Security Challenges in Vehicular Cloud Computing

Gongjun Yan, Ding Wen, Stephan Olariu, and Michele C. Weigle

***Abstract*—In a series of recent papers, Prof. Olariu and his co-workers have promoted the vision of vehicular clouds (VCs), a nontrivial extension, along several dimensions, of conventional cloud computing. In a VC, underutilized vehicular resources including computing power, storage, and Internet connectivity can be shared between drivers or rented out over the Internet to various customers. Clearly, if the VC concept is to see a wide adoption and to have significant societal impact, security and privacy issues need to be addressed. The main contribution of this work is to identify and analyze a number of security challenges and potential privacy threats in VCs. Although security issues have received attention in cloud computing and vehicular networks, we identify security challenges that are specific to VCs, e.g., challenges of authentication of high-mobility vehicles, scalability and single interface, tangled identities and locations, and the complexity of establishing trust relationships among multiple players caused by intermittent short-range communications. Additionally, we provide a security scheme that addresses several of the challenges discussed.**

***Index Terms*—Challenge analysis, cloud computing, privacy, security, vehicular cloud.**

I. Introduction

IN AN effort to help their vehicles compete in the marketplace, car and truck manufacturers are offering increasingly more potent onboard devices, including powerful computers, a large array of sensors, radar devices, cameras, and wireless transceivers. These devices cater to a set of customers that expect their vehicles to provide seamless extension of their home environment populated by sophisticated entertainment centers, access to Internet, and other similar wants and needs. Powerful onboard devices support new applications, including location-specific services, online gaming, and various forms of mobile infotainment [4].

In spite of the phenomenal growth of third-party applications catering to the driving public, it has been recently noticed that, most of the time, the huge array of onboard capabilities are chronically underutilized. In a series of recent papers [1]–[3], Olariu and his co-workers have put forth the vision of vehicular clouds (VCs), a nontrivial extension of conventional cloud computing, intended to harness the excess capabilities in our vehicles. Vehicles and roadside infrastructure with idle sophisticated onboard devices for long periods of time can be recruited to form a VC. A VC can be formed on-the-fly by dynamically integrating resources and collecting information. Vehicles can access the cloud and obtain, at the right time and the right place, all the needed resources and applications that they need or want.

Manuscript received September 21, 2011; revised March 20, 2012 and June 19, 2012; accepted July 21, 2012. This work was supported in part by Indiana University Kokomo under Grant 2263160, by the National Science Foundation (NSF) of China 11126333, and by NSF Grant CNS 0721586 and Grant CNS-1116238. The Associate Editor for this paper was L. Li.

G. Yan is with Indiana University Kokomo, Kokomo, IN 46904 USA (e-mail: goyan@iuk.edu).

D. Wen is with the Center for Military Computational Experiments and Parallel Systems Technology, National University of Defense Technology Changsha, Hunan 410073, China (e-mail: dingwen2010@gmail.com).

S. Olariu and M. C. Weigle are with Old Dominion University, Norfolk, VA 23529 USA (e-mail: olariu@cs.odu.edu; mweigle@cs.odu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2012.2211870

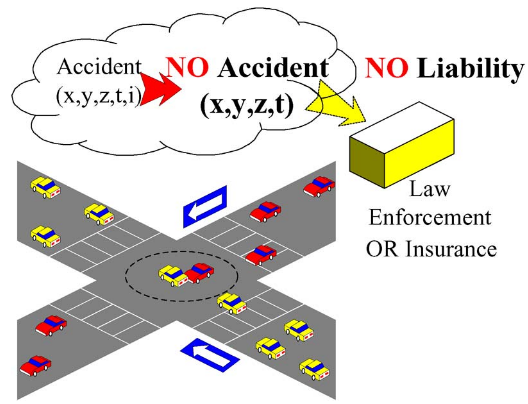

Fig. 1. Illustrating a security issue in VCs.

Obviously, security and privacy issues need to be addressed if the VC concept is to be widely adopted. Conventional networks attempt to prevent attackers from entering a system. However, in VC, all the users, including the attackers, are equal. The attackers and their targets may be physically colocated on one machine. The attackers can utilize system loopholes to reach their goals, such as obtaining confidential information and tampering with the integrity of information and the availability of resources. Fig. 1 shows one possible example of tampering with the integrity of information in the case of a road accident. Imagine that an accident has occurred at an intersection, and the accident will be reported to the VC. The driver liable for the accident can invade the VC and modify the accident record. Later, when the law enforcement or the vehicle insurance company query the accident, they cannot link the accident to the driver who caused it.

Superficially, the security issues encountered in VCs may look deceivingly similar to those experienced in other networks. However, a more careful analysis reveals that many of the classic security challenges are exacerbated by the characteristic features of VCs to the point where they can be construed as VC-specific. For example, the high mobility of vehicles is apt to cause significant challenges related to managing authentication, authorization, and accountability since the vehicles communicate through short-range dedicated short-range communications (DSRC) transceivers [5]. Vehicular mobility and tangled identities and locations also cause significant challenges of privacy [6]. Employing pseudonyms [7] is a common solution, but the high mobility makes the task of updating pseudonyms quite difficult.

The two main contributions of this work are to identify and analyze security challenges and privacy threats that are VC specific and to propose a reasonable security framework that addresses some of the VC challenges identified in this paper.

II. State of the Art

The security challenges in VC are a new, exciting, and unexplored topic. Vehicles will be autonomously pooled to create a cloud that can provide services to authorized users. This cloud can provide real-time services, such as mobile analytic laboratories, intelligent transportation systems, smart cities, and smart electric power grids. Vehicles will share the capability of computing power, Internet access, and storage to form conventional clouds. These researchers have only focused on providing a framework for VC computing, but as already mentioned, the issue of security and privacy has not yet been addressed in the literature. As pointed out by Hasan [8], cloud security becomes one of the major barriers of a widespread adoption of conventional cloud services. Extrapolating from the conclusions of [8], we anticipate that the same problems will be present in VCs.

Recently, vehicular ad hoc network (VANET) security and privacy have been addressed by a large number of papers. Yan *et al.* [9], [10] proposed active and passive location security algorithms. Radar can be employed as a "virtual eye," and onboard radar can detect the location of vehicles. Public Key Infrastructure (PKI) and digital signature-based methods have been well explored in VANETs [11]. A certificate authority (CA) generates public and private keys for nodes. The purpose of digital signature is to validate and authenticate the sender. The purpose of encryption is to disclose the content of messages only to entitled users. PKI is a method that is well suited for security purposes, particularly for roadside infrastructure. GeoEncrypt in VANETs has been proposed by Yan *et al.* [12]. Their idea is to use the geographic location of a vehicle to generate a secret key. Messages are encrypted with the secret key, and the encoded texts are sent to receiving vehicles. The receiving vehicles must be physically present in a certain geographic region specified by the sender to be able to decrypt the message.

