2018 IEEE International Conference on Smart Cloud

A Fog-Based Digital Forensics Investigation Framework for IoT Systems

Eyhab Al-Masri
School of Engineering and Technology
University of Washington Tacoma
Tacoma, Washington, USA
ealmasri@uw.edu

Yan Bai
School of Engineering and Technology
University of Washington Tacoma
Tacoma, Washington, USA
yanb@uw.edu

Juan Li
Computer Science Department
North Dakota State University
Fargo, ND, USA
j.li@ndsu.edu

***Abstract*—The increasing number of IoT devices is prompting the need to investigate digital forensic techniques that can be efficiently applied to solve computer-related crimes involving IoT devices. In digital forensics, it is common for forensic investigators to consider computing hardware and operating systems for forensic data acquisition. However, applying current forensic data acquisition techniques for further digital evidence analysis may not be applicable to some IoT devices. It is becoming increasingly challenging to determine what type of data should be collected from IoT devices and how traces from such devices can be leveraged by forensic investigators. In this paper, we introduce a fog-based IoT forensic framework (FoBI) that attempts to address the key challenges associated with digital IoT forensics. Throughout this paper, we discuss the overall architecture, use cases and implementation details of FoBI. We further use our FoBI framework to provide insights on improving the digital forensics processes involving IoT systems.**

Keywords—cloud computing, cloud forensics, IoT devices, IoT paradigm, digital forensics, forensic investigator

I. Introduction

The acquisition and analysis of any form of digital information that can be used as evidence in civil, criminal or administrative cases is what is known as computer forensics [1]. State and federal courts have developed policies and guidelines on the acquisition of digital evidence. Such digital evidence involves investigating computing hardware, including hard drives and storage media devices, for data gathering or tracing usage. For many years, digital forensics have relied on the file system for mining file metadata (e.g. date created, modified, author, etc.), examining deleted files, performing keyword matching and looking for common user habits or activities. As data storage evolved beyond local hard drives to include cloud-based storage, applying traditional digital forensic tools is no longer useful or applicable.

As the number of cloud-based customers continues to grow, the dependency on cloud-storage media becomes more evident. This means that digital forensic tools need to process large magnitudes of data in order to mine data. This requires forensic investigators to be trained and skilled on how to utilize the cloud for collecting evidence. Furthermore, forensic tools can be time consuming particularly when examining large volumes of digital evidence. This makes forensic examinations slow resulting in forensic data recovery becoming increasingly complex and difficult, particularly in the case of cloud computing where data can reside across many distributed locations.

In the case of the Internet of Things (IoT), devices can be connected to the Internet exchanging and sharing users' personal data. This makes IoT devices vulnerable to cyberattacks. Very little or no research is conducted into protecting the IoT devices from these threats or attacks. We are putting at risk personal devices such as video cameras, refrigerators, televisions, wearable fitness devices, among many others which can be used by cybercriminals for malicious attacks [2].

Unfortunately, traditional digital forensic techniques and tools that mainly depend on file systems to forensically recover data may no longer be effective in the IoT paradigm. For example, in carrying out a malicious attack, a cybercriminal may inject or install a malware via a webcam connected to an organization's network instead of directly injecting it via a computer of the organization. This makes digital forensic difficult and time consuming as it is hard to determine the source of the attack [3]. Furthermore, IoT forensics involves both software and hardware. It requires applying reverse engineering techniques to examine sources of malicious attacks and how they were performed [4]. The complexity in reverse engineering IoT vulnerabilities also requires extensive, concrete experiences in the problem domain [5].

The "Learning by experience" or what is often referred to as "Experiential Learning Model" (ELM) considers a cycle of four basic elements including: (a) concrete experience, (b) observation and reflection on that experience, (c) formation of abstract concepts based upon the reflection and (d) testing new concepts [6]. For IoT forensics, recovering data or analyzing log files may not be efficient. That is, an IoT system's makeup involves a mixture of different hardware and software components that differ in structure or assembly. This makes IoT forensics a backward problem-solving process that requires systematic analysis of the structure of devices, their function and mechanical operation [6]. Therefore, IoT forensics is not about collecting or recovering data or decoding the traces of a malicious attack, but the analysis or observations of the behavior of objects rather than synthesis. That is, assessing the capability of IoT devices is more critical than trying to understand what they are performing. This involves applying approaches such as learning by experience to continuously perform systematic analysis of an IoT system for effective case reporting.

It is common for forensic software tools to create a case file that keeps a record of all case-specific data involved in an investigation. For example, EnCase software [7] helps examiners or investigators collect and analyze the evidence they need. As part of this evidence, EnCase software collects files, network logs, computer usage data, among other elements. The ability to collect this data means that the investigation software (i.e. EnCase) has full access to the devices that are part of the evidence. However, in an IoT investigation, IoT devices may not be accessible to the investigation software and an attack may be carried out via a

978-1-5386-8000-1/18/$31.00 ©2018 IEEE
DOI 10.1109/SmartCloud.2018.00040

cloud computing environment which is outside the scope and reach of the investigation software.

As some companies project an increase of IoT devices reaching fifty billion devices by 2020 [8], it is imperative to develop techniques that will help a digital forensic investigation preserve digital evidence at an early stage. Unfortunately, digital forensics for the IoT paradigm has not been well studied. This paper introduces a novel model of digital forensics (FoBI) that uses fog computing to effectively search and preserve evidence and defend against a cyber-attack on an IoT system. The rest of this paper is organized as follows. Section II reviews some of the related work. Section III discusses some of the key challenges associated with IoT forensics. Section IV introduces our fog-based digital forensics investigation framework (FoBI). In this section, we also provide use cases and implementation details of FobBI. Finally, Section V provides conclusion and future work insights.

II. Related Work

A number of researchers have investigated the benefits and challenges associated with digital forensics in the cloud computing paradigm. Since the Internet of Things (IoT) is directly dependent on the cloud computing paradigm, it is imperative to consider the challenges associated with IoT devices connected to the cloud. In the following sections, we introduce some of the major related work in the area of digital forensics in cloud computing and IoT.

A. Digital Forensics in Cloud Computing

Biggs at al. discussed the impact of cloud computing on digital forensic investigations as part of a research project called CLOIDFIN [9]. The authors call for an international joint effort that incorporates major changes required for digital investigations to be effective in the cloud computing model. The authors also recommended cloud service provider vendors (e.g. SaaS, PaaS or IaaS) to ensure that policies are enforced, preventing non-compliance. However, the authors did not provide a concrete solution to the problem of digital investigation evidence preservation in the cloud computing environment.

In another paper, the authors in [10] introduced a model that can be used to archive data for medical service providers and hospitals [10]. This model focuses primarily on the Amazon security model for the purpose of archiving and data recovery and addresses a limited set security issues. In particular, the paper provides a backup and recovery model that is supplied by nearly all cloud service providers. Hence, it is not clear how this model which focuses on the archiving medical service providers' data can provide a general solution that would help examiners in non-medical investigation digital forensic cases.

In [11], the authors discuss inherent challenges in cloud forensics. The paper examines the possible use of cloud computing services to complete a standard digital investigation. The authors acknowledge that the evidence to be collected across multiple cloud deployment models via virtual machines (VMs) can significantly vary [11]. The paper recommends service-specific solutions. For example, for IaaS environments, the authors recommend snapshot analysis (or forensic image) cloning. The authors also provide suggested solutions based on different use cases such as Service Level Agreeement (SLA) verification, compliance issues, loss of evidence data, among others. The problem, however, is that solutions such as forensic image cloning may not be realistic if all cloud service providers will be required to clone virtual images. This translates into a significant increase in the amount of storage required on the cloud. Furthermore, the case-by-case analysis does not help in finding a powerful approach that can be applied in cyber-attacks involving cloud computing resources.

Although digital forensics in cloud computing is an important area of research, majority of the existing research work has focused on outlining the challenges associated with performing digital investigations on the cloud [12-18]. Very little research has been conducted into proposing models or solutions that can potentially solve the inherent problems in the preservation of digital evidence at the cloud computing level. Most of the current research has focused on the access control and data storage integrity or security [19-21]. With the introduction of the Internet of Things (IoT) that makes extensive use of the cloud computing model, the challenges become more evident and require models that can potentially solve the problem for both paradigms.

B. IoT Digital Forensics

Digital forensics involving IoT systems or IoT devices is still in its early stages. Nonetheless, several researchers on separate efforts have attempted to address issues related to the gathering of information or data in IoT systems. Shin et al. provided a summary of data collection and analysis methods and suggested potential research areas that can improve digital forensic for the IoT ecosystem [22]. The paper provides important forensic information that can be extracted from some of the existing IoT smart devices such as Amazon Echo, Z-Wave devices, and home routers. However, the paper does not provide solutions that can accommodate general-purpose IoT systems or applications. For example, the paper focuses on specific hardware (e.g. Amazon Echo) or network forensic (e.g. home routers).

The authors in [23] introduce a framework called CFIBD-IoT that is cloud-based consisting of three layers: (a) a cloud/IoT infrastructure layer, (b) a forensic evidence isolation layer and (c) a digital forensic investigation layer. The paper recommends the adoption of a standardized mechanism for extracting and isolating evidence such as ISO/IEC 27043 [23]. In [24], the authors propose forensic-aware IoT (FAIoT), a model that aims to support forensic investigators. Zegzhda et al. introduced a theoretical framework that attempts to increase the security of IoT applications through a topological sustainability [25]. Kebande et al. proposed a Digital Forensic Investigation Framework for IoT (called DFIF-IoT) that creates snapshots of virtual images that can later be used by forensic investigators [26]. Other similar approaches were also introduced in literature [27, 28].

Unfortunately, many of the existing limited research work take a passive digital forensic approach rather than a proactive one. That is, if an IoT device malfunctions, it is extremely difficult to determine the status of an IoT device. It would be ideal if there is a mechanism that takes into consideration the monitoring of IoT devices such that this information can be used as forensic evidence. In this paper, we introduce a fog-oriented digital forensic investigation model that helps digital investigators determine the state of an IoT device.

III. Challenges With IoT Forensics

Digital forensics for cloud computing transcends the acquisition of data stored on hardware resources used for the purpose of performing detailed analysis or examining traffic logs. This is due to the fact that many applications deployed in a cloud environment are typically distributed across a number of virtual instances and network resources. Therefore, it is imperative that digital forensic methods take into consideration the distributed nature in cloud computing and adapt to the changes in the way data and applications are deployed to accommodate this distributed nature.

In addition, gaining access to forensic evidence in a cloud environment involves the service provider who may be reluctant to share information or provide investigators access to their cloud-based environments. Even in the case that service providers collaborate with authorities, a cyber-attacker may purposefully remove or distort any traces of a malicious attack. Furthermore, dealing with digital forensic evidence on the cloud may differ based on the cloud environment. For example, the process of acquiring digital evidence differs from an Infrastructure-as-a-Service (IaaS), Platform-as-as-Service (PaaS) and Software-as-a-Service (SaaS). In SaaS and PaaS, the evidence acquisition methodology involves mainly service providers whereas in IaaS it involves both the client(s) and service providers. Through an IaaS, it is possible that clients clone or image virtual machines for forensic investigations. However, cloning in SaaS and PaaS may not be feasible or possible. Furthermore, IoT digital forensics adds another level of complexity for a variety of factors including: (a) the number of distributed IoT devices, (b) the variance in the technical specification of IoT devices, (c) the location of stored data, and (d) lack of evidence data due to IoT device processing and memory constraints.

Additionally, resource allocation in cloud computing environments can be automatically adjusted based on usage or consumption (i.e. auto-scaling). Therefore, it is extremely difficult in a cloud computing environment that a resource would be tampered with or get stolen. However, in an IoT system, it is possible that an IoT device gets tampered with, stolen or loses communication. Once an IoT device goes offline, it is nearly impossible to track the device or determine its current status. To illustrate the importance of this, consider, for example, an IoT security camera that acts as an edge device which continuously captures and processes images for intrusion detection. Assume that a cloud-based dashboard connects to this IoT device. Assume also that the IoT device has been tampered with and loses communication. It is then nearly impossible to resume communication with the device or determine the exact cause of the malfunction. With the overwhelming number of IoT devices that are connected to the cloud, considering the causes that have led to an IoT device to go offline is often neglected or ignored.

In addition, the miniaturization in the size of IoT devices serves as a level of restriction on what an IoT device can perform. Hence, if an IoT device malfunctions for any reason (e.g. tempering, lost, damaged, etc.), an IoT device may not have the capabilities to transmit large amounts of data to the cloud. Furthermore, such constraints restricts these IoT devices to be equipped with proper crypto stacks. As a result, the lack of proper built-in crypto stacks make them easily vulnerable to threats and cyber-attacks. To illustrate this, consider for example a refrigerator that can be controlled remotely via a mobile phone. A mobile phone can be used as a gateway to control or send instructions to the remote refrigerator. The refrigerator may not have a built-in hardware cryptographic component or uses primitive authentication techniques (e.g. authentication via HTTP header) due to its limited computational processing capabilities. This makes the refrigerator vulnerable to cyber-attacks. To overcome this problem, a refrigerator could in this case connect to a fog node that has sufficient computational processing power to receive instructions in a secure manner.

IV. FoBI: Fog-Based IoT Forensic Framework

In fog computing, computing power is distributed to the edge of a network [31]. Broadly speaking, fog computing is a network model in which computing and storage resources are placed at the network edge, in proximity to the mobile and IoT devices [32]. Several critical functions such as data filtering and aggregation are performed on fog [33]. Fog computing offers numerous advantages to IoT systems, such as improved scalability, reduced network latency, faster responsiveness, and potential improvement of security and privacy [34]. Therefore, it is useful to filter traffic and data going to IoT devices. This information can be used as digital evidence in case there is a cyber-attack or threat. Furthermore, by adding this fog layer it becomes possible to determine when an IoT device is malfunctioning and trace its history in such a way that it can mine the data stored locally to determine its status or send alerts in an effective, efficient and scalable manner.

To overcome many of the above-mentioned challenges, we introduce FoBI: fog-based IoT forensic framework. FoBI takes advantage of the fog computing paradigm which helps push intelligence to the edge of a network through a gateway. This is suitable for IoT systems that are data intensive and have a large number of deployed IoT devices. In such cases, some intelligence can be built in the fog node or gateway such that it can filter the data that requires transmission. Using FoBI, it is then possible to recover forensic evidence based on the data interaction between an IoT device and a fog node or gateway. This fog node, for example, can store the last known location of an IoT device. In the case of malfunction, the framework can retrieve log files associated with the malfunctioning device. When FoBI's investigation model analyzes the data and determines a suspicious activity, it will notify other IoT devices or nodes of a potential threat. This way, the threat does not propagate to other IoT devices and limits the cyber-attacker from affecting other IoT devices.

To illustrate this, consider the earlier example about the smart refrigerator as an IoT device. The refrigerator is connected locally to a fog node which filters the data that requires processing or transmission. Assume that this IoT device is part of a home automation system that consists of smart LEDs, clocks, doors, cameras, smart speakers, among others. Assume that a cyber-attacker was able to find a weak link in the network of IoT devices for controlling the LED. Assume that all IoT devices are connected through the fog gateway.

This fog gateway has built-in intelligence that determines if there is a suspicious activity on part of the LED. That is, instructions received by the LED are usually sent via a registered android-based device browser while one of the latest requests has been received via an unknown user-agent. In this case, FoBI analyzes the log files and determines that the user-agent for the past three months is consistent and has not changed. However, the user-agent associated with the latest request does not match this pattern or the one in the registration profile. In such cases, FoBI determine that there exists a potential violation of security and notifies other IoT devices in the registered network via a publish-subscribe messaging protocol such as Message Queuing Telemetry Transport (MQTT) to stop executing instructions remotely until further notice (i.e. until further analysis is completed or the owner is notified). In this manner, FoBI provides a way to prevent further attacks from taking place, making further attacks harder for the attacker, detect an attack as quickly as possible and mitigate the effects of the attack fast. A high-level architecture overview of our framework is shown in Fig. 1.

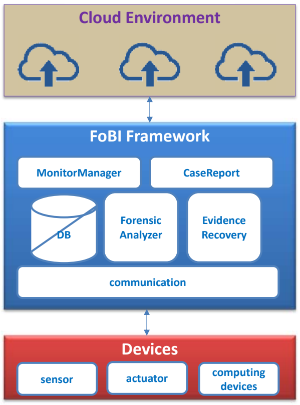

Fig. 1. FoBI Framework Architecture

As illustrated in Fig. 1, FoBI can run on a fog gateway (or node) and consists of six modules: (a) device monitoring manager, (b) forensic analyzer, (c) evidence recovery, (d) case reporting, (e) communication and (f) storage. FoBI continuously communicates with IoT devices via the communication module. The communication module is responsible for properly connecting IoT devices to the framework and setting up the necessary environment for IoT devices to send and receive data. The local storage (DB) is used to store a log of all the activities associated with the communication of IoT devices with the framework.

We assume that each IoT device is associated with a unique identifier (e.g. IP or MAC address). The log serves as a recording of all incoming and outgoing traffic between IoT devices and external networks. The format of the log is similar to that of existing tools that are used to examine network traffic (e.g. tcpdump). It includes, for example, the date, time, packet type, interface type and size. The monitoring manager module continuously monitors traffic or data sent to and from IoT devices. For example, it examines the traffic information to distinguish any suspicious activity such as unusual ports, identifiers, traffic size, among others.

In case a flag is raised by the monitoring manager of some suspicious activity, the forensic analyzer begins collecting clues for further investigation. For instance, assume that multiple IoT devices are connected through registered port number. The monitoring component acknowledges an incoming request that is attempting to connect to an unusual port. In this case, it will raise a flag which will cause the forensic analyzer to retrieve volatile data (i.e. in memory and active processes) and store them in a permanent location. This location can be either locally or on the cloud, depending on how the system is setup or configured. In addition, the forensic analyzer begins cloning any data on the IoT devices into its local storage. If the forensic analyzer determines a suspicious activity has taken place and reached an IoT device, the module disconnects the device and issues a restart signal.

The evidence recovery module is responsible for collecting and gathering data from affected IoT devices. That is, it creates a bit-stream image of all the data residing IoT devices (i.e. makes a forensic copy). It also attempts to determine running processes on the IoT device remotely. Once an IoT device is analyzed, the case report module generates a report determining whether there is a cyber-attack or threat. Based on the evidence collected, this module can issue alerts if a conceivable threat is found.

There are a number of forensic investigation models proposed in literature. However, the design of our FoBI framework is based on the principles of the DFRWS proposed during the 1st Digital Forensic Research Workshop in 2001 [30] which includes: (a) identification, (b) preservation, (c) collection, (d) examination, (e) analysis and (f) presentation. We believe that implementing these modules through a middleware architecture such as that of FoBI would be ideal for fog gateways. In the next two sections, we provide use case scenarios that demonstrate the usefulness of our proposed FoBI framework.

A. Smart Refrigerator Use Case

To illustrate the usefulness of our proposed FoBI framework, consider Bob who owns a smart refrigerator (also referred to as an Internet refrigerator) in his smart home. Assume that this IoT device is among a group of other devices that are part of a smart home which includes an IP camera, smart door lock, smart speaker (e.g. Amazon Echo with Alexa), among others. The refrigerator is connected to a fog gateway which has the FoBI management software running. These devices connect and communicate to FoBI via a Wi-Fi router. Each device has the connectivity information stored locally. Alice, an experienced cyber-attacker, exploits a flaw in the WPA2 cryptographic protocol that enabled her to acknowledge the user name and password of the Wi-Fi network. Alice begins exploiting other vulnerabilities in the network and discovers a number of connected IoT devices. Bob, the owner, usually connects to the smart refrigerator from his phone which is tied geographically to his location. However, Alice attempts to connect to the smart refrigerator from a location that is unknown to FoBI. In this case, a flag is raised by the monitoring manager which begins analyzing the requests made by this unknown user.

The monitoring manager module then determines that an HTTP request was issued from this authorized user from an unusual location with one of the required HTTP header attributes missing. In this case, the monitoring manager component identifies this as a suspicious request and immediately blocks it. It then raises a flag for the forensic analyzer to look into the details of the incoming request and compare it to earlier requests in an attempt to discover any historical evidence or patterns. The analyzer gathers sufficient information to conclude that this is a malicious attack. In this case, it communicates with the CaseReport module which then issues an alert to Bob with the details.

B. *Smart City Nework of Sensors*

In this use case scenario, we assume that a large number of IoT sensors are connected to a citywide network of sensors mounted on lampposts (e.g. Array of Thing system) [29]. Assume that a number of IoT devices are connected to a fog gateway that has the FoBI framework running. FoBI continuously monitors the data transmitted from the IoT devices to the cloud. However, during a usual routine check, FoBI determines that one of the IoT devices is malfunctioning or not responding. There are a number of factors why an IoT device may not be responding such as tampering, hardware/software malfunctioning, theft, damage, among other possible factors. The monitoring manager issues a flag to the forensic analyzer which begins examining historical activities of that particular IoT device.

The forensic analyzer determines that the IoT device has exhibited an inconsistent pattern when transmitting sensor data in the past week. Although the IoT device is programmed to send the fog gateway sensor data every hour, based on the historical activity of this IoT device, FoBI determines that the IoT device has failed to properly transmit data on at least six occasions over the past week. In this case, the forensic analyzer communicates this information with the care report module to generate a report with a low alert priority indicating possible software or hardware malfunctioning on part of the IoT device.

V. Conclusion

In this paper, we presented FoBI, a Fog-Based IoT forensic framework that can identify and mitigate cyber-attacks on IoT systems at early stages. As IoT devices proliferate, the number of security breaches and cyber-attacks will likely to increase. Unfortunately, current forensic methods are not suitable to collect forensic evidence when a cyber-attack involving an IoT system occurs. Throughout the paper, we discuss key challenges associated with cloud-computing and IoT forensics. We also discussed possible computing paradigms such as fog computing that can offer help in solving these challenges. We introduced a fog-based forensic framework that is principally based on the DFRWS Investigative Model. We also discussed the overall architecture of FoBI, use cases, and implementation details. We further used our FoBI framework to provide insights in improving digital forensics for IoT systems. For future work, we plan to evaluate the effectiveness of our FoBI framework by deploying IoT devices in a fog environment.

References

[1] B. Nelson, Guide to Computer Forensics and Investigations, Cengage Learning, 2016, p. 30.

[2] INTERPOL, ‘Internet of Things’ cyber risks tackled during INTERPOL Digital Security Challenge, https://www.interpol.int/News-and-media/News/2018/N2018-007, Last Accessed July 05, 2018.

[3] J. Vacius, B. Darius, G. Elvar, Methods and tools of digital triage in forensic context: survey and future directions, Symmetry. Basel : MDPI AG. ISSN 2073-8994. 2017, vol. 9, iss. 4, article 49, pp. 1-19.

[4] P. Shwartz,Y. Mathov, M. Bohadana, Y. Elovici, Opening Pandora's Box: Effective Techniques for Reverse Engineering IoT Devices, Smart Card Research and Advanced Applications, Springer International Publishing, pp. 1-21, 2017.

[5] D. Connett, Reverse Engineering – IoT Physical Vulnerabilities, DELPHI, cj.msu.edu/assets/ICC-2017-Connett-Reverse-Engineering-IoT-Physical-Vulnerabilities.pdf, Last Accessed July 05, 2018.

[6] Messler, Robert Reverse Engineering: Mechanisms, Structures, Systems & Materials. McGraw-Hill Education, 2014..

[7] Guidance Software (OpenText), https://www.guidancesoftware.com, Last Accessed July 05, 2018.

[8] Evans, D., The Internet of Things: How the Next Evolution of the Internet Is Changing Everything, Cisco Press, cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf, 2011, Last Accessed July 05, 2018.

[9] Biggs, S & Vidalis, S 2009, 'Cloud Computing: The impact on digital forensic investigations', International Conference for Internet Technology and Secured Transactions - ICITST, London, UK, 9-12 Nov. 2009, pp. 1-6.

[10] Ahmed, S & Raja, MYA 2010, 'Tackling cloud security issues and forensics model', High-Capacity Optical Networks and Enabling Technologies - HONET, Cairo, Egypt, 19-21 Dec. 2010, pp. 190-195.

[11] Birk, D & Wegener, C 2011, 'Technical Issues of Forensic Investigations in Cloud Computing Environments', 6th International Workshop on Systematic Approaches to Digital Forensic Engineering - IEEE/SADFE 2011, Oakland, CA, USA, pp. 1 - 10.

[12] K. Choo 2010, 'Cloud computing: Challenges and future directions', Trends & issues in crime & criminal justice, no. 400, pp. 1 - 6.

[13] D. Barrett and G. Kipper, Virtualization and Forensics: A Digital Forensic Investigator’s Guide to Virtual Environments, Syngress, 2010.

[14] S. Hraiz, "Challenges of digital forensic investigation in cloud computing," 2017 8th International Conference on Information Technology (ICIT), Amman, 2017, pp. 568-571.

[15] C. Esposito, A. Castiglione, F. Pop and K. K. R. Choo, "Challenges of Connecting Edge and Cloud Computing: A Security and Forensic Perspective," in IEEE Cloud Computing, vol. 4, no. 2, pp. 13-17, March-April 2017.

[16] V. Casola, A. Castiglione, K. K. R. Choo and C. Esposito, “Healthcare-Related Data in the Cloud: Challenges and Opportunities”, IEEE Cloud Computing, vol. 3, no. 6, pp. 10-14, Nov.-Dec. 2016.

[17] A.A. Atayero, O. Feyisetan, “Security Issues in Cloud Computing: The Potentials of Homomorphic Encryption”. Journal of Emerging Trends in Computing and Information Sciences, vol. 2, no. 10, pp. 546-552, 2011.

[18] F. Zafar et al., “A survey of Cloud computing data integrity schemes: Design challenges, taxonomy and future trends”, Computers & Security, vol. 65, pp. 29-49, March 2017.

[19] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, “Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing”, in IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 5, pp. 847-859, May 2011.

[20] Y. Li, Y. Yu, G. Min, W. Susilo, J. Ni, and K. K. R. Choo, “Fuzzy Identity-Based Data Integrity Auditing for Reliable Cloud Storage Systems”, IEEE Transactions on Dependable and Secure Computing.

[21] Q. Alam, S. U. R. Malik, A. Akhunzada, K. K. R. Choo, S. Tabbasum, and M. Alam, “A Cross Tenant Access Control (CTAC) Model for Cloud Computing: Formal Specification and Verification”, in Press at IEEE Transactions on Information Forensics and Security

[22] C. Shin, P. Chandok, R. Liu, S. J. Nielson and T. R. Leschke, "Potential Forensic Analysis of IoT Data: An Overview of the State-of-the-Art and Future Possibilities," 2017 IEEE International Conference on Internet of Things (iThings), 2017, pp. 705-710.

[23] V. R. Kebande, N. M. Karie and H. S. Venter, "Cloud-Centric Framework for isolating Big data as forensic evidence from IoT infrastructures," 2017 Next Generation Computing Applications (NextComp), Mauritius, 2017, pp. 54-60.

[24] Shams Z. and Ragib H., (2015). FAIoT: Towards Building a Forensics Aware Eco System for the Internet of Things. IEEE International Conference on Services Computing, pp. 279-284, 2015.

[25] D. Zegzhda and T. Stepanova, "Achieving Internet of Things security via providing topological sustainability," 2015 Science and Information Conference (SAI), London, 2015, pp. 269-276.

[26] V. R. Kebande and I. Ray, "A Generic Digital Forensic Investigation Framework for Internet of Things (IoT)," 2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud), Vienna, 2016, pp. 356-362.

[27] S., N.Perumal. M. Norwawi and V. Raman, "Internet of Things(IoT) digital forensic investigation model: Top-down forensic approach methodology," Digital Information Processing and Communications (ICDIPC), 2015 Fifth International Conference on, Sierre, 2015, pp. 19-23.

[28] B.Anggorojati, P. N.Mahalle, N. R.Prasad, & R Prasad, “Capabilitybased access control delegation model on the federated IoT network”. In Wireless Personal Multimedia Communications (WPMC), 2012 15th International Symposium on (pp. 604-608).

[29] City of Chacgo, Array of Things, https://arrayofthings.github.io/, Last Accessed July 5, 2018.

[30] G. Palmer, (2001) "DTR-T001-01 Technical Report. A Road Map for Digital Forensic Research", Digital Forensics Workshop (DFRWS), Utica, New York.

[31] Bonomi, F., Milito, R., Zhu, J. and Addepalli, S., 2012, August. Fog computing and its role in the internet of things. In Proceedings of the first edition of the MCC workshop on Mobile cloud computing (pp. 13-16). ACM.

[32] Luan, T.H., Gao, L., Li, Z., Xiang, Y., Wei, G. and Sun, L., 2015. Fog computing: Focusing on mobile users at the edge. arXiv preprint arXiv:1502.01815.

[33] Stojmenovic, I. and Wen, S., 2014, September. The fog computing paradigm: Scenarios and security issues. In Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on (pp. 1-8). IEEE.

[34] Alrawais, A., Alhothaily, A., Hu, C. and Cheng, X., 2017. Fog computing for the internet of things: Security and privacy issues. IEEE Internet Computing, 21(2), pp.34-42.