Proceedings of the 2014 Federated Conference on
Computer Science and Information Systems pp. 1–8

DOI: 10.15439/2014F503
ACSIS, Vol. 2

The Fog Computing Paradigm: Scenarios and Security Issues

Ivan Stojmenovic
SIT, Deakin University, Burwood, Australia
and
SEECS, University of Ottawa, Canada
Email: stojmenovic@gmail.com

Sheng Wen
School of Information Technology,
Deakin University,
220 Burwood Highway, Burwood, VIC, 3125, Australia
Email: wesheng@deakin.edu.au

***Abstract*—Fog Computing is a paradigm that extends Cloud computing and services to the edge of the network. Similar to Cloud, Fog provides data, compute, storage, and application services to end-users. In this article, we elaborate the motivation and advantages of Fog computing, and analyse its applications in a series of real scenarios, such as Smart Grid, smart traffic lights in vehicular networks and software defined networks. We discuss the state-of-the-art of Fog computing and similar work under the same umbrella. Security and privacy issues are further disclosed according to current Fog computing paradigm. As an example, we study a typical attack, man-in-the-middle attack, for the discussion of security in Fog computing. We investigate the stealthy features of this attack by examining its CPU and memory consumption on Fog device.**

***Index Terms*—Fog Computing, Cloud Computing, Internet of Things, Software Defined Networks.**

I. Introduction

CISCO recently delivered the vision of fog computing to enable applications on billions of connected devices, already connected in the Internet of Things (IoT), to run directly at the network edge [1]. Customers can develop, manage and run software applications on Cisco IOx framework of networked devices, including hardened routers, switches and IP video cameras. Cisco IOx brings the open source Linux and Cisco IOS network operating system together in a single networked device (initially in routers). The open application environment encourages more developers to bring their own applications and connectivity interfaces at the edge of the network. Regardless of Cisco's practices, we first answer the questions of what the Fog computing is and what are the differences between Fog and Cloud.

In Fog computing, services can be hosted at end devices such as set-top-boxes or access points. The infrastructure of this new distributed computing allows applications to run as close as possible to sensed actionable and massive data, coming out of people, processes and thing. Such Fog computing concept, actually a Cloud computing close to the 'ground', creates automated response that drives the value.

Both Cloud and Fog provide data, computation, storage and application services to end-users. However, Fog can be distinguished from Cloud by its proximity to end-users, the dense geographical distribution and its support for mobility [2]. We adopt a simple three level hierarchy as in Figure 1.

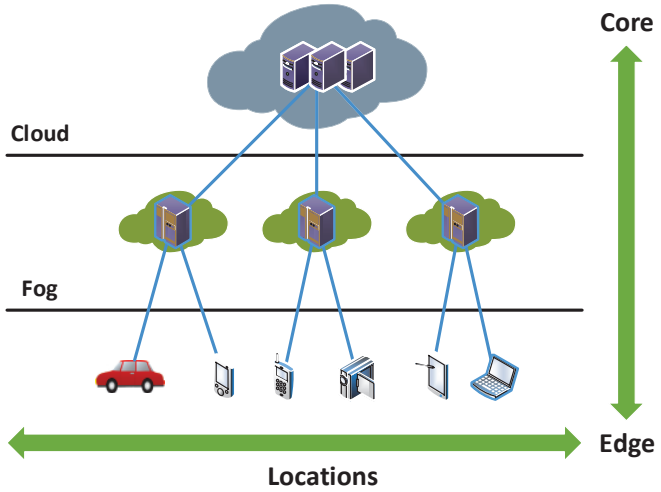

Fig. 1. Fog between edge and cloud.

In this framework, each smart thing is attached to one of Fog devices. Fog devices could be interconnected and each of them is linked to the Cloud.

In this article, we take a close look at the Fog computing paradigm. The goal of this research is to investigate Fog computing advantages for services in several domains, such as Smart Grid, wireless sensor networks, Internet of Things (IoT) and software defined networks (SDNs). We examine the state-of-the-art and disclose some general issues in Fog computing including security, privacy, trust, and service migration among Fog devices and between Fog and Cloud. We finally conclude this article with discussion of future work.

II. Why Do We Need Fog?

In the past few years, Cloud computing has provided many opportunities for enterprises by offering their customers a range of computing services. Current "pay-as-you-go" Cloud computing model becomes an efficient alternative to owning and managing private data centres for customers facing Web applications and batch processing [3]. Cloud computing frees the enterprises and their end users from the specification of many details, such as storage resources, computation limitation and network communication cost. However, this bliss becomes

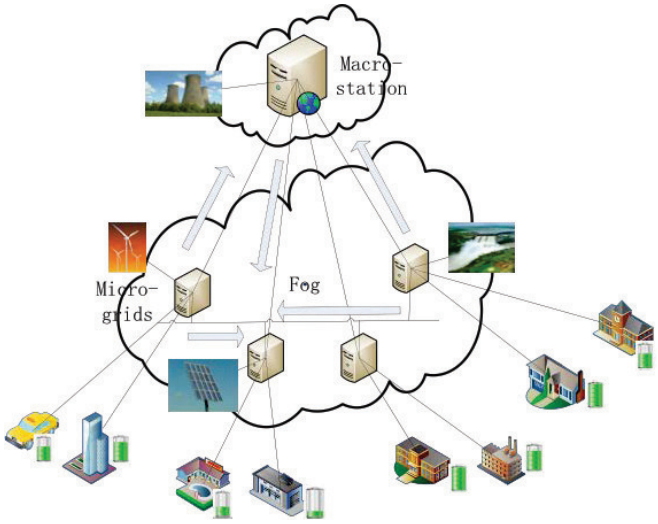

Fig. 2. Fog computing in smart grid.