Recently, some attention has been devoted to the general security problem in clouds, although not associated with vehicular networks [13]. The simple solution is to restrict access to the cloud hardware facilities. This can minimize risks from insiders [14]. Santos *et al.* [15] proposed a new platform to achieve trust in conventional clouds. A trust coordinator maintained by an external third party is imported to validate the entrusted cloud manager, which makes a set of virtual machines (VMs) such as Amazon's E2C (i.e., Infrastructure as a Service, IaaS) available to users. Garfinkel *et al.* [16] proposed a solution to prevent the owner of a physical host from accessing and interfering with the services on the host. Berger *et al.* [17] and Murray *et al.* [18] adopted a similar solution. When a VM boots up, system information such as the basic input output system (BIOS), system programs, and all the service applications is recorded, and a hash value is generated and transmitted to a third-party Trust Center. For every period of time, the system will collect system information of the BIOS, system programs, and all the service applications and transmit the hash value of system information to the third-party Trust Center. The Trust Center can evaluate the trust value of the cloud. Krautheim [19] also proposed a third party to share the responsibility of security in cloud computing between the service provider and client, decreasing the risk exposure to both. Jensen *et al.* [20] stated technical security issues of using cloud services on the Internet access. Wang *et al.* [21], [22] proposed public-key-based homomorphic authenticator and random masking to secure cloud data and preserve privacy of public cloud data. The bilinear aggregate signature has been extended to simultaneously audit multiple users. Ristenpart *et al.* [23] presented experiments of locating co-residence of other users in cloud VMs.

III. Vehicular Clouds: Paradigm Shift

A. Conceptual Overview

1) Cloud Computing: In recent years, cloud computing and its myriad applications that promise to change the way we think about computing and data storage have received a huge amount of attention. Cloud users do not need to install expensive hardware and software on their local machine. They can subscribe and use both hardware and software *as a service* when they want to use it. In addition, fees are charged based on the usage of the service. The users can access these services through Internet browsers, and no expensive client terminals are needed. Service providers can make good use of *excess* capabilities on the server side including processors, storage, and sensors that can be used to provide services to clients.

2) VANET: In VANETs, the vehicles communicate with each other and/or with the roadside infrastructure using the Federal Communications Commission-mandated DSRC [24], restricting the transmission range to 300–1000 m. There are two types of VANET networks: the zero-infrastructure and the infrastructure-based VANET. The zero-infrastructure VANET is created on-the-fly. There are many challenging security and privacy problems because no infrastructure is used for authentication and authorization. The infrastructure-based VANET can be formed based on the roadside infrastructure. The infrastructure can act as wireless access points for authentication and authorization purposes. By the same token, the vehicles can use the infrastructure to report events and to exchange information.

3) VCs: Similar to VANETs, there are two types of VCs. In the first type called *Infrastructure-based VC*, drivers will be able to access services by network communications involving the roadside infrastructure. In the second type called *Autonomous VC* (AVC) [2], vehicles can be organized on-the-fly to form VC in support of emergencies and other ad hoc events.

VCs provide services at three levels, i.e., application, platform, and infrastructure. Service providers use the levels differently based on what and how the services are offered. The fundamental level is called *Infrastructure as a Service* (IaaS), where infrastructure such as computing, storage, sensing, communicating devices, and software are created as VMs. The next level is *Platform as a Service* (PaaS), where components and services (such as httpd, ftpd, and email server) are provided and configured as a service. The top level is called *Software as a Service* (SaaS), where applications are provided in a "pay-as-you-go" fashion.

VCs provide a cost-efficient way to offer comprehensive services. For example, a cheaper vehicle with network access can access a VM with strong computation, communication, sensing capability, and large storage. Many applications such as traffic news, road conditions, or intelligent navigation systems can be provided by a VM [25].

B. Potential Applications of VC Computing

In this section, we review several possible applications of VCs.

1) *Vehicle maintenance:* Vehicles receive software updates from cloud whenever vehicle manufacturers upload a new version of software.
2) *Traffic management:* Drivers can receive traffic status reports (e.g., congestion) from VCs.
3) *Road condition sharing:* Road conditions such as flooding areas and black ice on the roadway can be shared in VCs. Drivers will be alerted if there are serious road conditions.
4) *Accident alerts at intersections:* Under demanding driving conditions such as fog, heavy storm, snow, and black ice, drivers can order this service to alert them of possible accidents at intersections. Infrastructure, e.g., a tall building, can include high-precision radar to detect car accidents. This infrastructure will cover the whole intersection and frequently scan the intersection. An intelligent algorithm will be applied to each scan result to predict the possibility of accidents.
5) *Safety applications:* Applications related to life-critical scenarios such as collision avoidance and adaptive cruise control require strong security protection, even from surrounding environmental security threats.
6) *Intelligent parking management:* Vehicles will be able to book a parking spot using the VC. All the parking information will be available on clouds without central control. Requests from different physical places can be transferred to the most desired parking lots.
7) *Planned evacuations:* In some disasters such as a hurricanes and tsunamis, VCs will be instrumental in organized evacuations.

IV. Analyzing Security in a Vehicular Cloud

In this section, we introduce a set of security analyses that are specially associated with VCs.

A. Security and Privacy Attacks in VC

1) Attacker Model: Traditional security systems are often designed to prevent attackers from entering the system. However, security systems in the VC have a much harder time keeping attackers at bay, because multiple service users with high mobility can share the same physical infrastructure. In the VC environment, an attacker can equally share the same physical machine/infrastructure as their targets, although both of them are assigned to different VMs. To this point, attackers can have more advantages than the attackers on traditional systems. In addition, the attackers are physically moving from place to place as vehicles are mobile nodes. It is much harder to locate the attackers.

The main targets of an attacker are given as follows:

1) confidentiality, such as identities of other users, valuable data and documents stored on the VC, and the location of the VMs, where the target's services are executing;
2) integrity, such as valuable data and documents stored on the VC, executable code, and result on the VC;
3) availability, such as physical machines and resources, privileges, services, and applications.

One possible form of attack is given below:

1) Find the geographic location of the target vehicle and physically move close the target machine;
2) Narrow down the possible areas where the target user's services are executing by mapping the topology of VC;
3) Launch multiple experimental accesses to the cloud, and find out if the target user is currently on the same VM;
4) Request the services on the same VM where the target user is on;
5) Use system leakage to obtain higher privilege to collect the assets [23].

Due to the features of the VC, there are several challenges for attackers as well. High mobility of vehicles is like a double-edged sword. It makes it hard for attackers to harm a specific target vehicle. First, the vehicle's access of each virtual machine can be transitory as vehicles constantly move from one district to another one, if each district is associated with a virtual machine. Additionally, attackers need to locate on which machine/infrastructure a specific target is located because all users in the VC are distributed on virtual machines. However, it is possible to locate the co-residence of other users. Experiments have been done to catch and compare the memory of processors, and users can find co-residence in the same physical machine [23]. Third, the attackers must be physically co-located with the target user on the same physical machines. This will require attackers to be physically present at the same region with the target vehicles or shadow with the target vehicles at the same speed. These challenges make attacking extremely difficult because coexistence is hard to achieve and is temporary. Finally, the attackers have to collect valuable information with certain privileges or with security tokens.

2) Threats: The threats in the VC can be classified using STRIDE [26]: a system developed by Microsoft for classifying computer security threats. The threat categories are given here.

1) *Spoofing user identity:* The attackers pretend to be another user to obtain data and illegitimate advantages. One classic example is the "man-in-the-middle attack," in which the attackers pretend to be Bob when communicating with Alice and pretend to be Alice when

communicating with Bob. Both Alice and Bob will send decryptable messages to the attackers.

2) *Tampering:* The attackers alter data and modify and forge information.
3) *Repudiation:* The attackers manipulate or forge the identification of new data, actions, and operations.
4) *Information disclosure:* The attackers uncover personally identifiable information such as identities, medical, legality, finance, political, residence and geographic records, biological traits, and ethnicity.
5) *Denial of Service:* The attackers mount attacks that consume system resources and make the resources unavailable to the intended users.
6) *Elevation of privilege:* The attackers exploit a bug, system leakage, design flaw, or configuration mistake in an operating system or software application to obtain elevated access privilege to protected resources or data that are normally protected from normal users.

B. Authentication of High-Mobility Nodes

Security authentication in the VC includes verifying user identity and message integrity. To conduct authentication, there are some metrics that can be adopted [27].

1) Ownership: A user owns some unique identity (e.g., identity card, security token, and software token).
2) Knowledge: A user knows some unique things [e.g., passwords, personal identification number and human challenge response (i.e., security questions)].
3) Biometrics: These include the signature, face, voice, and fingerprint.

However, it is challenging to authenticate vehicles due to high mobility. First, high mobility makes it hard to authenticate messages with a location context. For example, accident alert message associated with locations and events at a specified time are hard to verify because the locations of vehicles are constantly changing. Second, high mobility and a short transmission range may result in the recipient being out of reach. It is likely that a vehicle at the border of access point can change its access point when the authentication message is transmitted back. Third, the security token (security key pairs) is hard update. Some vehicles can even park for years without starting a single time. These situations will make the updating tasks of the security token significantly difficult.

In addition, it is challenging to authenticate a vehicle's or driver's identity in the VC. To protect privacy, these identities are often replaced by pseudonyms. The authentication of identity can be complex and makes Sybil attacks possible [28].

C. Establishing Trust Relationships

Trust is one of the key factors in any secure system. A trust relationship can exist in several ways. The network service providers and the vehicle drivers have access to trust. There will be a large number of government agents, e.g., the Department of Motor Vehicles (DMV) and the Bureau of Motor Vehicles (BMV) are trusted organizations. The relationship between the BMV and vehicle drivers is identity uniqueness and legitimacy.

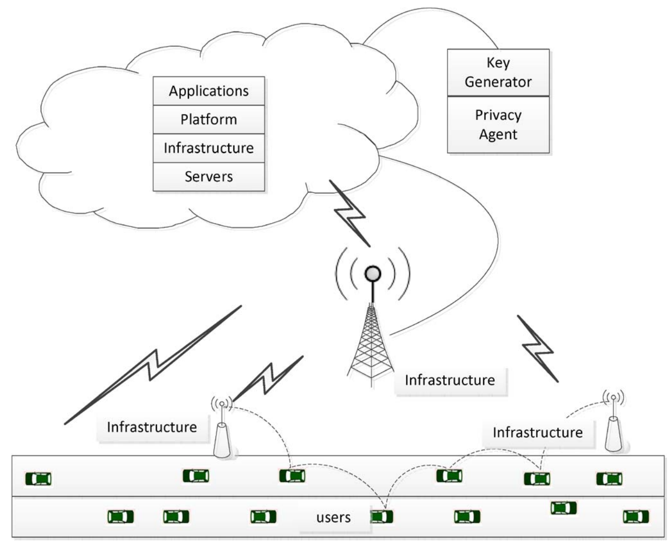

Fig. 2. Vehicles often communicate through multihop routing. A request response will include multiple participants, including users, infrastructure, servers, platform, application, and key generator and privacy agent.

However, the large population of vehicles creates challenges to building trust relationships to all the vehicles at any time. There will be occasional exceptions. In addition, drivers are increasingly concerned about their privacy. Tracking vehicles/drivers will cause worries in most cases. As a result, pseudonyms are often applied to vehicles. On the other hand, a certain level of trust of identity is needed. Some applications such as accident reliability investigation by law enforcement or insurance companies require the driver's identity to be responsible for accidents. Therefore, we assume that a low level of trust relationship exists in VANETs. To obtain a high-level trust relationship, the security scheme discussed in Section IV needs to be executed.

In VCs, it is far more challenging to build trust relationships than in vehicular networks and conventional cloud computing. Fig. 2 shows an example of multiple participants in a VC. The VC is often based on DSRC. Many applications need multi-hop routing, with multiple nodes involved in communication. Therefore, the VC has inherited the challenge of establishing trust relationships among multiple vehicles, roadside infrastructure, service providers, network channels, and even the secret key generator.

In this paper, we assume that the VC cloud infrastructure is trusted, the VC service providers are trusted, the vast majority of VC users are trustworthy, and the attackers have the same privileges as normal users.

D. Location Validation and Pseudonymization

Most, if not all, VC applications rely on accurate location information. Therefore, location information must be validated. There are two approaches to validate location information: active and passive. Vehicles or infrastructure with radar (or camera, etc.) can perform active location validation. Radar input can be used to validate location information. Vehicles or infrastructure without radar, or in a situation where radar detection is not possible, can validate location information by applying statistical methods [9], [29].

A vehicle's identity is often tangled with owner's identity. Because of legal and insurance issues, a vehicle's unique identity (such as vehicle identity number, Internet Protocol address, and hostname) is often linked to the owner's identity. Therefore, tracking a vehicle can often invade its owner's privacy. To protect privacy, one can replace vehicular identity by a pseudonym. The real identity can only be discovered by the Pseudonymization Service Center, which is a secured and trusted entity. The pseudonym is subject to timeout. After expiration, a new pseudonym will be assigned. Digital license plates (DLPs) or electronic license plates, which are a wireless device periodically broadcasting a unique identity string, have been proposed. Temporary public keys as DLPs can protect privacy and can be broadcast [11].

E. Scalability

Security schemes for VCs must be scalable to handle a dynamically changing number of vehicles. Security schemes must handle not only regular traffic but special traffic as well, e.g., the large volume of traffic caused by special events (e.g., football games, air shows, etc.)

The dynamics of traffic produces dynamic demands on security. For example, imagine a downtown area with several supermarkets and stores that take orders from vehicles in traffic, complete with credit card information. To protect credit card information, comprehensive cryptographic algorithms must be applied. However, the comprehensive algorithms decrease the efficiency of communication response time. Therefore, better algorithms and, perhaps, less comprehensive security schemes are needed to speed up the response time.

F. Single-User Interface

Single-user access interface is another challenge to VCs. When the number of service accesses in a cloud increases, the number of VMs that provide the service will increase to guarantee quality of service. More VMs will be created and assigned. With the increase in VMs, security concerns grow as well. When the number of service accesses decreases, the number of VMs that provide the service will decrease to improve resource utilization. Some VMs will be destroyed and recycled. These procedures are transparent to vehicles. Vehicles only see one access interface and do not need to know the changing of VMs. To achieve scalability, a simple solution is to clone and expand the service in a different cloud. However, a single interface obviously makes scalability even more difficult.

G. Heterogeneous Network Nodes

Conventional cloud computing and fixed networks often have homogeneous end users. As it turns out, vehicles have a large array of (sometimes) vastly different onboard devices. Some high-end vehicles have several advanced devices, including a Global Positioning System (GPS) receiver, one or more wireless transceivers, and onboard radar devices. In contrast, some economy models have only a wireless transceiver. Some other vehicles have different combinations of GPS receivers, wireless transceivers, and radar. Different vehicle models have different device capabilities such as speed of processor, volume of memory, and storage. These heterogeneous vehicles as network nodes create difficulties to adapting security strategies. For example, PKI encryption and decryption algorithms will require vehicles to meet certain hardware conditions.

H. VC Messages

1) Safety Messages: The initial motivation of VANET was the dissemination of traffic safety messages. Based on the emergency level, there are three types of safety messages.

1) Level one: public traffic condition information. Vehicles exchange traffic information (e.g., traffic jam) that indirectly affects other vehicles' safety, e.g., a traffic jam increases the likelihood of accidents. This type of message is not sensitive to communication delay, but privacy needs to be protected.
2) Level two: cooperative safety messages. Vehicles exchange messages in cooperative accident avoidance applications. These messages are often time critical, and privacy needs to be protected.
3) Level three: liability messages. After accidents happen, there will be liability messages generated by law enforcement authorities. These messages contain important evidence for liability claims and are bonded by a certain time range. Privacy information is naturally protected.

A common format of safety messages is timestamp, geographic location, speed, percentage of speed change since the last message, direction, acceleration, and percentage of acceleration change since last message. The safety message will append information such as public traffic condition and accidents. The appended message can help determine liability. Driver identity information is not necessary to be part of the safety message. Pseudonyms can be applied to protect the driver's identity. The signature of the safety message can be described as follows: Following the ElGamal signature scheme [30], we define three parameters.

1) H: a collision-free hash function;
2) p: a large prime number that will ensure that computing discrete logarithms modulo p is very difficult;
3) $g < p$: a randomly chosen generator out of a multiplicative group of integers modulo p.

Each vehicle has long-term PKI public/private key pairs:

- private key: S;
- public key: $\langle g, p, T \rangle$, where $T = g^S \bmod p$.

It should be noted that a message m can be combined as $m|T$, where T is the timestamp. The timestamp can ensure the freshness of the message. For each message m to be signed, three steps are followed.

1) Generate a per-message public/private key pair of S_m (private) and $T_m = g^{S_m} \bmod p$ (public).
2) Compute the message digest $d_m = H(m|T_m)$ and the message signature $X = S_m + d_m S \bmod (p-1)$, where mod is the modulo operation and $|$ is the concatenation operator.
3) Send m, T_m, and X.

To verify the message, three steps are followed.

1) Compute the message digest $d_m = H(m|T_m)$.
2) Compute $Y_1 = g^X$ and $Y_2 = T_m T^{d_m}$.
3) Compare $Y_1 = Y_2$. If $Y_1 = Y_2$, then the signature is correct.

The reason is

$$Y_1 = g^X = g^{S_m + d_m S} = g^{S_m} g^{d_m S} = T_m g^{S d_m} = T_m T^{d_m} = Y2.$$

2) Confidential Messages: To ensure the confidentiality of a sensitive message, the message will be both signed and encrypted. Suppose that vehicle A sends a sensitive message m to vehicle B. Each vehicle has its own PKI public/private key pairs. Thinking of the overhead of PKI processing time, we can adapt a symmetric encryption algorithm. However, to exchange a secret key, we still need to use PKI support. The handshake of exchanging the secret key is defined as follows:

$$A \rightarrow B : B|K|T_{\mathrm{pub}_B}, SigB|K|T_{\mathrm{pri}_A}$$

where A and B are the identities of vehicles A and B, respectively; K is the secret key shared by A and B; m is the sensitive message; T is the timestamp; pub_B is the public key of B; and pri_A is the private key of A.

Once A and B both know the secret key K, they can communicate by using a well-known message authentication code (MAC or HMAC). Hashing the sensitive message is done as follows:

$$A \leftrightarrow B : m, \mathrm{MAC}_K m.$$

There are potential problems with this approach. As a drawback of symmetric encryption, nonrepudiation (i.e., integrity and origin of data) cannot be ensured, although the likelihood of data being surreptitiously changed is extremely low. This is a compromise solution between efficiency and security. To achieve a higher level of security for sensitive messages, one can apply active security mechanisms [9] or adopt PKI encryption at the cost of losing a certain amount of efficiency. In multihop networks, the key handshake in this scheme does not scale well in zero-infrastructure VANET, but it can scale well with the aid of roadside infrastructure.

I. Key Management

1) Key Assignment and Rekeying: In VANETs, some organizations can serve as CAs: governmental transportation authorities, vehicle manufacturers, or nonprofit organizations.

Initially, a vehicle will receive a key pair from the manufacturer or some governmental authority. Key assignment is on the basis of a unique ID with a certain expiration time. Upon expiration, the key pair has to be renewed at the local DMV/BMV. The renewal/expiration period can be the same period of vehicular state inspection, e.g., mandatory annual state inspection in many U.S. states.

2) Key Verification: To verify key pairs, we assume that every vehicle trusts CAs and that CAs are tamper-proof. Key validation can be done at the CAs or sub-CAs. Let pub_i of vehicle i be the public key issued by a CA j, i.e., CA_j. Vehicle i will have a certificate $\mathrm{cert}_i[\mathrm{pub}_i]$ assigned by CA_j when CA_j assigns the public key. The process of validating public key will compute the following certificate at CA_j:

$$\mathrm{cert}_i[\mathrm{pub}_i] = \mathrm{pub}_i|\mathrm{sig}_{\mathrm{pri}_{\mathrm{CA}_j}}(\mathrm{pub}_i|\mathrm{ID}_{\mathrm{CA}_j})$$

where $\mathrm{pri}_{\mathrm{CA}_j}$ is the private key of CA_j, and $\mathrm{ID}_{\mathrm{CA}_j}$ is the identity of CA_j. The idea is to sign the special message $\mathrm{pub}_i|\mathrm{ID}_{\mathrm{CA}_j}$ using the private key of CA_j. The digital signature algorithm has been discussed in Section IV-H1.

3) Key Revocation: Key revocation is an important and effective way to prevent attacks. There are certain cases when key pairs will be exposed to attackers. It is obvious that an exposed key pair needs to be disabled. One of the advantages of PKI is that PKI can revoke a key pair. Vehicles will be aware that the exposed key pair has been revoked and refuse to communicate with vehicles with invalid key pairs. PKI uses certificate revocation lists (CRLs) to revoke keys. CRLs include a list of the most recently revoked certificates and are instantly distributed to vehicles. In VANETs, the infrastructure can serve as CRL distributors.

The CAs can revoke key pairs by using onboard tamper-proof devices. Suppose that CAs want to revoke the key pairs of vehicle V. CAs will send out the revoke message signed by public key of V to the tamper-proof devices. After receiving this revoking message, the tamper-proof device will validate the message and revoke the key pairs. The tamper-proof device will also send back an ACK to the CA to confirm the operation. To improve communication between V and CA, the vehicle's location is retrieved to select the closest CA. If the latest vehicle location failed to be retrieved, the last location will be used to select the closest CA. In this case, the CA will use a broadcasting message to revoke the key pairs. The broadcasting message can be sent out by using several media such as FM, Internet, and satellite.

To avoid attackers reporting other vehicles to CA to revoke the key pairs of other vehicles, revocation will be triggered by a certain number of neighboring vehicles. There is another risk that attackers can launch planned attacks. For example, several attackers can surround a well-behaved vehicle and report the well-behaved vehicle as a misbehaving vehicle. Prevention of this risk is very challenging. Due to the dynamics of traffic, it is costly to launch such an attack. One possible solution is to build behavior history records and credit the past behavior into values, just like the bank credit system. A similar solution has been discussed as Map History [9].

V. Research Approach

In this section, we offer a first attempt to addressing several of the challenges previously discussed. We begin by describing the two VC models, i.e., infrastructure- and ad-hoc-based models. We then demonstrate algorithms to enhance authentication of high-mobility vehicles, configure customized security schemes, and improve scalability of security schemes.

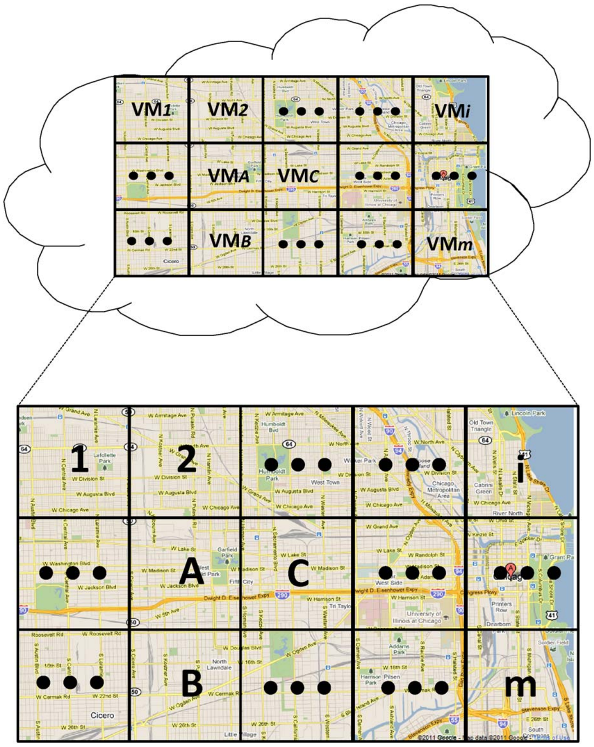

Fig. 3. Downtown area partitioned into cells, each mapped to a virtual machine.

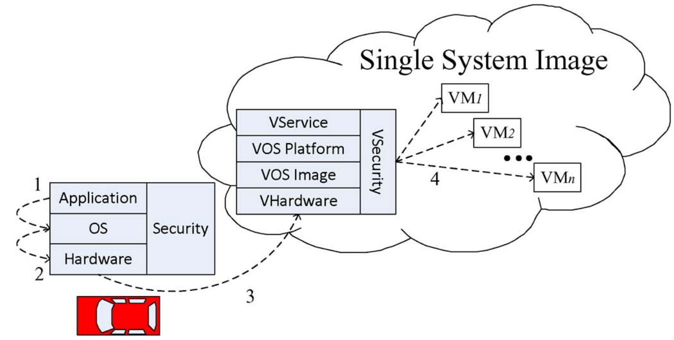

Fig. 4. Vehicle node in a cell can communicate with a virtual machine that is responsible for the cell.

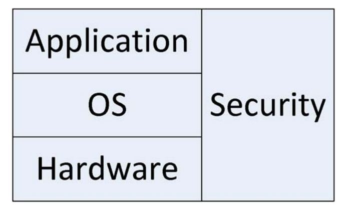

Fig. 5. Vehicle node image is located on each individual vehicle.

A. The Cloud Model

The cloud in this proposal is associated with a number of grids. A city or a traffic area is partitioned into grids. The grid size is predefined, e.g., 700 m^2 and with two GPS coordinates. The grid of a city is shown in Fig. 3. Each cell is associated with a virtual machine in the cloud. The virtual machine can dynamically request resources from cloud. For example, when the grid is congested, the corresponding virtual machine will request more communicating, storage, and computing resources. The cloud will be able to borrow these resources from the idle virtual machine, which is associated with sparse traffic grid. Therefore, the traffic of the whole city can be mapped to the cloud.

This cloud model provides high capability in customizing cloud services and the security scheme. For example, a downtown area is often queried about vacant parking spots and congestion status. The corresponding virtual machine can be specially configured and optimized in the smart parking and congestion control services. At a busy intersection, a collision-warning service can be specialized and optimized in the virtual machine. A possible solution is to collect and sort all the vehicles' mobility information at the intersection. When vehicles are too close to each other by considering the headway distance and relative speed, the vehicles will receive an alarm from the cloud. Even cheaper cars that have no radar cruise control system can get benefits from the cloud collision warning system.

What distinguishes vehicles from standard nodes in a conventional cloud is *autonomy* and *mobility*. Indeed, large numbers of vehicles spend substantial time on the road and may be involved in dynamically changing situations; we argue that, in such situations, the vehicles have the potential to cooperatively solve problems that would take a centralized system an inordinate amount of time, rendering the solution useless [2]. Vehicles automatically form a cloud by connecting virtual cells, which can be a group of vehicles. Each virtual cell is associated with a virtual machine in which vehicles rent or contribute their spare computing, storage, and sensing resource. The group of vehicles moves at almost the same speed. Since vehicles are cloud constructors and cloud users, all vehicles inside a cell can directly receive packets from each other. A cell leader can be elected to communicate with other clouds [9].

1) Virtual Machines of VCs: This objective concerns how a cloud is formed and how the service can be provided. We first consider the basic modules of the VC and then introduce the process of a service request and response.

The communication between a vehicle and the cloud is through a unique entry. The cloud provides a single system image to each individual virtual machine shown as Fig. 4. Each vehicle has a node image, which includes hardware drivers, operating system image, security system, and applications, as shown in Fig. 5. When the applications of the vehicle send a request to the cloud, the request will be forwarded to the operating system and, then, the hardware (network driver). The request will be sent by the wireless network and received by the cloud single system image. The allocator of the cloud will locate which virtual machine should be responsible for the request and forward the request to the virtual machine. If the request needs to access other virtual machines, e.g., to check the traffic congestion status of a city in a remote state, the virtual machine can communicate with other virtual machines as well.

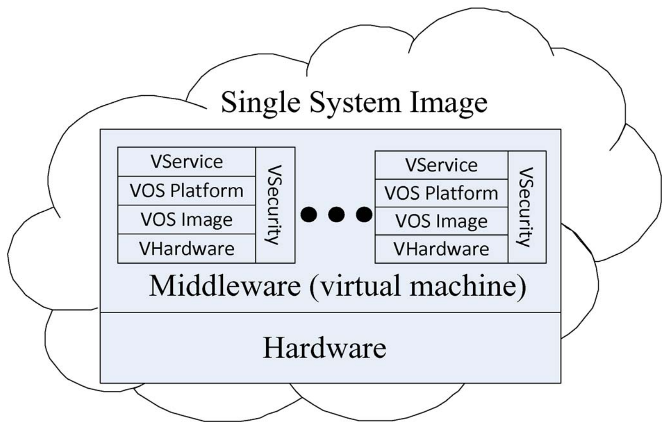

Fig. 6. Cloud provides a single system image and is composed by a number of virtual machines.

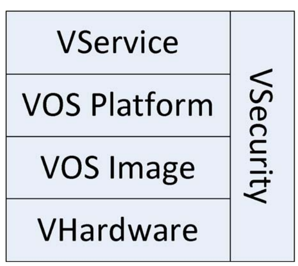

Fig. 7. Single virtual machine located in the cloud.

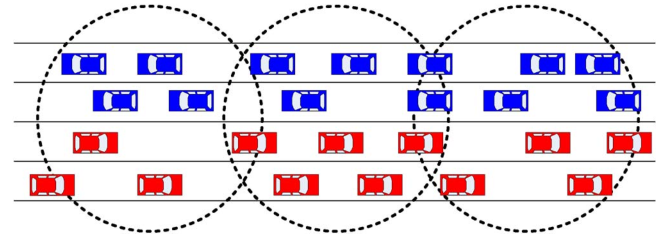

Fig. 8. Trust relationship in AVCs can be built on the basis of a group of vehicles. The behavior of a vehicle can be monitored by all members.

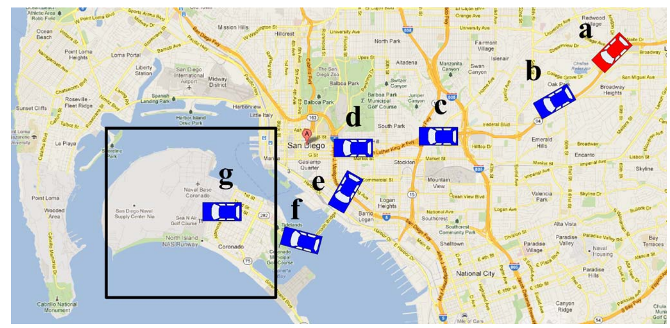

Fig. 9. Geographic location-based security mechanism. The shaded square is the naval base. Only the vehicles in the shaded rectangle region (i.e., vehicle $\boldsymbol{g}$ can decrypt and access the received ciphertext sent by vehicle $\boldsymbol{a}$).

The VC is a single system image composed of a number of virtual machines. A single image can be created by a layer of middleware between the hardware manager system and a number of virtual machines, as shown in Fig. 6. The middleware is a cloud operating system and a platform to allocate a large number of virtual machines. Each virtual machine is composed of virtual hardware, virtual operating system image, virtual operating system platform, virtual security system, and virtual services, as shown in Fig. 7. The virtual hardware is composed of several real computers that virtually act as real hardware and provide the interface of the hardware. The virtual operating system image can be any current operating system, such as Linux/Unix or Windows. The virtual operating system platform includes not only the operating system but system applications such as web server and databases. The virtual security system is a set of complete security solutions, including hardware and software. The customized security protocols can be configured and replaced in this module. The virtual services are actual services that are configured for the related traffic area/grid.

B. Securing VCs

1) Trust Relationship: For infrastructure-based VC, trust relationships can be built by infrastructures that are constructed by authorities such as BMV/DMV or other transportation agencies. Infrastructure will be authenticated and assigned with security key pairs. Infrastructure stores the key pairs in tamper-proof devices. As shown in Fig. 2, vehicles communicate with infrastructure as access point to the VC. The infrastructure is sufficiently capable to handle large numbers of accesses in its transmission range. The scalability of trust relationships can be achieved because the infrastructure is connected to each other by fixed networks.

For AVCs, trust relationships can be built as well. A cell leader can be elected to represent the members in the cell to communicate with other cells. For security reasons, the cell leader is monitored by its neighbors. When the leader sends and receives aggregated position packets, all the members in the cell will compare the positions in the packets based on their knowledge. By remaining silent, they confirm that the packets have not been altered. Otherwise, they broadcast protest packets against the leader. The other neighbors will put the leader and the protestor vehicle into the question table after receiving the protest packet. Then, the opinion of the other neighbors is taken into account. If the majority of vehicles regard the leader as malicious, the record of the leader is moved to the distrust table, as discussed by Yan *et al.* [9]. Otherwise, the records sent by the leader are placed in the trust table (see Fig. 8).

2) Authentication and Confidentiality: To provide authentication and confidentiality, we propose a geographic location-based security mechanism to ensure physical security on top of conventional methods. Messages are encrypted with a geographic location key that specifies a decryption region. This provides *physical* security because a vehicle has to be physically present in the decryption region to decrypt ciphertext encrypted with this geographic location key. As an example, Fig. 9 shows a shaded square that is a location-based security region. Sender vehicle a specifies the region, creates the location key, encrypts the message, and sends ciphertext to vehicles in this region. Vehicles outside this region such as b, c, d, and e cannot decrypt the message. Only vehicle f can decrypt the message because it is physically inside the decryption region. Since the decryption region can be dynamically specified, attacks are extremely expensive and difficult to mount.

C. *Configuring Security Strategies*

It is important to allow the VC to dynamically configure the security protocols and to independently replace security strategies. We will start with the configuration of security protocols and then describe an intelligent task management method.

1) More Vehicles Involved, More Secure Cloud Needed: The cloud will provide vehicles a single system image that is transparent of details of security scheme changes. As vehicles are dynamically moving in and out of a cell, the security protocols of a cell in its virtual machine need to be dynamically adjusted. We observe the fact that the more vehicles are involved, the more secure and the stricter a protocol should be. Similar facts can be found in daily life. Airports are often crowded, and security is often stricter than that in many other places. Events such as football games, auto races, and air shows often attract more people, as well as more policemen who patrol the area more often to ensure the security of attendees.

Therefore, it is important to know the expected volume of vehicles at any time to dynamically switch security protocols. We are interested in the following problem to evaluate the expected number of vehicles at any given time. Consider a cell with finite capacity N. At time $t = 0$, the cell contains $n_0 \geq 0$ cars. After that, cars arrive and depart at time-dependent rates, as described next. If the cell contains k, $(0 \leq k \leq N)$ cars at time t, then the car arrival rate $\alpha_k(t)$ is

$$\alpha_k(t) = \frac{N-k}{N}\lambda(t)$$

and the car departure rate $\beta_k(t)$ is

$$\beta_k(t) = k\mu(t)$$

where, for all $t \geq 0$, $\lambda(t)$ and $\mu(t)$ are *integrable* on $[0, t]$. It is worth noting that both $\alpha_k(t)$ and $\beta_k(t)$ are functions of both t and k. In particular, it may well be the case that, for $t_1 \neq t_2$, $\alpha_k(t_1) \neq \alpha_k(t_2)$, and similarly for $\beta_k(t_1)$ and $\beta_k(t_2)$, giving a mathematical expression to the fact that, at different times of the day, for example, the departure rate depends on not only the number of cars present in the cell but on the time-dependent factors as well.

Consider the counting process $\{X(t)|t \geq 0\}$ of continuous parameter t, where, for every positive integer k, $(1 \leq k \leq N)$, the event $\{X(t) = k\}$ occurs if the cell contains k, cars at time t. We let $P_k(t)$ denote the probability that the event $\{X(t) = k\}$ occurs. In other words

$$P_k(t) = \Pr\left[\{X(t) = k\}\right].$$

In addition to $P_k(t)$, of interest are the expected number $E[X(t)]$ and the variance $Var[X(t)]$ of the number of cars in the cell at time $t > 0$, as well as the limiting behavior of these parameters as $t \to \infty$, whenever such a limit exists and/or makes sense.

To make the mathematical derivations more manageable, we set $P_k(t) = 0$ for $k < 0$ and $k > N$. Thus, $P_k(t)$ is well defined for all integers $k \in (-\infty, \infty)$ and for all $t \geq 0$. In particular, the assumption about the cell containing n_0 cars at $t = 0$ translates into $P_k(0) = 1$ if $k = n_0$ and 0 otherwise.

Let t, $(t \geq 0)$, be arbitrary, and let h be sufficiently small such that, in the time interval $[t, t+h]$, the probability of two or more arrivals or departures, or of a simultaneous arrival and departure, is $o(h)$. With h chosen as stated, the probability $P_k(t+h)$ that the cell contains k, $(0 \leq k \leq N)$ cars at time $t+h$ has three components.

1) $P_k(t)[1 - h(N - k/N)\lambda(t) - kh\mu(t) + o(h)]$.
2) $P_{k-1}(t)[h(N - k + 1/N)\lambda(t) + o(h)]$.
3) $P_{k+1}(t)[(k+1)h\mu(t) + o(h)]$.

Here, by assumption, $P_k = 0$ for $k < 0$ and $k > N$.

The expression of probability $P_k(t)$ can be derived by

$$P_k(t) = 1 - e^{-h(t)} \int_0^t \mu(u) e^{h(u)} \mathrm{d}u$$

where

$$h(x) = \int_0^x \left[\frac{\lambda(s)}{N} + \mu(s)\right] \mathrm{d}s.$$

We can write the linearity of expectation as

$$E\left[X(t)\right] = e^{-h(t)} \left[n_0 + \int_0^t \lambda(u) e^{h(u)} \mathrm{d}u\right].$$

D. *Enhancing Scalability of Security Schemes*

When vehicle population increases in a certain area, not only the scalability of the VC but also the scalability of security schemes becomes a tough problem. In our cloud model, the scalability of the security scheme can be enhanced by a virtual machine division algorithm, a highly scalable algorithm. When the number of access of a virtual machine grows sufficiently large, compared to an empirical threshold, the virtual machines (as a super-VM) will divide itself into multiple subvirtual machines (as sub-VMs). Each virtual machine will obtain the same amount of resources as the original super VM. The middleware of the super VM can randomly forward request to subvirtual machines to load balance. The middleware of the super VM also caches the most recently accessed and frequent information. It caches and executes information such as frequently asked questions (FAQs) and answers. If access from a vehicle hits the FAQ, the middleware directly sends back the answer. If the access misses the FAQ, the middleware then forwards access to a relatively idle VM. This can further reduce the workload of sub-VMs (see Fig. 10).

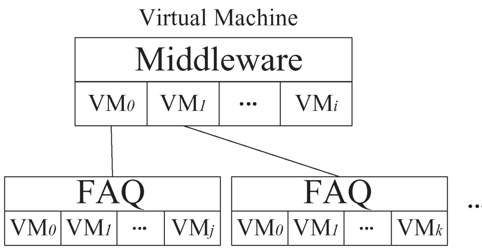

Fig. 10. Virtual machine can be divided into multilayers of VMs. Each layer is composed by multiple VMs. The middleware can also be deployed with a cache of frequently accessed information.

VI. Concluding Remarks

In this paper, we have addressed the security challenges of a novel perspective of VANETs, i.e., taking VANETs to clouds. We have first introduced the security and privacy challenges that VC computing networks have to face, and we have also addressed possible security solutions. Although some of the solutions can leverage existing security techniques, there are many unique challenges. For example, attackers can physically locate on the same cloud server. The vehicles have high mobility, and the communication is inherently unstable and intermittent. We have provided a directional security scheme to show an appropriate security architecture that handles several, not all, challenges in VCs. In future work, we will investigate the brand-new area and design solutions for each individual challenge. Many applications can be developed on VCs. As future work, a specific application will need to analyze and provide security solutions.

Extensive work of the security and privacy in VCs will become a complex system and need a systematic and synthetic way to implement intelligent transportation systems [32], [33]. Only with joint efforts and close cooperation among different organizations such as law enforcement, government, the automobile industry, and academics can the VC computing networks provide solid and feasible security and privacy solutions.

Acknowledgment

The authors would like to thank three anonymous referees for their constructive comments and criticism that helped us improve the organization of this paper.

References

[1] S. Arif, S. Olariu, J. Wang, G. Yan, W. Yang, and I. Khalil, "Datacenter at the airport: Reasoning about time-dependent parking lot occupancy," *IEEE Trans. Parallel Distrib. Syst.*, 2012, [Online]. Available: https://csdl2.computer.org/csdl/trans/td/preprint/ttd2012990021-abs.html, to be published.

[2] S. Olariu, M. Eltoweissy, and M. Younis, "Toward autonomous vehicular clouds," *ICST Trans. Mobile Commun. Comput.*, vol. 11, no. 7–9, pp. 1–11, Jul.–Sep. 2011.

[3] S. Olariu, I. Khalil, and M. Abuelela, "Taking VANET to the clouds," *Int. J. Pervasive Comput. Commun.*, vol. 7, no. 1, pp. 7–21, 2011.

[4] L. Li, J. Song, F.-Y. Wang, W. Niehsen, and N. Zheng, "IVS 05: New developments and research trends for intelligent vehicles," *IEEE Intell. Syst.*, vol. 20, no. 4, pp. 10–14, Jul./Aug. 2005.

[5] G. Yan and S. Olariu, "A probabilistic analysis of link duration in vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1227–1236, Dec. 2011.

[6] H. Xie, L. Kulik, and E. Tanin, "Privacy-aware traffic monitoring," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 61–70, Mar. 2010.

[7] D. Huang, S. Misra, G. Xue, and M. Verma, "PACP: An efficient pseudonymous authentication based conditional privacy protocol for vanets," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 736–746, Sep. 2011.

[8] R. Hasan, *Cloud Security*. [Online]. Available: www.cs.jhu.edu/~ragib; http://www.ragibhasan.com/research/cloudsec.html

[9] G. Yan, S. Olariu, and M. C. Weigle, "Providing VANET security through active position detection," *Comput. Commun.*, vol. 31, no. 12, pp. 2883–2897, Jul. 2008, Special Issue on Mobility Protocols for ITS/VANET.

[10] G. Yan, S. Olariu, and M. Weigle, "Providing location security in vehicular ad hoc networks," *IEEE Wireless Commun.*, vol. 16, no. 6, pp. 48–55, Dec. 2009.

[11] J. Sun, C. Zhang, Y. Zhang, and Y. M. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 9, pp. 1227–1239, Sep. 2010.

[12] G. Yan and S. Olariu, "An efficient geographic location-based security mechanism for vehicular ad hoc networks," in *Proc. IEEE Int. Symp. TSP*, Macau SAR, China, Oct. 2009, pp. 804–809.

[13] A. Friedman and D. West, "Privacy and security in cloud computing," *Center for Technology Innovation: Issues in Technology Innovation*, no. 3, pp. 1–11, Oct. 2010.

[14] J. A. Blackley, J. Peltier, and T. R. Peltier, *Information Security Fundamentals*. New York: Auerbach, 2004.

[15] N. Santos, K. P. Gummadi, and R. Rodrigues, "Toward trusted cloud computing," in *Proc. HotCloud*, Jun. 2009.

[16] T. Garfinkel, B. Pfaff, J. Chow, M. Rosenblum, and D. B. Terra, "Virtual machine-based platform for trusted computing," in *Proc. ACM SOSP*, 2003, pp. 193–206.

[17] S. Berger, R. Cáceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn, "VTPM: Virtualizing the trusted platform module," in *Proc. 15th Conf. USENIX Sec. Symp.*, Berkeley, CA, 2006, pp. 305–320.

[18] D. G. Murray, G. Milos, and S. Hand, "Improving XEN security through disaggregation," in *Proc. 4th ACM SIGPLAN/SIGOPS Int. Conf. VEE*, New York, 2008, pp. 151–160.

[19] F. J. Krautheim, "Private virtual infrastructure for cloud computing," in *Proc. Conf. Hot Topics Cloud Comput.*, 2009, pp. 1–5.

[20] M. Jensen, J. Schwenk, N. Gruschka, and L. L. Iacono, "On technical security issues in cloud computing," in *Proc. IEEE Int. Conf. Cloud Comput.*, 2009, pp. 109–116.

[21] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *Proc. IEEE INFOCOM*, San Diego, CA, 2010, pp. 1–9.

[22] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proc. 14th ESORICS*, 2009, pp. 355–370.

[23] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds," in *Proc. 16th ACM Conf. CCS*, 2009, pp. 199–212.

[24] SIRIT-Technologies, White paper. DSRC technology and the DSRC industry consortium (DIC) prototype team.

[25] D. Wen, G. Yan, N. Zheng, L. Shen, and L. Li, "Toward cognitive vehicles," *IEEE Intell. Syst. Mag.*, vol. 26, no. 3, pp. 76–80, May–Jun. 2011.

[26] Microsoft, The stride threat model. [Online]. Available: http://msdn.microsoft.com

[27] Fed. Fin. Inst. Examination Council, Authentication in an Internet banking environment2009. [Online]. Available: http://www.ffiec.gov/pdf/authentication_guidance.pdf

[28] J. Douceur, "The sybil attack," in *Proc. Rev. Papers 1st Int. Workshop Peer-to-Peer Syst.*, 2002, vol. 2429, pp. 251–260.

[29] G. Yan, W. Yang, E. F. Shaner, and D. B. Rawat, "Intrusion-tolerant location information services in intelligent vehicular networks," *Commun. Comput. Inf. Sci.,*, vol. 135, pp. 699–705, 2011.

[30] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inf. Theory*, vol. IT-31, no. 4, pp. 469–472, Jul. 1985.

[31] Nat. Inst. Stand. Technol., Gaithersburg, MD, The NIST Definition of Cloud Computing, 2011.

[32] J. Li, S. Tang, X. Wang, W. Duan, and F.-Y. Wang, "Growing artificial transportation systems: A rule-based iterative design process," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 322–332, Jun. 2011.

[33] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.

Gongjun Yan received the Ph.D. degree from Old Dominion University, Norfolk, VA, in 20[illegible].

He is an Assistant Professor of informatics with Indiana University Kokomo. His research interests include information security and privacy, intelligent vehicles, vehicular ad hoc networks, and wireless communications.

Ding Wen is currently a Professor with the Center for Military Computational Experiments and Parallel Systems Technology, National University of Defense Technology Changsha, Hunan, China. His research interests include intelligent systems and unmanned systems.

Stephan Olariu received the Ph.D. degree in computer science from McGill University, Montreal, QC, Canada, in 1986.

He is currently a Professor of computer science with Old Dominion University, Norfolk, VA. He has held many different roles and responsibilities as a member of numerous organizations and teams. Much of his experience has involved the design and implementation of robust protocols for wireless networks and, particularly, sensor networks and their applications. He is currently applying mathematical modeling and analytical frameworks to the resolution of problems ranging from securing communications to predicting the behavior of complex systems and evaluating the performance of wireless networks.

Michele C. Weigle received the Ph.D. degree in computer science from the University of North Carolina, Chapel Hill, in 2003.

She is currently an Associate Professor of computer science with Old Dominion University, Norfolk, VA. Her research interests include vehicular networks, mobile ad-hoc networks, wireless networking, sensor networks, network simulation and modeling, and Internet congestion control.

AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

AQ1 = Please confirm which of the two provided web address for reference [8] should we use.

END OF ALL QUERIES

Security Challenges in Vehicular Cloud Computing

Gongjun Yan, Ding Wen, Stephan Olariu, and Michele C. Weigle

***Abstract*—In a series of recent papers, Prof. Olariu and his co-workers have promoted the vision of vehicular clouds (VCs), a nontrivial extension, along several dimensions, of conventional cloud computing. In a VC, underutilized vehicular resources including computing power, storage, and Internet connectivity can be shared between drivers or rented out over the Internet to various customers. Clearly, if the VC concept is to see a wide adoption and to have significant societal impact, security and privacy issues need to be addressed. The main contribution of this work is to identify and analyze a number of security challenges and potential privacy threats in VCs. Although security issues have received attention in cloud computing and vehicular networks, we identify security challenges that are specific to VCs, e.g., challenges of authentication of high-mobility vehicles, scalability and single interface, tangled identities and locations, and the complexity of establishing trust relationships among multiple players caused by intermittent short-range communications. Additionally, we provide a security scheme that addresses several of the challenges discussed.**

***Index Terms*—Challenge analysis, cloud computing, privacy, security, vehicular cloud.**

I. Introduction

In an effort to help their vehicles compete in the marketplace, car and truck manufacturers are offering increasingly more potent onboard devices, including powerful computers, a large array of sensors, radar devices, cameras, and wireless transceivers. These devices cater to a set of customers that expect their vehicles to provide seamless extension of their home environment populated by sophisticated entertainment centers, access to Internet, and other similar wants and needs. Powerful onboard devices support new applications, including location-specific services, online gaming, and various forms of mobile infotainment [4].

In spite of the phenomenal growth of third-party applications catering to the driving public, it has been recently noticed that, most of the time, the huge array of onboard capabilities are chronically underutilized. In a series of recent papers [1]–[3], Olariu and his co-workers have put forth the vision of vehicular clouds (VCs), a nontrivial extension of conventional cloud computing, intended to harness the excess capabilities in our vehicles. Vehicles and roadside infrastructure with idle sophisticated onboard devices for long periods of time can be recruited to form a VC. A VC can be formed on-the-fly by dynamically integrating resources and collecting information. Vehicles can access the cloud and obtain, at the right time and the right place, all the needed resources and applications that they need or want.

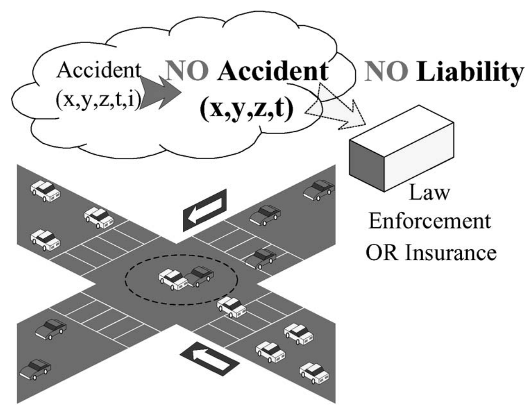

Fig. 1. Illustrating a security issue in VCs.

Obviously, security and privacy issues need to be addressed if the VC concept is to be widely adopted. Conventional networks attempt to prevent attackers from entering a system. However, in VC, all the users, including the attackers, are equal. The attackers and their targets may be physically colocated on one machine. The attackers can utilize system loopholes to reach their goals, such as obtaining confidential information and tampering with the integrity of information and the availability of resources. Fig. 1 shows one possible example of tampering with the integrity of information in the case of a road accident. Imagine that an accident has occurred at an intersection, and the accident will be reported to the VC. The driver liable for the accident can invade the VC and modify the accident record. Later, when the law enforcement or the vehicle insurance company query the accident, they cannot link the accident to the driver who caused it.

Superficially, the security issues encountered in VCs may look deceivingly similar to those experienced in other networks. However, a more careful analysis reveals that many of the classic security challenges are exacerbated by the characteristic features of VCs to the point where they can be construed as VC-specific. For example, the high mobility of vehicles is apt to cause significant challenges related to managing authentication, authorization, and accountability since the vehicles communicate through short-range dedicated short-range communications (DSRC) transceivers [5]. Vehicular mobility and tangled identities and locations also cause significant challenges of

Manuscript received September 21, 2011; revised March 20, 2012 and June 19, 2012; accepted July 21, 2012. This work was supported in part by Indiana University Kokomo under Grant 2263160, by the National Science Foundation (NSF) of China 11126333, and by NSF Grant CNS 0721586 and Grant CNS-1116238. The Associate Editor for this paper was L. Li.

G. Yan is with Indiana University Kokomo, Kokomo, IN 46904 USA (e-mail: goyan@iuk.edu).

D. Wen is with the Center for Military Computational Experiments and Parallel Systems Technology, National University of Defense Technology Changsha, Hunan 410073, China (e-mail: dingwen2010@gmail.com).

S. Olariu and M. C. Weigle are with Old Dominion University, Norfolk, VA 23529 USA (e-mail: olariu@cs.odu.edu; mweigle@cs.odu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2012.2211870

privacy [6]. Employing pseudonyms [7] is a common solution, but the high mobility makes the task of updating pseudonyms quite difficult.

The two main contributions of this work are to identify and analyze security challenges and privacy threats that are VC specific and to propose a reasonable security framework that addresses some of the VC challenges identified in this paper.

II. State of the Art

The security challenges in VC are a new, exciting, and unexplored topic. Vehicles will be autonomously pooled to create a cloud that can provide services to authorized users. This cloud can provide real-time services, such as mobile analytic laboratories, intelligent transportation systems, smart cities, and smart electric power grids. