Vehicular Fog Computing: Architecture, Use Case, and Security and Forensic Challenges

Cheng Huang, Rongxing Lu, and Kim-Kwang Raymond Choo

The authors formalize the vehicular fog computing architecture and present a typical use case in vehicular fog computing. They discuss several key security and forensic challenges and potential solutions.

ABSTRACT

Vehicular fog computing extends the fog computing paradigm to conventional vehicular networks. This allows us to support more ubiquitous vehicles, achieve better communication efficiency, and address limitations in conventional vehicular networks in terms of latency, location awareness, and real-time response (typically required in smart traffic control, driving safety applications, entertainment services, and other applications). Such requirements are particularly important in adversarial environments (e.g., urban warfare and battlefields in the Internet of Battlefield Things involving military vehicles). However, there is no one widely accepted definition for vehicular fog computing and use cases. Thus, in this article, we formalize the vehicular fog computing architecture and present a typical use case in vehicular fog computing. Then we discuss several key security and forensic challenges and potential solutions.

INTRODUCTION

An observation on the Internet of Things (IoT) trend in a 2017 Gartner report [1] is the movement away from cloud-, Thing-, and gateway-centric IoT implementations to the edge (also referred to as fog computing or edge computing in the literature). In such an implementation, the bulk of the application logic, data storage, and analytics are placed on the actual device instead of a cloud or gateway server. This can result in significant bandwidth saving for the heterogeneous communication network.

One potential application of fog computing is in vehicle-based settings, such as the integration of fog computing with conventional vehicle ad hoc networks (VANET) to form the Internet of Vehicles (IoV) or vehicular fog computing. In the latter architecture, vehicles are regarded as intelligent devices that are mobile and equipped with multiple sensors, and have the computational/communication capability to gather useful traffic information. The information is gathered not only from the intra-vehicle sensors but also from the environment external to the vehicle(s). Fog nodes can be deployed at the edge of vehicular networks to efficiently and effectively collect, process, organize, and store traffic data in real time. When acquiring and processing a large amount of data from urban/highway areas via smart vehicles, vehicular fog computing architecture can facilitate or provide a wide range of vehicle-based services to the driver and passengers, such as smart traffic control, road safety improvement, and entertainment services. Similarly, there are potential applications in Internet of Battlefield Things deployment.

Fog computing, especially vehicular fog computing, is still in its early stage, with many unresolved and under-explored technical and operational challenges, ranging from architecture to clear use cases to security issues and so on. There has been interest in fog computing not only from academia, but also from the industry such as the establishment of the OpenFog Consortium [2]. Vehicular fog computing is one area that is relatively under-studied, despite the increasing trend in smart vehicles in practice. For example, in a recent report on connected vehicles by the IHS automotive company [3], it is estimated that there will be 152 million actively connected cars on the road by 2020, and an average car will produce up to 30 TB of data each day. This will result in a significant increase in bandwidth consumption and competition, in the sense that a connected vehicle would need to compete against other devices for finite bandwidth.

One potential solution to reduce the communication overhead is to have the server be geographically closer to the vehicle to serve the vehicle-based applications' demands in real time. This will require a significant investment in the underpinning infrastructure. However, to ensure optimal quality of protection (QoP) and quality of service (QoS), we need to strike a balance between performance, security, and privacy requirements.

In this article, we discuss the architecture, use cases, and security issues in this emerging vehicular fog computing paradigm. In the next section, we present a high-level overview of vehicular fog computing architecture and describe its benefits.

VEHICULAR FOG COMPUTING ARCHITECTURE: OVERVIEW

SYSTEM ARCHITECTURE

A high-level architecture of vehicular fog computing is presented in Fig. 1, which comprises three types of entities, namely smart vehicles as the data generation layer, roadside units/fog nodes as the fog layer, and cloud servers as the cloud layer.

Cheng Huang is with the University of Waterloo; Rongxing Lu is with the University of New Brunswick; Kim-Kwang Raymond Choo is with the University of Texas at San Antonio as well as the University of South Australia.

