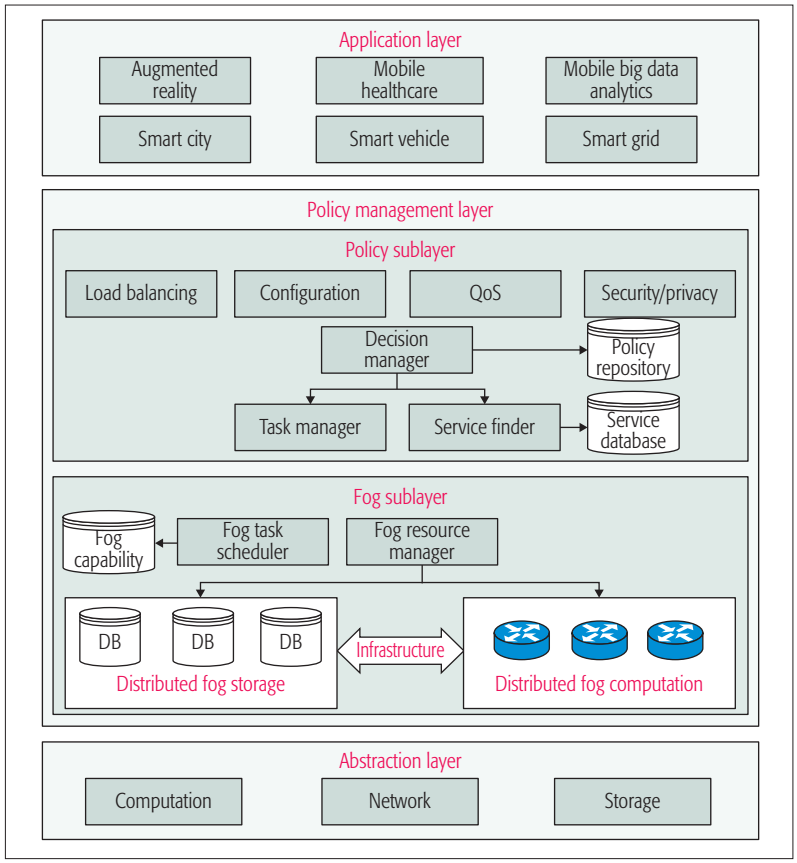

Figure 2. The soft architecture of fog devices.

Access Control in Fog Computing

In fog computing, users pay for the services provided by fog/cloud as tenants; thus, users, fog devices, and cloud service providers, in their own interest, do not trust each other. Access control problems in fog computing are classified into three types:

1. If users want to use the storage and computation services, they must be authorized by the fog/cloud, and some policies should be used to control access to data and services.
2. The fog and cloud need access control reciprocally.
3. Virtual machines (VMs) need an access control mechanism to avoid side-channel attacks.

Therefore, access control in fog computing is an important tool to preserve user privacy and ensure system security, and how to design it to meet design requirements and resource constraints is a challenging task.

Access Control Requirements

In order to build secure and efficient access control in the environment of fog computing, the following requirements have to be taken into consideration, which are the main distinctions between fog computing and cloud computing paradigms.

Latency: The execution time (computation cost), offloading task time, network foraging time, and speed of policy decision can all lead to latency. Providing the end user low-latency-guaranteed services and applications is essential for fog computing. Access control systems in fog computing must grant access decisions in a reasonable time.

Efficiency: Most fog devices are resource-rich, but some are resource-constrained (e.g., handheld devices). Efficiency of access control is still a challenge for any access control system that can implement any policy. It may lead to the decision process being delayed, resulting in unacceptable latency to other parts of the network.

	Cloud computing	Fog computing
Deployment	Network core	Network edge
Ownership	Business entities (BEs)	BEs, individuals
Hardware	Ample and scalable	General limited
Architecture	Centralized	Distributed
Access	Fixed and wireless	Mainly wireless
Target user	General Internet users	Mobile users
Service	Virtualization	Virtualization
Latency	High	Low
Mobility	N/A	Yes
Local awareness	N/A	Yes

Table 2. Characteristic comparisons between cloud and fog.

Generality: Systems and services of fog computing are based on various techniques, with differences in hardware and software. The same abstraction to top-layer services and applications needs to be provided due to the heterogeneity of fog devices. Generic APIs are needed to deal with existing APIs and protocols.

Aggregation: Data is collected by user devices that are geo-distributed. In order to reduce latency, it must be aggregated by fog devices closer to users. The data from user devices may be meaningless; thus, the aggregation must be done intelligently and seamlessly. The authority change of the data before and after aggregation is a challenge.

Privacy protection: Since it is unavoidable to interchange data among different domain administrations, due to the decentralized architecture of fog computing, protecting the privacy of data is a critical requirement in fog access control.

Resource restriction: One of the main requirements of fog access control is resource restriction. This is because the computation resources at the client side and the edge of the network are limited.

Policy management: It is a key part of the architecture of fog computing. As a result, the access control model in fog computing must have the ability to support releasing, invoking, and deleting or creating a policy.

Access Control Models

There are different models to design access control. This subsection describes the existing models and evaluates their possible applications in fog access control.

Discretionary access control (DAC) model: In a DAC model, the data owner has the ability to decide its access permissions for others and set them accordingly, based on users' identities in some group. A DAC model is more flexible and less secure, so it is generally used in environments that emphasizes convenience and does not need a high level of security, such as the UNIX operating system. DAC models are typically applied only in inheritance applications, which will result in a large amount of management overhead in the fog computing environment with multiple applications and multiple users.

Mandatory access control (MAC) model: The MAC model is designed based on the require-

ment of resource-user mapping. Consequently, it is better suited for a distributed system than the DAC model. The MAC model is usually applied in multi-layer security systems, where each object as well as subject is identified with different security levels. The key rules are no-write-down and no-read-up to maintain the confidentiality of information. However, the access right is decided by the system administrator, not by the subject.

Role-based access control (RBAC) model: The motivation behind RBAC is that the responsibility of a subject is more important than who the subject is. In a RBAC model, users can access objects in the system using their roles, which are assigned in view of task functions. The RBAC model is more scalable than the DAC and MAC models, and more geared to use in the fog/cloud computing environment, typically for those who cannot track with fixed identities.

The above models have been developed for allocating user permissions statically. However, the relationship between resources and users is dynamic in fog and cloud computing. To fulfill fog and cloud access control requirements, the traditional access control models are improved as follows.

Attribute-based access control (ABAC) model: Attribute-based encryption (ABE) is deemed to be an appropriate technique for the access control problems in cloud computing, as it can protect data privacy and grant the data owner to set the access policy directly. Based on it, the ABAC model was proposed to satisfy the security and flexibility of cloud computing. The most important feature of ABAC is fine-grained access control, where the data is related to access policies, and users are assigned some attributes. The user is able to access data only when his/her attributes satisfy the specified access policy.

As described in Fig. 3, the data owner defines an access policy (e.g., (*CS* AND *PhD*) OR *Prof*) and encrypts the data according to the policy. Every user obtains a secret key based on his/her attributes; for example, the attributes of Alice and Bob are {*CS*, *Prof*} and {*EE*, *PhD*}, respectively. In the decryption phase, one can decrypt the ciphertext only when the attributes embedded in the secret key satisfy the access policy defined in the ciphertext. From Fig. 3, Alice can decrypt the ciphertext successfully and access the data, but Bob cannot.

Usage-control-based access control (UCON) model: The main point of UCON is to manage sessions used by users after access rights have been granted. The characteristics of UCON include:

1. Attribute mutability indicates that attribute update will be dealt with.
2. Control continuity indicates that an access decision will be evaluated.

The UCON model is a pattern based on attributes, and its access rights to resources are assigned predicated on object, subject, or environment properties, which are defined in the form of obligation policy, condition, and authorization.

Reference monitoring access control (RMAC) model: RMAC consists of a set of reference validation mechanisms and access decision requests generated by a policy decision point that enforces an access control policy over subjects' ability to process operations on distributed objects. The reference validation mechanism has to meet several features, such as evaluability, invocability, non-bypass ability, and tamper-proof ability. However, the RMAC model has a central-based architecture in traditional systems, which will incur high computation cost and significant latency when executing all access requests; thus, it is unsuitable for dynamic distributed systems such as fog computing.

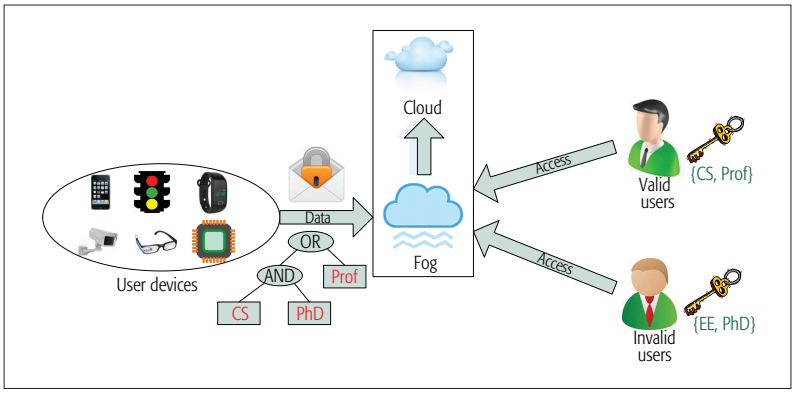

Figure 3. ABE-based access control.

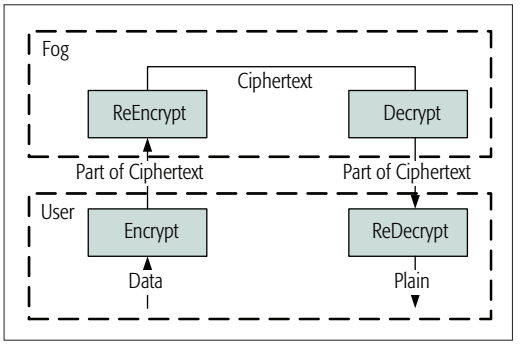

Figure 4 CP-ABE-based access control supporting outsourcing.

Proxy re-encryption (PRE) model: The PRE model has been designed as a cryptographic primitive to convert a ciphertext with Alice's key into another ciphertext with Bob's key by using a proxy. The main application of PRE is to provide a secure distributed storage system (i.e., fog computing). To propose a secure and efficient PRE scheme for fog computing, there are some requirements that need to be considered, such as unidirectionality, non-interactivity, proxy invisibility, key optimality, non-transferability, collusion safety, and chosen-ciphertext safeness.

State-of-the-Art Access Control in Fog Computing

Access control was well studied in the context of cloud computing, machine-to-machine communications, and smart grids. However, as we know, fog devices are deployed at the edge of the network, so these studies may not be suitable in fog computing. The work environment of fog devices faces a host of threats that do not exist in a well governed cloud. So far, there are few studies investigating the progress of access control in various edge paradigms, including fog computing.

The OPENi framework[1] is a personal cloudlet and APIs framework, where the personal cloudlet is a virtual space that stores data securely and allows users to control their data, and frictionless

Sharing resources among potential untrusted tenants will increase the risk of side-channel attacks, which makes us need to rethink problems of access control and security caused by virtualization and multi-tenancy. Additionally, the interference of multi-tenancy computation may lead to unauthorized information flow, and the same problems also exist in cloud computing.

interoperability among cloud computing services is allowed in the APIs frame diagram. In this framework, the owner of the cloudlet creates and stores access control lists in the NoSQL database for setting access permissions for each resource based on OAuth 2.0.

VORTEX[2] is a data sharing platform made for IoT by Prismtech. Via IoT supporting devices, fog, cloud, and web applications, data can be shared in a timely, efficient, and seamless manner. Besides addressing privacy and security problems, VORTEX also supports symmetric and asymmetric authentication, and fine-grained access control.

To support secure sharing and communication for fog computing, D'souza *et al.* [6] designed a policy management framework. However, the proposal is just a preliminary framework, and the details of how to build the policy repository, identity a user, make a decision, and protect identity and data privacy are not mentioned.

Salonikias *et al.* [7] presented a distributed RMAC scheme for fog computing. In this scheme, security policies and attributes are preserved in a distributed policy information point (PIP) and the policy decision point (PDP) that is in charge of access control evaluation implemented on fog devices, and the policy enforcement point (PEP) that enforces the access decisions has to be implemented on the edge of the network. They also proposed a policy propagation control method to synchronize the listed policies in the PDP when the PIP's policies are updated.

Stojmenovic *et al.* [8] discussed the authentication and authorization problems between cloud and fog and among fog devices. Using cryptographic primitives such as ABE, even if the link between fog and cloud is fragile, the user can also be authenticated and authorized to fog devices.

Li *et al.* [9] referred the attributes (e.g., location, Wi-Fi network, unlock failures) collected via smart devices as dynamic attributes, and proposed an algorithm that incorporated dynamic attributes within the ABE scheme for access control to verify the access authority in real time.

Zaghdoudi *et al.* [10] used distributed hash tables to construct a generic, robust, and scalable access control scheme for ad hoc mobile cloud computing (MCC) and fog computing, where the access control model is applicable to create spontaneous networks in the case of mobile infrastructures with widespread availability, and responds to MCC access control demands.

Mollah *et al.* [11] proposed a lightweight cryptographic scheme to achieve access control and data sharing, where all security-oriented operations are offloaded to nearby edge servers (e.g., fog devices), which are hosted within one hop of the end devices in IoT.

Challenges and Future Research Directions

In this section, we present some important but challenging problems, and outline possible research directions.

ABE-Based Access Control

ABE-based access control is described in Fig. 3. The data owner is the only person who formulates access policies through attributes, which indicate what kind of users can or cannot access the data. ABE-based access control meets the requirement that users, fog, and cloud are in different security domains. Meanwhile, all data is encrypted and stored in the fog devices and cloud servers; thus, side-channel attacks because of sharing of physical resources are in vain, and access control can work for multi-tenancy and virtualization environments. Some researchers [8, 9] believe that ABE is an appropriate technique for access control in fog and cloud computing.

However, due to the unique requirements of fog computing, access control schemes of cloud computing perhaps not suitable for fog computing directly. Constructing the ABE-based access control that works best for fog computing deserves further research. To realize fine-grained, cryptographically enforced access control in fog computing, problems such as latency and policy management have remained the most significant challenges. Some open problems in this area are as follows.

Outsourcing heavy computing: There are four algorithms in the typical ABE scheme: setup, keygen, encrypt, and decrypt. Generally, the length of ciphertext is related to the attribute number in the access policy, and decryption operations are associated with the number of attributes in users' secret keys. In fog computing, data is generated and encrypted by some wearable devices with limited computing power, and decrypted by mobile users. Due to access control, the computation amount for the user is heavy. On the other hand, fog devices are close to users, and have much more computing power than users. As shown in Fig. 4, fog devices can be used to store the ciphertext and execute some of the encryption and decryption, so as to relieve the computation overhead of users. Thus, ABE-based access control supporting encryption and decryption outsourcing may be more suited to fog computing.

Managing the access policies: In fog computing, access structures and users' attributes change. ABE-based access control must have the ability to support creating, revoking, and updating the attributes of access structures and users. Particularly, for any multiuser encryption system, the revocation technique is necessary to cope with malicious behaviors. The revocation technique in an ABE scheme is more complex than that in a traditional public key scheme. For instance, in an ABE scheme, attributes that users use to generate their secret keys may be the same, which increases the difficulty in designing the revocation technique. In an ABE scheme, the revocation technique generally falls into two categories: attribute revocation and user revocation. At present, basically two methods can achieve revocation: indirect and direct revocation. Whether the fog is in the best position to execute indirect revocation, and how the fog and cloud cooperate with each other during revocation need to be further researched.

Access Control for Multi-Tenancy and Virtualization

Virtualization is a primary technique for providing isolated environments in fog computing as well as a major factor in fog node performance. Sharing resources among potential untrusted tenants will increase the risk of side-channel attacks, which makes us need to rethink the problems of access control and security caused by virtualization and

[1] https://opensourceprojects.eu/p/openi/, accessed 13 Aug. 2017.

[2] http://www.prismtech.com/vortex/, accessed 13 Aug. 2017.)

multi-tenancy. Additionally, the interference of multi-tenancy computation may lead to unauthorized information flow; the same problems also exist in cloud computing.

As a hypervisor is scalable and software programmable, it is network-independent, including network topology, network routing, and network addressing, and it has the ability to discourage unwanted traffic before it reaches the network. It is used to propose an access control approach with multi-tenancy called CloudPolice [12], which is said to be more robust and scalable than network-based technologies. Several solutions could be envisioned to promote hypervisor-based access control policies. A naive method is to have all policies and the entire mapping deployed in all hypervisors; thus, the destination policy can be directly applied to all the flows. But this is not a scalable approach. Another naive method is to use a centralized repository for group membership and policies. Hypervisors access this repository to cache access control policies and determine each new flow. However, this centralized service has to maintain low response time and high availability, and it may become a target for the denial-of-service attack. One way to handle scalability is to allow the hypervisors to reconcile and push back rate limiting or a packet dropping filter in accordance with the access control policies of the hypervisor of destination VMs to that of source VMs.

Conclusion

Relying on fog devices on the edge of the network that have more processing capacity than user devices and are closer to users than the cloud, fog computing reduces latency of applications. In view of the advantages of fog computing, smart transportation systems, vehicular ad hoc networks, *et al.* may turn to the fog computing paradigm in order to obtain better QoS. Thus, the access control issues in these emerging application networks are studied. In this article, we have carried out a survey of access control in fog computing. First, we have introduced the developing definitions and the soft architecture of fog computing, as well as its characteristics distinguishing it from the cloud. Then we have analyzed the popular access control models, access control problems, and possible access control requirements in fog computing. Next, recent access control schemes have been surveyed, and some open research problems have been explored.

Acknowledgment

This work was supported by the Science and Technology Plan of Shenzhen, China (JCYJ20160307150216309, JCYJ20170302151321095), and Tencent Rhinoceros Birds — Scientific Research Foundation for Young Teachers of Shenzhen University, China.

References

[1] J. Zheng *et al.*, "The Internet of Things," *IEEE Commun. Mag.*, vol. 49, no. 11, Nov. 2011, pp. 30–31.
[2] F. Bonomi *et al.*, "Fog Computing and Its Role in the Internet of Things," *1st ACM MCC Wksp. Mobile Cloud Computing, MCC@SIGCOMM 2012*, 2012, pp. 13–16.
[3] L. M. Vaquero and L. Rodero-Merino, "Finding Your Way in the Fog: Towards a Comprehensive Definition of Fog Computing," *ACM SIGCOMM Comp. Commun. Rev.*, vol. 44, no. 5, 2014, pp. 27–32.
[4] S. Yi *et al.*, "Fog Computing: Platform and Applications," *3rd IEEE Wksp. Hot Topics in Web Systems and Technologies*, 2015, pp. 73–78.
[5] F. Bonomi *et al.*, "Fog Computing: A Platform for Internet of Things and Analytics," *Big Data and Internet of Things: A Roadmap for Smart Environments*, ser. Studies in Computational Intelligence. Springer, 2014, vol. 546, pp. 169–86.
[6] C. D'Souza, G. Ahn, and M. Taguinod, "Policy-Driven Security Management for Fog Computing: Preliminary Framework and a Case Study," *15th IEEE Int'l. Conf. Info. Reuse and Integration*, IRI 2014, 2014, pp. 16–23.
[7] S. Salonikias, I. Mavridis, and D. Gritzalis, "Access Control Issues in Utilizing Fog Computing for Transport Infrastructure," *10th Int'l. Conf. Critical Info. Infrastructures Security*, CRITIS 2015, ser. LNCS, vol. 9578, Springer, 2015, pp. 15–26.
[8] I. Stojmenovic *et al.*, "An Overview of Fog Computing and Its Security Issues," *Concurrency and Computation: Practice and Experience*, vol. 28, 2016, pp. 2291–3005.
[9] F. Li *et al.*, "Robust Access Control Framework for Mobile Cloud Computing Network," *Comp. Commun.*, vol. 68, 2015, pp. 61–72.
[10] B. Zaghdoudi, H. K. Ayed, and W. Harizi, "Generic Access Control System for Ad Hoc MCC and Fog Computing," *15th Int'l. Conf. Cryptology and Network Security, CANS 2016*, ser. LNCS, vol. 10052. Springer, 2016, pp. 400–15.
[11] M. B. Mollah, M. A. K. Azad, and A. Vasilakos, "Secure Data Sharing and Searching at the Edge of Cloud-Assisted Internet of Things," *IEEE Cloud Computing*, vol. 4, no. 1, 2017, pp. 34–42.
[12] L. Popa *et al.*, "Cloudpolice: Taking Access Control Out of the Network," *9th ACM Wksp. Hot Topics in Networks*, 2010, pp. 1–6.

Biographies

Peng Zhang (zhangp@szu.edu.cn) received her B.S. degree from the Naval University of Engineering, Wuhan, China, in 2005, and her M.S. and Ph.D. degrees from Shenzhen University, China, in 2008 and 2011, respectively. Now she is a lecturer at the College of Information Engineering, Shenzhen University. She has authored over 30 journal and conference papers and 20 patents. Her current research interests include attribute-based encryption, homomorphic encryption, security in cloud/fog computing, privacy-preserving machine learning, and blockchain.

Joseph K. Liu (joseph.liu@monash.edu) received his Ph.D. degree in information engineering from the Chinese University of Hong Kong in 2004, specializing in cyber security, protocols for securing lightweight devices, and applied cryptography. He is currently a senior lecturer with the Faculty of Information Technology, Monash University, Australia. His current technical focus is particularly cyber security in the cloud computing paradigm, blockchain, lightweight security, and privacy enhanced technology. His papers have received more than 4000 citations and his H-index is 36. He is the corresponding author of this paper.

F. Richard Yu [S'00, M'04, SM'08] (richard.yu@carleton.ca) is a professor at Carleton University, Canada. His research interests include connected vehicles, security, and green ICT. He serves on the Editorial Boards of several journals, including Co-Editor-in-Chief for *Ad Hoc & Sensor Wireless Networks*, and Lead Series Editor for *IEEE Transactions on Vehicular Technology*, *IEEE Transactions on Green Communications and Networking*, and *IEEE Communications Surveys & Tutorials*. He is a Distinguished Lecturer and a member of Board of Governors of the IEEE Vehicular Technology Society.

Mehdi Sookhak (m.sookhak@ieee.org) received his Ph.D. degree in computer science from the University of Malaya in 2015. He is currently a postdoctoral fellow at Carleton University funded by the Canadian Natural Sciences and Engineering Research Council. His areas of interest include cryptography and information security, mobile cloud computing, fog computing, vehicular cloud computing, data storage security, access control, and distributed systems.

Man Ho Au (csallen@comp.polyu.edu.hk) received his Ph.D. degree from the University of Wollongong, Australia, in 2009. He was a lecturer with the School of Computer Science and Software Engineering, University of Wollongong. He is currently an assistant professor with the Department of Computing, Hong Kong Polytechnic University. He works in the area of information security and applied cryptography. He has authored over 80 refereed journal and conference papers.

Xiapu Luo (csxluo@comp.polyu.edu.hk) received his Ph.D. degree in computer science from Hong Kong Polytechnic University in 2007 and was with the Georgia Institute of Technology as a postdoctoral research fellow. He is currently a research assistant professor with the Department of Computing, Hong Kong Polytechnic University. He is also a researcher with the Shenzhen Research Institute, Hong Kong Polytechnic University. His current research focuses on network security and privacy, Internet measurement, and smartphone security.

In fog computing, access structures and users' attributes cloud change. ABE based access control must have the ability to support creating, revoking and updating the attributes of access structures and users. Particularly, for any multiuser encryption system, the revocation technique is necessary to cope with malicious behaviors.