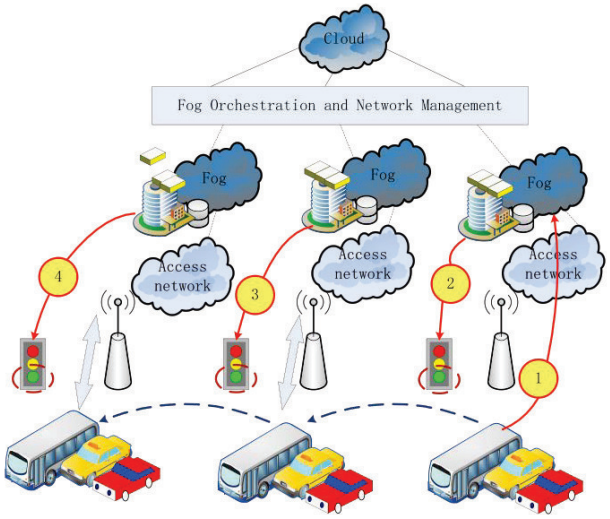

Fig. 3. Fog computing in smart traffic lights and connected vehicles.

a problem for latency-sensitive applications, which require nodes in the vicinity to meet their delay requirements [2]. When techniques and devices of IoT are getting more involved in people's life, current Cloud computing paradigm can hardly satisfy their requirements of mobility support, location awareness and low latency.

Fog computing is proposed to address the above problem [1]. As Fog computing is implemented at the edge of the network, it provides low latency, location awareness, and improves quality-of-services (QoS) for streaming and real time applications. Typical examples include industrial automation, transportation, and networks of sensors and actuators. Moreover, this new infrastructure supports heterogeneity as Fog devices include end-user devices, access points, edge routers and switches. The Fog paradigm is well positioned for real time big data analytics, supports densely distributed data collection points, and provides advantages in entertainment, advertising, personal computing and other applications.

III. What Can We Do With Fog?

We elaborate on the role of Fog computing in the following six motivating scenarios. The advantages of Fog computing satisfy the requirements of applications in these scenarios.

Smart Grid: Energy load balancing applications may run on network edge devices, such as smart meters and micro-grids [4]. Based on energy demand, availability and the lowest price, these devices automatically switch to alternative energies like solar and wind. As shown in Figure 2, Fog collectors at the edge process the data generated by grid sensors and devices, and issue control commands to the actuators [2]. They also filter the data to be consumed locally, and send the rest to the higher tiers for visualization, real-time reports and transactional analytics. Fog supports ephemeral storage at the lowest tier to semi-permanent storage at the highest tier. Global coverage is provided by the Cloud with business intelligence analytics.

Smart Traffic Lights and Connected Vehicles: Video camera that senses an ambulance flashing lights can automatically change street lights to open lanes for the vehicle to pass through traffic. Smart street lights interact locally with sensors and detect presence of pedestrian and bikers, and measure the distance and speed of approaching vehicles. As shown in Figure 3, intelligent lighting turns on once a sensor identifies movement and switches off as traffic passes. Neighbouring smart lights serving as Fog devices coordinate to create green traffic wave and send warning signals to approaching vehicles [2]. Wireless access points like WiFi, 3G, road-side units and smart traffic lights are deployed along the roads. Vehicles-to-Vehicle, vehicle to access points, and access points to access points interactions enrich the application of this scenario.

Wireless Sensor and Actuator Networks: Traditional wireless sensor networks fall short in applications that go beyond sensing and tracking, but require actuators to exert physical actions like opening, closing or even carrying sensors [2]. In this scenario, actuators serving as Fog devices can control the measurement process itself, the stability and the oscillatory behaviours by creating a closed-loop system. For example, in the scenario of self-maintaining trains, sensor monitoring on a train's ball-bearing can detect heat levels, allowing applications to send an automatic alert to the train operator to stop the train at next station for emergency maintenance and avoid potential derailment. In lifesaving air vents scenario, sensors on vents monitor air conditions flowing in and out of mines and automatically change air-flow if conditions become dangerous to miners.

Decentralized Smart Building Control: The applications of this scenario are facilitated by wireless sensors deployed to measure temperature, humidity, or levels of various gases in the building atmosphere. In this case, information can be exchanged among all sensors in a floor, and their readings can be combined to form reliable measurements. Sensors will use distributed decision making and activation at Fog devices to

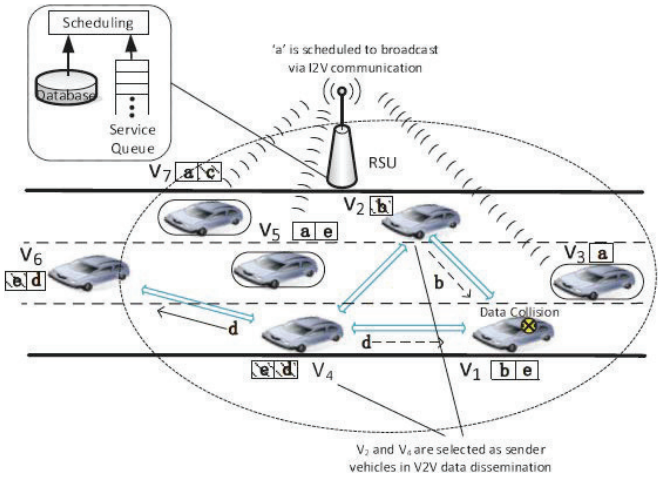

Fig. 4. Fog computing in SDN in vehicular networks [6].

react to data. The system components may then work together to lower the temperature, inject fresh air or open windows. Air conditioners can remove moisture from the air or increase the humidity. Sensors can also trace and react to movements (e.g, by turning light on or off). Fog devices could be assigned at each floor and could collaborate on higher level of actuation. With Fog computing applied in this scenario, smart buildings can maintain their fabric, external and internal environments to conserve energy, water and other resources.

IoT and Cyber-physical systems (CPSs): Fog computing based systems are becoming an important class of IoT and CPSs. Based on the traditional information carriers including Internet and telecommunication network, IoT is a network that can interconnect ordinary physical objects with identified addresses [5]. CPSs feature a tight combination of the system's computational and physical elements. CPSs also coordinate the integration of computer and information centric physical and engineered systems. IoT and CPSs promise to transform our world with new relationships between computer-based control and communication systems, engineered systems and physical reality. Fog computing in this scenario is built on the concepts of embedded systems in which software programs and computers are embedded in devices for reasons other than computation alone. Examples of the devices include toys, cars, medical devices and machinery. The goal is to integrate the abstractions and precision of software and networking with the dynamics, uncertainty and noise in the physical environment. Using the emerging knowledge, principles and methods of CPSs, we will be able to develop new generations of intelligent medical devices and systems, 'smart' highways, buildings, factories, agricultural and robotic systems.

Software Defined Networks (SDN): As shown in Figure 4, Fog computing framework can be applied to implement the SDN concept for vehicular networks. SDN is an emergent computing and networking paradigm, and became one of the most popular topics in IT industry [7]. It separates control and data communication layers. Control is done at a centralized server, and nodes follow communication path decided by the server. The centralized server may need distributed implementation. SDN concept was studied in WLAN, wireless sensor and mesh networks, but they do not involve multi-hop wireless communication, multi-hop routing. Moreover, there is no communication between peers in this scenario. SDN concept together with Fog computing will resolve the main issues in vehicular networks, intermittent connectivity, collisions and high packet loss rate, by augmenting vehicle-to-vehicle with vehicle-to-infrastructure communications and centralized control. SDN concept for vehicular networks is first proposed in [6].

IV. State-of-the-Art

A total of eight articles were identified on the concept of Fog computing [1], [2], [8], [9], [10], [11], [12], [13], [14]. There are some other concepts, not declared as Fog computing, fall under the same umbrella. We will also discuss these work in the subsection of similar work.

A. Related Work

K. Hong et al. proposed mobile Fog in [11]. This is a high level programming model for geo-spatially distributed, large-scale and latency-sensitive future Internet applications. Following the logical structure shown in Figure 1, low-latency processing occurs near the edge while latency-tolerant large-scope aggregation is performed on powerful resources in the core of the network (normally the Cloud). Mobile Fog consists of a set of event handlers and functions that an application can call. Mobile Fog model is not presented as generic model, but is built for particular application, while leaving out functions that deal with technical challenges of involved image processing primitives. Fog computing approach reduces latency and network traffic.

B. Ottenwalder et al. presented a placement and migration method for Cloud and Fog resources providers [13]. It ensures application-defined end-to-end latency restrictions and reduces the network utilization by planning the migration ahead of time. They also show how the application knowledge of the complex event processing system can be used to reduce the required bandwidth of virtual machines during their migration. Network intensive operators are placed on distributed Fog devices while computationally intensive operators are in the Cloud. Migration costs are amortized by selecting migration targets that ensure a low expected network utilization for a sufficiently long time. This work does not optimize workload mobility because Fog devices are also able to carry computationally intensive tasks. It also does not optimize the size of control information or mobility overhead, and does not describe network control policies for finding optimal paths for different applications.

In [11], K. Hong et al. proposed an opportunistic spatio-temporal event processing system that uses prediction-based continuous query handling. Their system predicts future query regions for moving consumers and starts processing events early so that the live situational information is available when

the consumer reaches the future location. Historical events for a location are processed before the mobile user arrives at that location. Live event processing begins at the moment the user arrives. To mitigate large speed of mobile user, authors propose using parallel resources to enable pipeline processing of future locations in several time steps looking ahead. Further, they proposed taking several predictions for each time step and opportunistically compute the events for all of those locations. When the user arrives at that time, the prediction among those that is closest to truth will be selected and its events returned.

J. Zhu et al. applied existing methods for web optimization in a novel manner [14]. Within Fog computing context, these methods can be combined with unique knowledge that is only available at the Fog devices. More dynamic adaptation to the user's conditions can also be accomplished with network edge specific knowledge. As a result, a user's Web page rendering performance is improved beyond that achieved by simply applying those methods at the Web server.

In the mobile Cloud concept [12], pervasive mobile devices share their heterogeneous resources and support services. Neighbouring nodes in a local network form a group called a local Cloud. Nodes share their resources with other nodes in the same local Cloud. A local resource coordinator serving as Fog device is elected from the nodes in each local Cloud. The work [12] proposed an architecture and mathematical framework for heterogeneous resource sharing based on the key idea of service-oriented utility functions. Normally heterogeneous resources are quantified in disparate scales, such as power, bandwidth and latency. However, authors in [12] present a unified framework where all these quantities are equivalently mapped to "time" resources. They formulate optimization problems for maximizing the sum and product of the utility functions, and solve them via convex optimization approaches.

The work [10] first reviews the reliability requirements of Smart Grid, Cloud, and sensors and actuators. This work then combines them towards reliable Fog computing. However, it only concludes that building Fog computing based projects is challenging and does not offer any novel concept for the reliability of the network of smart devices in the Fog computing paradigm.

B. Similar Work

BETaaS [15] proposed replacing Cloud as the resident for machine-to-machine applications by 'local Cloud' of gateways. The 'local Cloud' is composed of devices that provide smart things with connectivity to the Internet, such as smart phones, home routers and road-side units. This enables applications that are limited in time and space to require simple and repetitive interactions. It also enables the applications to respond in consistent manner.

Demand Response Management (DRM) is a key component in the smart grid to effectively reduce power generation costs and user bills. The work [16] addressed the DRM problem in a network of multiple utility companies and consumers where every entity is concerned about maximizing its own benefit. In their model, utility companies communicate with each other, while users receive price information from utility companies and transmit their demand to them. They propose a Stackelberg game [17] between utility companies and end-users to maximize the revenue of each utility company and the payoff of each user. Stackelberg equilibrium of the game has a unique solution. They develop a distributed algorithm which converges to the equilibrium with only local information available for both utility companies and end-users. Utility companies play a non-cooperative game. They inform users whenever they change price, and users then update their demand vectors and inform utility companies. This iterates until convergence. The main drawback of this algorithm is a significant communication overhead between users and utility companies. Though DRM helps to facilitate the reliability of power supply, the smart grid can be susceptible to privacy and security issues because of communication links between the utility companies and the consumers. They study the impact of an attacker who can manipulate the price information from the utility companies, and propose a scheme based on the concept of shared reserve power to improve the grid reliability and ensure its dependability.

The work [18] investigated how energy consumption may be optimized by taking into consideration the interaction between both parties. The energy price model is a function of total energy consumption. The objective function optimizes the difference between the value and cost of energy. The power supplier pulls consumers in a round-robin fashion, and provides them with energy price parameter and current consumption summary vector. Each user then optimizes his own schedule and reports it to the supplier, which in turn updates its energy price parameter before pulling the next consumers. This interaction between the power company and its consumers is modelled through a two-step centralized game, based on which the work [18] proposed the Game-Theoretic Energy Schedule (GTES) method. The objective of the GTES method is to reduce the peak to average power ratio by optimizing the users energy schedules.

The closest work for SDN in vehicular networks are several implementations in wireless sensor network and mesh networks [19], [20]. Moreover, B. Zhou et al. studied adaptive traffic light control for smoothing vehicles' travel and maximizing the traffic throughout for both single and multiple lanes [21], [22]. In addition, the work [23] proposed a three-tier structure for traffic light control. First, an electronic toll collection (ETC) system is employed for collecting road traffic flow data and calculating the recommended speed. Second, radio antennas are installed near the traffic lights. Third, road traffic flow information can be obtained by wireless communication between the antennas and ETC devices. A branch-and-bound-based real-time traffic light control algorithm is designed to smooth vehicles' travels.

V. Security and Privacy in Fog Computing

Security and privacy issues were not studied in the context of fog computing. They were studied in the context of smart grids [24] and machine-to-machine communications [25].

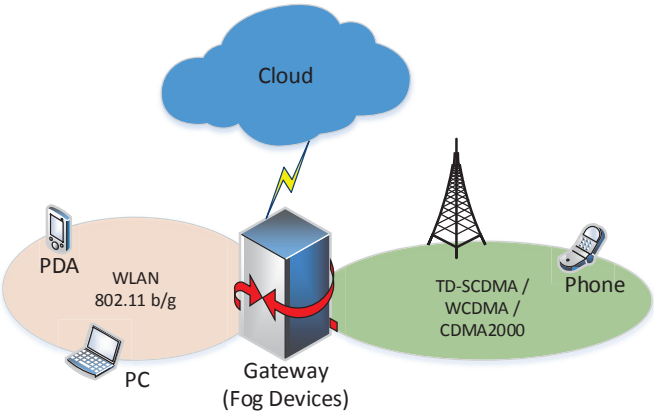

Fig. 5. A scenario for a man-in-the-middle attack towards Fog.

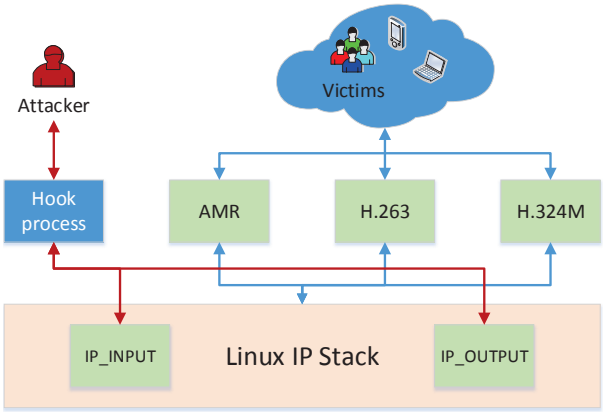

Fig. 6. A system design of man-in-the-middle-attack in Fog.

There are security solutions for Cloud computing. However, they may not suit for Fog computing because Fog devices work at the edge of networks. The working surroundings of Fog devices will face with many threats which do not exist in well managed Cloud. In this section, we discuss the security and privacy issues in Fog Computing.

A. Security Issues

The main security issues are authentication at different levels of gateways as well as (in case of smart grids) at the smart meters installed in the consumer's home. Each smart meter and smart appliance has an IP address. A malicious user can either tamper with its own smart meter, report false readings, or spoof IP addresses. There are some solutions for the authentication problem. The work [26] elaborated public key infrastructure (PKI) based solutions which involve multicast authentication. Some authentication techniques using Diffie-Hellman key exchange have been discussed in [27]. Smart meters encrypt the data and send to the Fog device, such as a home-area network (HAN) gateway. HAN then decrypts the data, aggregates the results and then passes them forward.

Intrusion detection techniques can also be applied in Fog computing [28]. Intrusion in smart grids can be detected using either a signature-based method in which the patterns of behaviour are observed and checked against an already existing database of possible misbehaviours. Intrusion can also be captured by using an anomaly-based method in which an observed behaviour is compared with expected behaviour to check if there is a deviation. The work [29] develops an algorithm that monitors power flow results and detects anomalies in the input values that could have been modified by attacks. The algorithm detects intrusion by using principal component analysis to separate power flow variability into regular and irregular subspaces.

B. An Example: Man-in-the-Middle Attack

Man-in-the-middle attack has potential to become a typical attack in Fog computing. In this subsection, we take man-in-the-middle attack as an example to expose the security problems in Fog computing. In this attack, gateways serving as Fog devices may be compromised or replaced by fake ones [30]. Examples are KFC or Star Bar customers connecting to malicious access points which provide deceptive SSID as public legitimate ones. Private communication of victims will be hijacked once the attackers take the control of gateways.

1) Environment Settings of Stealth Test: Man-in-the-middle attack can be very stealthy in Fog computing paradigm. This type of attack will consume only a small amount of resources in Fog devices, such as negligible CPU utilization and memory consumption. Therefore, traditional anomaly detection methods can hardly expose man-in-the-middle attack without noticeable features of this attack collected from the Fog. In order to examine how stealthy the man-in-the-middle attack can be, we implement an attack environment shown in Figure 5. In this scenario, a 3G user sends a video call to a WLAN user. Since the man-in-the-middle attack requires to control the communication between the 3G user and the WLAN user, the key of this attack is to compromise the gateway which serves as the Fog device.

Two steps are needed to realize the man-in-the-middle attack for the stealth test. First, we need to compromise the gateway, and second, we insert malicious code into the compromised system. For susceptible gateways, we can either refresh the ROM of a normal gateway or place a fake active point in

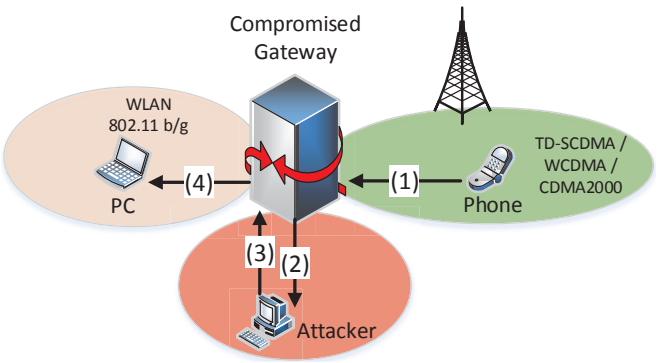

Fig. 7. The hijacked communication in Fog (e.g. from phone to PC).

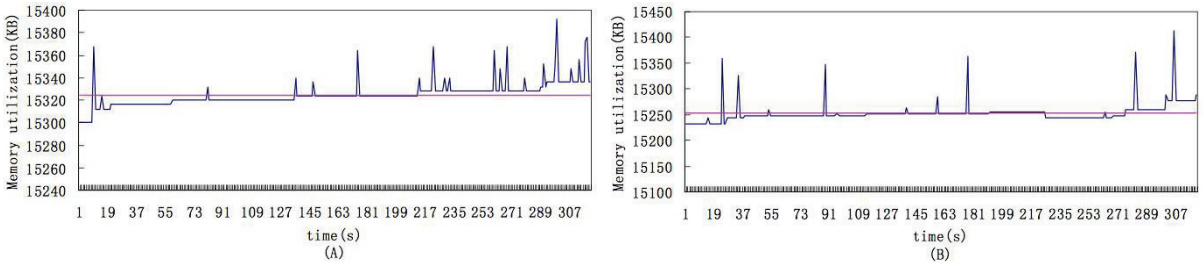

Fig. 8. Memory Consuming of man-in-the-middle-attack in Fog.

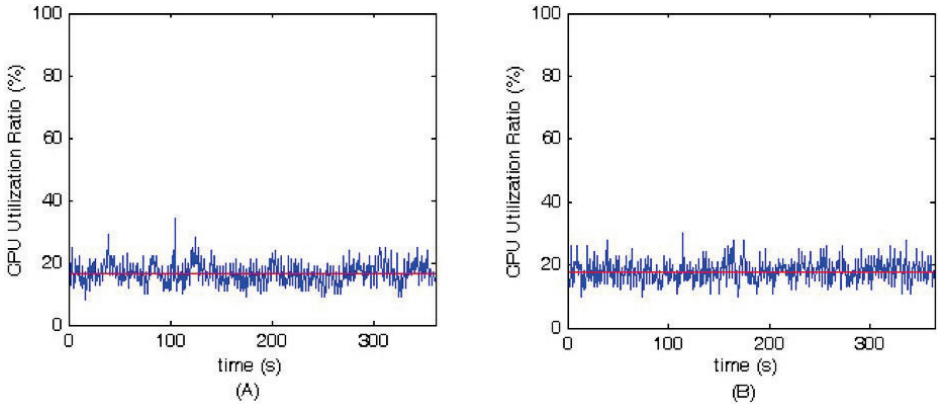

Fig. 9. CPU consuming of man-in-the-middle-attack in Fog.

the environment. Both methods can be easily implemented in the real world, such as in the KFC or Star Bar environments. In our experiment, we choose the former and use Broadcom BCM5354 as the gateway [31]. This device has a high-performance MIPS32 processor, IEEE 802.11 b/g MAC/PHY and USB2.0 controller. Video communication is set up on BCM5354 between a 3G mobile phone and a laptop which adopts Wifi for connection. We refresh the ROM of BCM4354 and update its system to the open-source Linux kernel 2.4.

In order to hijack and replay victims' video communication, we insert a hook program into the TCP/IP stack of the compromised system. Hook is a technique of inserting code into a system call in order to alter it [32]. The typical hook works by replacing the function pointer to the call with its own, then once it is done doing its processing, it will then call the original function pointer. The system structure is implemented in Figure 6. We further employ the relevant APIs and data structures in the system to control the gateway device, such as boot strap, diagnostics and initialization code. The IP packets from WLAN will be transferred to and processed in 3G related modules. We plug a 3G USB modem on BCM5354 device, on which we implement H.324M for video and audio tunnel with 3G CS. H.263 and AMR functions are also implemented as the video and audio codec modules in the system.

2) Work Flow of Man-in-the-Middle Attack: The communication between 3G and WLAN needs a gateway to translate the data of different protocols into the suitable formats. Therefore, all the communication data will firstly arrive at the gateway and then be forwarded to other receivers.

In our experiment, the man-in-the-middle attack is divided into four steps. We illustrate the hijacked communication from 3G to WLAN in Figure 7. In the first two steps, the embedded hook process of the gateway redirects the data received from the 3G user to the attacker. The attacker replays or modifies the data of the communication at his or her own computer, and then send the data back to the gateway. In the final step, the gateway forwards the data from the attacker to the WLAN user. In fact, the communication from the WLAN user will also be redirected to the attacker at first, and then be forwarded by the hook in the gateway to the 3G user. We can see clearly from Figure 7 that the attacker can monitor and modify the data sent from the 3G user to the WLAN user in the 'middle' of the communication.

3) Results of Stealth Test: Traditional anomaly detection techniques rely on the deviation of current communication from the features of normal communication. These features include memory consumption, CPU utilization, bandwidth usage, etc. Therefore, to study the stealth of man-in-the-middle attack, we examine the memory consumption and the CPU utilization of gateway during the attack. If man-in-the-middle attack does not greatly change the features of the communication, it can be proofed to be a stealthy attack. For simplicity, we assume the attacker will only replay the data at his or her own computer but will not modify the data.

Firstly, we compare the memory utilization of gateway before and after a video call tunnel is built in our experiment.

The results are shown in Figure 8, and the red line in plots indicates the average amount of memory consumption. We can see clearly that man-in-the-middle attack does not largely influence the video communication. In Figure 8(A), the average value is 15232 K Bytes, while after we build the video tunnel on gateway, the memory consumption reaches 15324.8 K Bytes in Figure 8(B). Secondly, we show the CPU consumption of gateway in Figure 9. Based on the results in Figure 9, we can also see that man-in-the-middle attack does not largely influence the video communication. In the Figure 8(A), the average value is 16.6704%, while after the video tunnel is built, the CPU consumption reaches 17.9260%. We therefore conclude that man-in-the-middle attack can be very stealthy in Fog computing because of the negligible increases in both memory consumption and CPU utilization in our experiments.

Man-in-the-middle attack is simple to launch but difficult to be addressed. In the real world, it is difficult to protect Fog devices from compromise as the places for the deployment of Fog devices are normally out of religious surveillance. Encrypted communication techniques may also not protect users from this attack since attackers can set up a legitimate terminal and replay the communication without decryption. Particularly, complex encryption and decryption techniques may not be suitable for some scenarios. For example, the encryption and decryption techniques will consume lots of battery power in 3G mobile phones. In fact, this attack is not limited to the scenario of our experiment environment. We can find many applications running in Fog computing are susceptible to man-in-the-middle attack. For example, many Internet users communicate with each other using MSN (Windows Live Massager). The communication data of MSN is normally not encrypted and can be modified in the 'middle'. Future work is needed to address the man-in-the-middle attack in Fog computing.

C. Privacy Issues

In smart grids, privacy issues deal with hiding details, such as what appliance was used at what time, while allowing correct summary information for accurate charging. R. Lu et al. described an efficient and privacy-preserving aggregation scheme for smart grid communications [33]. It uses a super-increasing sequence to structure multi-dimensional data and encrypt the structured data by the homomorphic cryptogram technique. A homomorphic function takes as input the encrypted data from the smart meters and produces an encryption of the aggregated result. The Fog device cannot decrypt the readings from the smart meter and tamper with them. This ensures the privacy of the data collected by smart meters, but does not guarantee that the Fog device transmits the correct report to the other gateways. For data communications from user to smart grid operation center, data aggregation is performed directly on cipher-text at local gateways without decryption, and the aggregation result of the original data can be obtained at the operation center [33]. Authentication cost is reduced by a batch verification technique.

VI. Conclusions and Future Work

We investigate Fog computing advantages for services in several domains, and provide the analysis of the state-of-the-art and security issues in current paradigm. Based on the work of this paper, some innovations in compute and storage may be inspired in the future to handle data intensive services based on the interplay between Fog and Cloud.

Future work will expand on the Fog computing paradigm in Smart Grid. In this scenario, two models for Fog devices can be developed. Independent Fog devices consult directly with the Cloud for periodic updates on price and demands, while interconnected Fog devices may consult each other, and create coalitions for further enhancements.

Next, Fog computing based SDN in vehicular networks will receive due attention. For instance, an optimal scheduling in one communication period, expanded toward all communication periods, has been elaborated in [6]. Traffic light control can also be assisted by the Fog computing concept. Finally, mobility between Fog nodes, and between Fog and Cloud, can be investigated. Unlike traditional data centres, Fog devices are geographically distributed over heterogeneous platforms. Service mobility across platforms needs to be optimized.

References

[1] F. Bonomi, "Connected vehicles, the internet of things, and fog computing," in *The Eighth ACM International Workshop on Vehicular Inter-Networking (VANET)*, Las Vegas, USA, 2011.

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC'12. ACM, 2012, pp. 13–16.

[3] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, Apr 2010.

[4] C. Wei, Z. Fadlullah, N. Kato, and I. Stojmenovic, "On optimally reducing power loss in micro-grids with power storage devices," *IEEE Journal of Selected Areas in Communications*, 2014 to appear.

[5] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[6] K. Liu, J. Ng, V. Lee, S. Son, and I. Stojmenovic, "Cooperative data dissemination in hybrid vehicular networks: Vanet as a software defined network," *Submitted for publication*, 2014.

[7] K. Kirkpatrick, "Software-defined networking," *Commun. ACM*, vol. 56, no. 9, pp. 16–19, Sep. 2013.

[8] Cisco, "Cisco delivers vision of fog computing to accelerate value from billions of connected devices," Cisco, Tech. Rep., Jan. 2014.

[9] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, "Opportunistic spatio-temporal event processing for mobile situation awareness," in *Proceedings of the 7th ACM International Conference on Distributed Event-based Systems*, ser. DEBS'13. ACM, 2013, pp. 195–206.

[10] H. Madsen, G. Albeanu, B. Burtschy, and F. Popentiu-Vladicescu, "Reliability in the utility computing era: Towards reliable fog computing," in *Systems, Signals and Image Processing (IWSSIP), 2013 20th International Conference on*, July 2013, pp. 43–46.

[11] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, "Mobile fog: A programming model for large-scale applications on the internet of things," in *Proceedings of the Second ACM SIGCOMM Workshop on Mobile Cloud Computing*, ser. MCC'13. ACM, 2013, pp. 15–20.

[12] T. Nishio, R. Shinkuma, T. Takahashi, and N. B. Mandayam, "Service-oriented heterogeneous resource sharing for optimizing service latency in mobile cloud," in *Proceedings of the First International Workshop on Mobile Cloud Computing and Networking*, ser. MobileCloud'13. ACM, 2013, pp. 19–26.

[13] B. Ottenwalder, B. Koldehofe, K. Rothermel, and U. Ramachandran, "Migcep: Operator migration for mobility driven distributed complex event processing," in *Proceedings of the 7th ACM International Conference on Distributed Event-based Systems*, ser. DEBS'13. ACM, 2013, pp. 183–194.

[14] J. Zhu, D. Chan, M. Prabhu, P. Natarajan, H. Hu, and F. Bonomi, "Improving web sites performance using edge servers in fog computing architecture," in *Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on*, March 2013, pp. 320–323.

[15] BETaaS, "Building the environment for the things as a service," BETaaS, Tech. Rep., Nov. 2012.

[16] S. Maharjan, Q. Zhu, Y. Zhang, S. Gjessing, and T. Basar, "Dependable demand response management in the smart grid: A stackelberg game approach," *Smart Grid, IEEE Transactions on*, vol. 4, no. 1, pp. 120–132, March 2013.

[17] D. Korzhyk, V. Conitzer, and R. Parr, "Solving stackelberg games with uncertain observability," in *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 3*, ser. AAMAS '11, 2011, pp. 1013–1020.

[18] Z. Fadlullah, D. Quan, N. Kato, and I. Stojmenovic, "Gtes: An optimized game-theoretic demand-side management scheme for smart grid," *Systems Journal, IEEE*, vol. 8, no. 2, pp. 588–597, June 2014.

[19] T. Luo, H.-P. Tan, and T. Quek, "Sensor openflow: Enabling software-defined wireless sensor networks," *Communications Letters, IEEE*, vol. 16, no. 11, pp. 1896–1899, Nov. 2012.

[20] Y. Daraghmi, C.-W. Yi, and I. Stojmenovic, "Forwarding methods in data dissemination and routing protocols for vehicular ad hoc networks," *Network, IEEE*, vol. 27, no. 6, pp. 74–79, November 2013.

[21] B. Zhou, J. Cao, X. Zeng, and H. Wu, "Adaptive traffic light control in wireless sensor network-based intelligent transportation system," in *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, Sept 2010, pp. 1–5.

[22] B. Zhou, J. Cao, and H. Wu, "Adaptive traffic light control of multiple intersections in wsn-based its," in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, May 2011, pp. 1–5.

[23] C. Li and S. Shimamoto, "An open traffic light control model for reducing vehicles co2 emissions based on etc vehicles," *Vehicular Technology, IEEE Transactions on*, vol. 61, no. 1, pp. 97–110, Jan 2012.

[24] W. Wang and Z. Lu, "Survey cyber security in the smart grid: Survey and challenges," *Comput. Netw.*, vol. 57, no. 5, pp. 1344–1371, Apr. 2013.

[25] R. Lu, X. Li, X. Liang, X. Shen, and X. Lin, "Grs: The green, reliability, and security of emerging machine to machine communications," *Communications Magazine, IEEE*, vol. 49, no. 4, pp. 28–35, April 2011.

[26] Y. W. Law, M. Palaniswami, G. Kounga, and A. Lo, "Wake: Key management scheme for wide-area measurement systems in smart grid," *Communications Magazine, IEEE*, vol. 51, no. 1, pp. 34–41, January 2013.

[27] Z. Fadlullah, M. Fouda, N. Kato, A. Takeuchi, N. Iwasaki, and Y. Nozaki, "Toward intelligent machine-to-machine communications in smart grid," *Communications Magazine, IEEE*, vol. 49, no. 4, pp. 60–65, April 2011.

[28] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in cloud," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 42–57, 2013.

[29] J. Valenzuela, J. Wang, and N. Bissinger, "Real-time intrusion detection in power system operations," *Power Systems, IEEE Transactions on*, vol. 28, no. 2, pp. 1052–1062, May 2013.

[30] L. Zhang, W. Jia, S. Wen, and D. Yao, "A man-in-the-middle attack on 3g-wlan interworking," in *Communications and Mobile Computing (CMC), International Conference on*, vol. 1, April 2010, pp. 121–125.

[31] Broadcom bcm 5354. [Online]. Available: http://www.broadcom.com/products/Wireless-LAN/802.11-Wireless-LAN-Solutions/BCM5354

[32] Wikipedia. (2014) Hooking, what is hooking? [Online]. Available: http://en.wikipedia.org/wiki/Hooking

[33] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, "Eppa: An efficient and privacy-preserving aggregation scheme for secure smart grid communications," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 9, pp. 1621–1631, Sept 2012.