Digital Object Identifier: 10.1109/MCOM.2017.1700322

0163-6804/17/$25.00 © 2017 IEEE

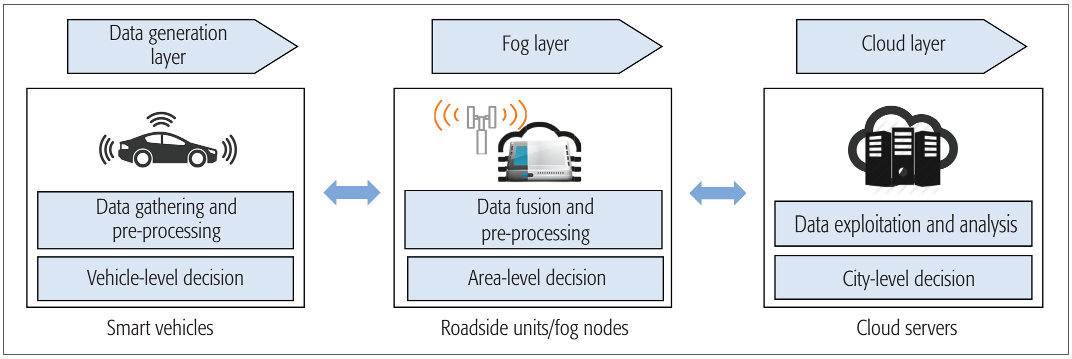

Figure 1. Architecture of vehicular fog computing.

Application type	Service	Description
Traffic control	Smart navigation	Plan optimal routes for smart vehicles
	Smart traffic lights	Schedule traffic lights of each intersection in the city to control traffic flows
Driving safety	Road condition detection	Detect environment information of smart vehicles and make adjustments accordingly
	Emergency warning	Broadcast emergency warning information to nearby smart vehicles, such as car accidents and work zones
Entertainment	Commercial advertisement	Publish advertisements of public interest (e.g., Amber alerts) to nearby smart vehicles
	Multimedia	Provide multimedia services for smart vehicles, such as music and video

Table 1. Application examples of vehicular fog computing.

Smart Vehicles: Smart vehicles play an important role as the key data generator in a vehicular fog computing system, due to their real-time computing, sensing (e.g., cameras, radars and GPS), communication, and storage capabilities. The amount of data collected by the various sensors in a smart vehicle has been estimated to be around 25 GB/h in a single day (e.g., 20–60 MB/s for cameras, 10 kB/s for radar, and 50 kB/s for GPS). Some of these data can be processed by the smart vehicle itself, in order to inform real-time decision making (i.e., vehicle-level decision), while other data will be shared and uploaded to the fog nodes for analysis and used for other purposes (e.g., traffic and infrastructure planning, as well as surveillance).

Roadside Units/Fog Nodes: Roadside units, generally deployed in different areas of a city, can easily be upgraded to act as fog nodes. This will allow the collection of data sent by smart vehicles, processing of the collected data, and reporting of the (processed) data to the cloud servers. These units/nodes also act as the middleware/intermediate devices on the function of a connecting link between the cloud servers and the smart vehicles in a vehicular fog computing system. Unlike existing vehicular networks, these units/nodes will have more functions and provide more diverse services for smart vehicles, such as navigation, video streaming, and smart traffic lights. In other words, these units/nodes are not just relays or broadcasters; they also process data, store data, and make decisions as a fog layer. (i.e., area-level decisions).

Cloud Servers: Cloud servers provide city-level monitoring and centralized control from a remote location. These servers will obtain the data uploaded by the fog nodes while performing computationally intensive analytics to make optimal decisions from a holistic perspective (e.g., city-level decision). For instance, they will monitor, manage, and control the city's road traffic infrastructures to achieve optimal city-level traffic control.

Potential Benefits

The vehicular fog computing architecture, if implemented correctly, can deliver wide-ranging benefits such as those shown in Table 1. Although many current and fast-developing vehicular fog computing systems may have unique features, most vehicular fog computing systems are generally organized in an architecture similar to that shown in Fig. 1, with the following common characteristics: The fog nodes are extensions of the cloud servers from remote areas to the edge in order to offer more efficient and effective services. This will allow vehicle-based applications to benefit in terms of response time, communication, and storage. These properties are particularly important in an adversarial setting (e.g., Internet of Battlefield Things involving military vehicles).

Response time: Most vehicular applications require real-time response, especially for traffic control and safety enhancement applications. However, conventional vehicular cloud computing architecture is not designed to meet this low-latency requirement, since data collected from smart vehicles will be processed remotely instead of locally. Due to the transmission delay and any potential connectivity issues (e.g., out of range), the average response time for cloud-based applications and locally processed applications will likely be more than a second and under 10 ms, respectively. Hence, fog nodes in a vehicular fog computing system, located in proximity to smart vehicles, can significantly reduce the response time for vehicular applications.

Communication: In the foreseeable future, the number of smart vehicles (including smart military vehicles) is likely to increase and perhaps become the norm. Thus, it is likely that the amount of data generated and transmitted by such vehicles will increase exponentially at a high frequency (similar to the current big data trend). In conventional vehicular cloud computing scenarios, raw data

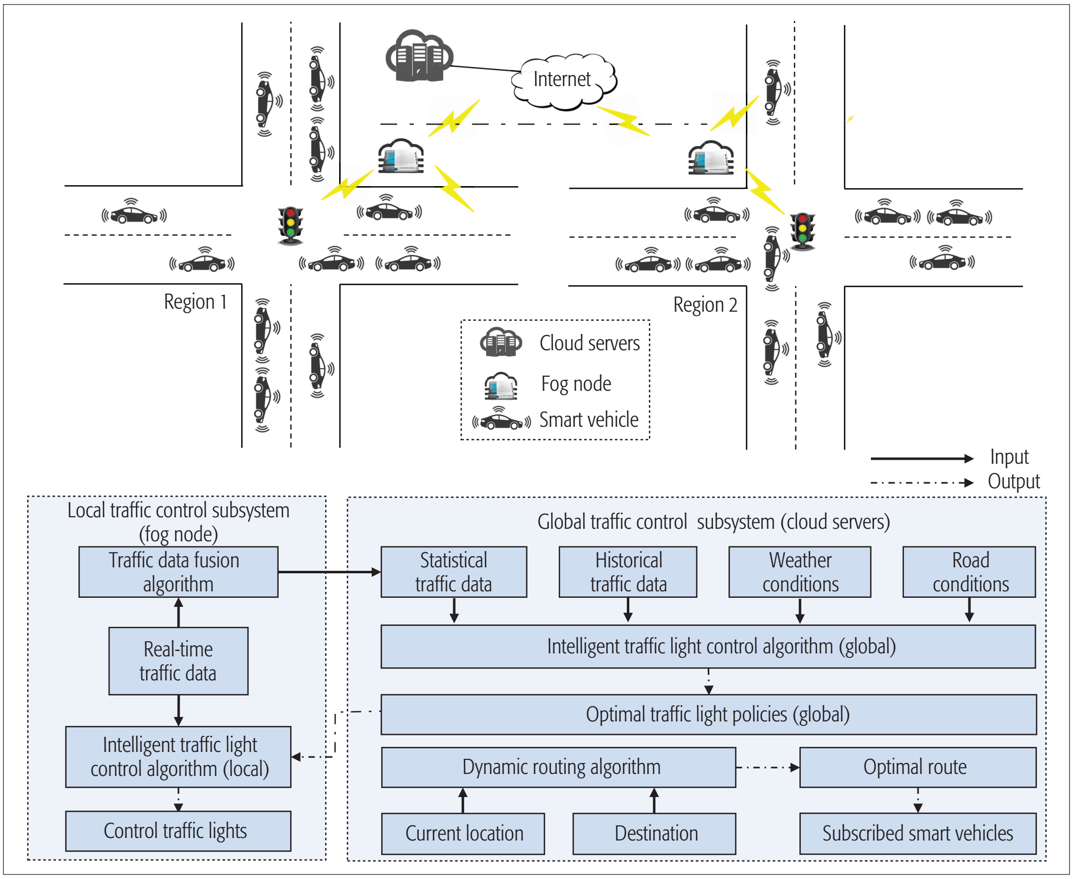
Internet
Region 1
Region 2
Cloud servers
Fog node
Smart vehicle
Input
Output
Local traffic control subsystem (fog node)
Traffic data fusion algorithm
Real-time traffic data
Intelligent traffic light control algorithm (local)
Control traffic lights
Global traffic control subsystem (cloud servers)
Statistical traffic data
Historical traffic data
Weather conditions
Road conditions
Intelligent traffic light control algorithm (global)
Optimal traffic light policies (global)
Dynamic routing algorithm
Optimal route
Current location
Destination
Subscribed smart vehicles

Figure 2. A fog-assisted traffic control system: an overview.

If the data is too large and frequent, communications will be a bottleneck for most vehicular applications. Therefore, the fog nodes in a vehicular fog computing system can alleviate such limitations by pre-processing the collected data so that the data can be aggregated/filtered prior to uploading. This allows data volume and frequency to be reduced.

is directly uploaded to the cloud servers for subsequent processing. Despite potential advances in communication technologies, the bandwidth required for efficient transmission of such a big volume of data is not guaranteed due to a wide range of logistical, political, and geographical factors, particularly in a conflict zone. If the data is too large and frequent, communications will be a bottleneck for most vehicular applications. Therefore, the fog nodes in a vehicular fog computing system can alleviate such limitations by pre-processing the collected data so that the data can be aggregated/filtered prior to uploading. This allows data volume and frequency to be reduced.

Storage: For conventional vehicular cloud computing architecture, almost all application data will be stored in the remote cloud servers. This may not be practical due to the changing nature of vehicular applications and collected data. For example, data and vehicular applications are increasingly becoming location-aware. Thus, the ability to access stored data in real time (e.g., data stored in decentralized location-aware fog nodes) will reduce the storage burden on the remote cloud servers.

A Potential Use Case: A Fog-Assisted Traffic Control System

We use a fog-assisted traffic control system as a use case to explain the vehicular fog computing architecture. A fog-assisted traffic management system is designed to deliver benefits such as reducing road traffic congestion and car accidents. A typical implementation will consist of two subsystems: one responsible for the local area and one responsible for the global area (a.k.a city-wide area).

Local Traffic Control Subsystem

The local traffic control subsystem is tasked with monitoring and managing traffic flow in a local area. A fog node's communication range covers a region of the city and can involve several intersections, as shown in Fig. 2. If a smart vehicle is physically located within the communication range of a fog node, it can send and receive messages to and from the fog nodes. Specifically, when a smart vehicle drives into a region within the coverage of a fog node, it will frequently report its current location, speed, weather conditions, and road conditions to the specific node until it leaves this region.

Based on the data received from the smart vehicles, the local traffic control subsystem can perform the following. For example, in the first phase, the fog node will monitor and control the local traffic flow by scheduling the traffic light at each intersection for the smart vehicles in its region. An intelligent traffic light control algorithm (local) is implemented at the fog node. By using each vehicle's reported data as the input, the fog node calculates the traffic information such as road segment occupancy, and then runs the intelligent traffic light control algorithm to avoid traffic build-up by managing the red and green phase proportion of each traffic light. This phase should be operated in real time and have low latency. When smart vehicles are being operated at high speed and are constantly on the move, the

A number of traffic management algorithms, which may be deployed in a fog-assisted traffic control system, have been proposed in recent years. These include the traffic scheduling algorithms, ITLC and ATL, for controlling the traffic lights of an isolated traffic intersection and the entire road network [4], and a distributed real-time routing algorithm to avoid traffic congestion.

response time for controlling the traffic flow will be significantly impacted by the traveling speed of these vehicles. It would not be useful to update traffic lights for vehicles leaving this intersection. During the second phase, the fog node will pre-process and aggregate these received data as the statistical traffic information, and report such information to the cloud servers. Concretely, the data reported to the cloud servers will not be the same as each vehicle's data has a different data format (e.g., location and speed). Using a traffic data fusion algorithm, the fog node will integrate the received data as traffic volumes (e.g., number of vehicles, average speed of the vehicles, and average waiting time of the vehicles at each intersection) and then report the output of this algorithm to the cloud servers.

Global Traffic Management Subsystem

The global traffic control subsystem is responsible for controlling and managing the traffic flow from a city-wide perspective. As shown in Fig. 2, the cloud servers are remote, gathering data sent by the fog nodes and performing (big) data analytics to mine the traffic information. Traffic controlling algorithms used in the cloud servers include an intelligent traffic light control algorithm (global) and a dynamic routing algorithm.

The intelligent traffic light control algorithm in the cloud servers is more complicated compared to that at the fog nodes. The algorithm in the cloud servers is intended to predict and adjust traffic control systems (e.g., traffic lights) by considering not only real-time traffic volume but also other relative information (e.g., historical traffic records, weather conditions, and road conditions). Thus, this is a more time-consuming exercise at the cloud server. However, the response time will not be a critical metric for this algorithm since the traffic volume of the whole city and weather conditions are unlikely to vary significantly over a short period. The mining results of this algorithm will be the optimal traffic light policies, and the cloud servers will distribute these policies as the feedback to all fog nodes in the city. Additionally, the cloud servers can provide a navigation service for some smart vehicles to help control the traffic flow. A dynamic routing algorithm is also required. Using a participating smart vehicle's current location and destination as the input, the algorithm will output the smart vehicle's optimal route by predicting and simulating the traffic conditions.

A number of traffic management algorithms, which may be deployed in a fog-assisted traffic control system, have been proposed in recent years. These include the traffic scheduling algorithms ITLC and ATL for controlling the traffic lights of an isolated traffic intersection and the entire road network [4], and a distributed real-time routing algorithm to avoid traffic congestion [5].

Security and Forensic Challenges in Vehicular Fog Computing

Research in understanding and mitigating security risks in vehicular fog computing is still in its infancy. Existing security research mainly focuses on the identification of potential attacks, threats, and vulnerabilities of fog-assisted vehicular applications. Generally speaking, attacks in vehicular fog computing can be categorized into passive and active attacks, and there are two kinds of attackers: an external attacker and an insider. An external attacker is not equipped with key materials in a vehicular fog computing system, while an insider attack is one originating from compromised smart vehicles, fog nodes, or cloud nodes that hold the key materials.

A passive attack does not destroy the functionality of a vehicular fog computing system but attempts to disclose private information (e.g., eavesdropping). Passive attacks by an internal attacker are generally more damaging than those conducted by an external attacker, particularly in an adversarial setting, since the insider is more likely to be able to circumvent existing security controls. An active attack is an attempt to deliberately disrupt the operations of a vehicular fog computing system (e.g., distributed denial of service [DDoS] attacks, modifying data of smart vehicles or the decision data of fog nodes and cloud servers, and data exfiltration). An active attack is easy to detect as long it has an enormous impact on the system. However, sometimes the attacker is prone to perform an inconspicuous attack (insider or outsider) during an ultra short period, which is hard to find.

Security and Forensic Requirements

A secure vehicular fog computing implementation should provide the following baseline security and forensic properties.

Confidentiality: Confidentiality ensures that any unauthorized access attempts to either data-at-rest and data-in-transit in a vehicular fog computing system will be detected and prevented.

Integrity: Integrity ensures that any unauthorized attempts to modify data being transmitted or stored will be detected. In a vehicular fog computing system, it is critical to meet the integrity requirement since unauthorized modification may result in serious and/or fatal consequences, especially in life-critical vehicular application contexts such as a traffic control system.

Authentication: Authentication ensures that any two communication entities are able to corroborate the data in transmission.

Access control: Access control is designed to limit fog node access only to authorized entities (e.g., participating and non-compromised smart vehicles to gain access to the fog nodes for some subscribed services like navigation and entertainment).

Non-repudiation: Non-repudiation ensures that any entity in the system is not able to deny a previous action (e.g., sending data).

Availability: Availability ensures that whenever a vehicular application attempts to access the fog nodes or cloud servers, they are always available.

Reliability: Reliability ensures that the data collected from smart vehicles and fog nodes has not been modified or fabricated.

Forensics: Forensics ensures that the capability to identify, collect, and analyze data from smart vehicles, fog nodes, and the underlying infrastructure for tracing and identifying the malicious sources.

In general, most of the above-mentioned security requirements can be achieved partly using cryptographic techniques. For example, fully homomorphic encryption primitives can be employed to achieve confidentiality and functionality at the same time. However, most security mechanisms only effectively defend against passive attacks, and there is no foolproof security solution. Once one or more fog nodes have been compromised [6], for example, to launch attacks within a fog-assisted traffic control system, more sophisticated security mechanisms will be necessary to detect and deter such attacks.

An Example: A Compromise Attack on Fog-Assisted Traffic Control System

In a fog-assisted traffic control, fog nodes are regularly deployed at public roadsides without any physical isolation due to their location-aware nature. Hence, such nodes are more vulnerable to physical compromise attacks compared to cloud servers that are generally protected physically. With public access to fog nodes, attackers can attempt a variety of attacks, such as false data injection, black/gray hole attack, and on-off attack. A node compromise attack can be broadly generalized into the following three stages:

- The attacker gains administrative access to a fog node by physically capturing and compromising the particular node.
- The attacker alters the functions of the compromised fog node and redeploys it back to the system.
- The attacker manages the compromised fog nodes and launches different attacks to disrupt the process of traffic control.

Two types of attackers are considered after the fog nodes are compromised. Specifically, an attacker who seeks to degrade the performance of the system (hereafter referred to as an evil attacker) and an attacker who seeks to benefit himself/herself in the system (i.e., a selfish attacker). Specifically, the evil attacker will minimize the road traffic network's utility to maximize the total travel time in this network (i.e., average travel time increases). Suppose that T' denotes the total travel time computed from a traffic model for the attacked network, while T is the total travel time computed from the same traffic model for a normal network. The goal of this attacker is to find the maximum $T' - T$ using the compromised fog nodes. In contrast, the selfish attacker will maximize his/her own interests by changing the traffic flow of the network. That is, the attacker will minimize travel time between locations A and B by amending the strategies of compromised fog nodes since he/she would likely travel from A to B. Let $T'_{A,B}$ denote the travel time computed from a traffic model for the target network, while $T_{A,B}$ is the travel time computed from the same traffic model for the normal network. The purpose of this attacker is to locate the maximum $T_{A,B} - T'_{A,B}$.

However, in comparison to the evil attackers, selfish attackers are more difficult to detect as attack time may be extremely short, and the compromised fog node is most likely to behave normally outside the attack. In addition, in an attack performed by a selfish attacker, modifications to the system are likely to be minimal (e.g., only sufficient to reduce the waiting time at an intersection the attacker is approaching). For fog-assisted traffic control, traffic light control in each intersection is determined by the fog node itself, and a minor modification in the traffic light's strategy will not result in a significant influence or impact on the entire system. Thus, we introduce two security mechanisms as potential countermeasures for selfish attackers.

In a fog-assisted traffic control, fog nodes are regularly deployed on public roadside without any physical isolation due to its location-awareness nature. Hence, such nodes are more vulnerable to physical compromise attacks, as compared to cloud servers that are generally protected physically.

Potential Countermeasures for Selfish Attacks

To deal with selfish attackers, it is important to identify and detect compromised fog nodes in the system. It is not practical to physically check all the fog nodes deployed in the system (e.g., the state of Texas) for compromise, and real-time detection is not realistic. We also need to ensure that the security solutions do not significantly impact functionality and performance. Hence, we posit the use of an evidence-based digital forensic approach and a traffic-based analysis approach based on real-time and historical traffic data.

Evidence-based digital forensic approach: A potentially effective way to locate compromised fog nodes is to forensically analyze artifacts from smart vehicles and fog nodes that have been or are believed to be compromised. For example, smart vehicles can directly communicate with the fog node and make judgments based on the behavior of each fog node. If any smart vehicle or fog node flags another node or vehicle as suspicious or exhibits abnormal behavior (e.g., an usually short/long waiting time at a particular intersection), a forensic investigation into these vehicles or nodes can be conducted. Based on the findings of the forensic investigation, the vehicles may be restricted from further interacting with the system or the nodes replaced. Depending on the findings and the context, we could monitor the behavior of the vehicle or node before making the final determination (e.g., compromised, malicious, or false alarm). To generate evidence, an authentication mechanism is needed between the smart vehicles and the fog nodes, and each fog node needs to periodically broadcast its digitally signed identity/signature that can be verified by the smart vehicles.

To investigate the utility of this approach, we simulate the traffic condition based on SUMO [7] and OpenStreetMap [8]. As shown in Fig. 3a, we choose two random junctions (one is compromised and the other is normal) in the city of Waterloo, Canada. We then generate different traffic rates (i.e., 1 vehicle/s, 1 vehicle/5 s, and 1 vehicle/30 s). The routes for these smart vehicles are created randomly during one hour. When smart vehicles pass through a junction, they have a probability p_d to accurately identify a compromised fog node and a probability p_e to mistakenly regard a normal fog node as a compromised one. Formally, the probability $1 - p_d$ indicates the false negative rate, while the probability p_e indicates the false positive rate. We define p_d as 0.8, 0.4, and 0.2 and p_e as 0.1, 0.2, and 0.4, and the numerical results of the simulation are shown in Fig. 3b. The number of reports from the compromised fog node is slightly more than that of those of the normal fog node. Since the smart vehicles' diverse abilities (low p_d or high p_e) make a lot of "noises," it is a challenge to identify the compromised fog node from the received reports.

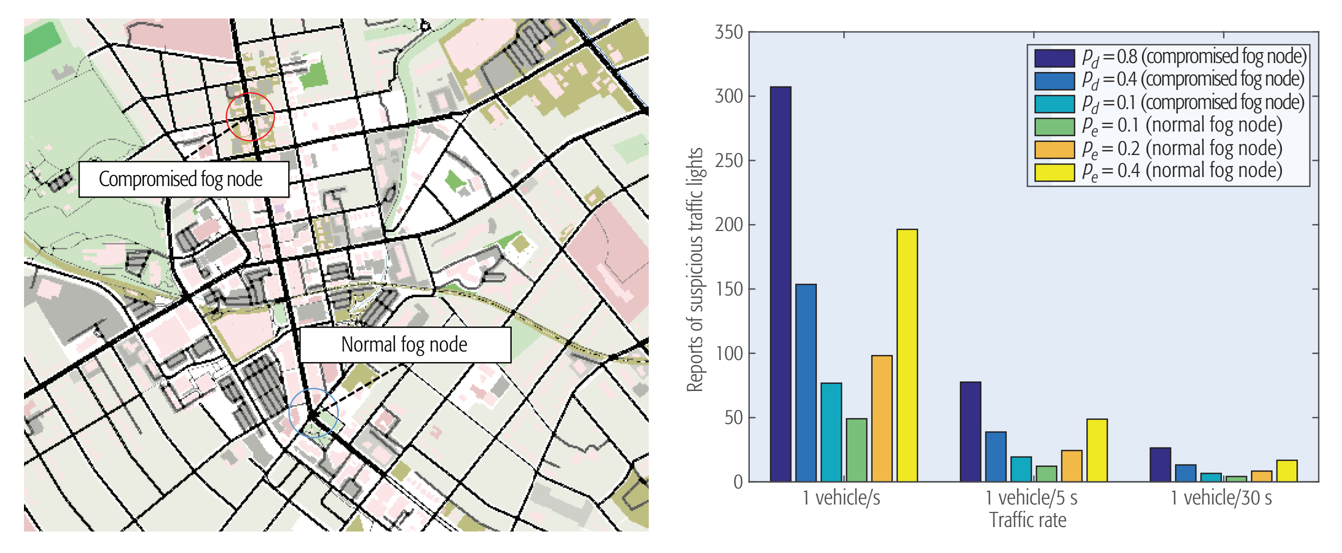

Figure 3. Simulation of evidence-based digital forensic approach: a) Waterloo map downloaded from OpenStreetMap; b) numerical results of simulation based on different settings.

Therefore, a reputation system needs to be in place to prevent "badmouthing" attacks from colluding smart vehicles. Finally, the suspicious vehicles or fog nodes should be manually examined to confirm the abnormal behaviors from the smart vehicles, and the reputation of smart vehicles should be updated based on the feedback.

Traffic-based analysis approach: Another possible practical approach is based on big data analytics and deep learning algorithms. The cloud servers have access to historical/archival traffic data reported by each fog node. No matter how sophisticated an attacker is, data from compromised fog nodes will display/have some (minor) different behavior and characteristics than data from normal nodes. For example, as long as normal fog nodes are deployed around the compromised ones, which modify the traffic lights without considering logical traffic flow, the traffic flow at the normal fog nodes will be irregular to some extent. The association between the fog nodes can also be mined to identify the compromised fog nodes based on the real-time traffic flow changes. A simple example is the traffic prediction model. The current traffic data is utilized to forecast the future traffic flow, which is compared to the real traffic flow at that time to locate fog nodes that may be compromised. In addition, the historical traffic data can also be a baseline reference to identify compromised fog nodes. The relationship between the present traffic flow and the historical traffic flow can also help to distinguish abnormal traffic data and then can be deeply mined to identify compromised fog nodes. That is, the traffic features extracted from the statistical traffic information, such as the average velocity and number of vehicles in a junction, can be analyzed using data clustering (e.g., *K*-means and outlier detection) and classification methods (e.g., naive Bayes and support vector machine).

In summary, a node compromise attack is likely to be mitigated by an evidence-based digital forensic approach and/or a traffic-based analysis approach based on real-time and historical traffic data. Specifically, in the case of established authentication schemes, certificate-based and identity-based encryption/signatures can easily be applied and implemented, despite having limitations such as authentication efficiency and revocation costs [9]. In terms of efficiency, the group authentication scheme in [10] is a potential solution, but the use of such a scheme may complicate forensic investigations. A recently proposed reputation-based system [11] for vehicular networks, based on the Dirichlet distribution, can potentially be extended to help detect malicious vehicles and compromised fog nodes. One could also combine reputation-based systems and truth discovery approaches, such as majority voting and weighted averaging, for enhanced accuracy. Although there are a number of traffic monitoring and prediction systems in the literature (e.g., [12]), how to detect the abnormal traffic flow and finally detect the compromised fog nodes based on a large volume of traffic data still remains a research and an operational challenge.

Conclusion

In this article, we present an architecture for vehicular fog computing, and discuss the potential benefits, security, and forensic challenges and mitigation strategies using the fog-assisted traffic control system as a use case.

To keep pace with technological advances and the changing nature and needs of our society, there are a number of research opportunities in this space. One such challenge is to effectively strike a balance between functionality, security, and privacy in specific vehicular application contexts (e.g., data privacy [13] and location privacy [14]). Extending the work of Ab Rahman, Glisson, Yang, and Choo [15]}, how to best integrate forensics techniques and best practices into the design and development of a vehicular fog computing system so that it is forensically ready/friendly is another potential research topic. Having a forensically ready/friendly vehicular fog computing system will allow the real-time identification, collection, and analysis of data that can be used to inform mitigation strategies.

References

[1] B. Menezes and S. B. Alaybeyi, “Iot Components Will Require Changes to Enterprise Networks,” Gartner Report, vol. G00316844, 2017, pp. 1–14.

[2] O. Consortium, “Openfog Architecture Overview”; https://www.openfogconsortium.org/wp-content/uploads/OpenFog-Architecture-Overview-WP-2-2016.pdf, 2016, accessed 3 Mar. 2017.

[3] S. Institute, “The Connected Vehicle: Big Data, Big Opportunities”; https://www.sas.com/content/dam/SAS/en_us/doc/whitepaper1/connected-vehicle-107832.pdf, 2016, accessed 11 Mar. 2017.

[4] M. B. Younes and A. Boukerche, “Intelligent Traffic Light Controlling Algorithms Using Vehicular Networks,” *IEEE Trans. Vehic. Tech.*, vol. 65, no. 8, 2016, pp. 5887–99.

[5] S. J. Pan, I. S. Popa, and C. Borcea, “DIVERT: A Distributed Vehicular Traffic Re-Routing System for Congestion Avoidance,” *IEEE Trans. Mob. Comp.*, vol. 16, no. 1, 2017, pp. 58–72.

[6] A. Laszka *et al.*, “Vulnerability of Transportation Networks to Traffic-Signal Tampering,” *7th ACM/IEEE Int’l. Conf. Cyber-Physical Systems 2016*, Vienna, Austria, Apr. 11–14, 2016, pp. 16:1–16:10.

[7] D. Krajzewicz *et al.*, “Recent Development and Applications of SUMO — Simulation of Urban MObility,” *Int’l. J. Advances in Systems and Measurements*, vol. 5, no. 3&4, Dec. 2012, pp. 128–38.

[8] OpenStreetMap, “Waterloo Map Downloaded from Openstreetmap Dataset”; https://www.openstreetmap.org/#map=13/43.4572/-80.5035, 2017, accessed 4 Apr. 2017.

[9] F. Qu *et al.*, “A Security and Privacy Review of VANETS,” *IEEE Trans. Intelligent Transportation Systems*, vol. 16, no. 6, 2015, pp. 2985–96.

[10] C. Lai *et al.*, “GLARM: Group-Based Lightweight Authentication Scheme for Resource-Constrained Machine to Machine Communications,” *Computer Networks*, vol. 99, 201, pp. 66–816.

[11] H. Hu *et al.*, “Tripsense: A Trust-Based Vehicular Platoon Crowdsensing Scheme with Privacy Preservation in VANETS,” *Sensors*, vol. 16, no. 6, 2016, p. 803.

[12] R. Lu *et al.*, “A Lightweight Conditional Privacy-Preservation Protocol for Vehicular Trafficmonitoring Systems,” *IEEE Intelligent Systems*, vol. 28, no. 3, 2013, pp. 62–65.

[13] R. Lu *et al.*, “A lightweight Privacy-Preserving Data Aggregation Scheme for Fog Computing-Enhanced IoT,” *IEEE Access*, vol. 5, 2017, pp. 3302–12.

[14] H. Zhu *et al.*, “An Efficient Privacy-Preserving Location-Based Services Query Scheme in Outsourced Cloud,” *IEEE Trans. Vehic. Tech.*, vol. 65, no. 9, 2016, pp. 7729–39.

[15] N. H. Ab Rahman *et al.*, “Forensic-by-Design Framework for Cyber-Physical Cloud Systems,” *IEEE Cloud Computing*, vol. 3, no. 1, 2016, pp. 50–59.

Biographies

Cheng Huang [S’15] (c225huan@uwaterloo.ca) received his B.Eng. and M.Eng. from Xidian University, China, in 2013 and 2016, respectively, and was a project officer with the INFINITUS laboratory at the School of Electrical and Electronic Engineering, Nanyang Technological University until July 2016. Since September 2016, he has been a Ph.D. student with the Department of Electrical and Computer Engineering, University of Waterloo, Ontario, Canada. His research interests are in the areas of applied cryptography, cyber security, and privacy.

Rongxing Lu [S’09-M’11-SM’15] (rlu1@unb.ca) is an assistant professor at the Faculty of Computer Science, University of New Brunswick. He was awarded the Governor General’s Gold Medal, Canada, in 2012, and the IEEE ComSoc Asia Pacific Outstanding Young Researcher Award in 2013. He is Secretary of IEEE ComSoc CIS-TC. His research interests include applied cryptography, privacy enhancing technologies, and IoT-big data security and privacy.

Kim-Kwang Raymond Choo [SM’15] (raymond.choo@fulbrightmail.org) holds the cloud technology endowed professorship at the University of Texas at San Antonio. His awards include the ESORICS 2015 Best Research Paper Award, the 2015 Winning Team of Germany’s University of Erlangen-Nuremberg Digital Forensics Research Challenge, the 2014 Australia New Zealand Policing Advisory Agency’s Highly Commended Award, the 2010 Australian Capital Territory Pearcey Award, a Fulbright Scholarship, a 2008 Australia Day Achievement Medallion, and the British Computer Society’s Wilkes Award. He is an Australian Computer Society fellow.

To keep pace with technological advances and the changing nature and needs of our society, there are a number of research opportunities in this space. One such challenge is to effectively strike a balance between functionality, security and privacy in specific vehicular application contexts.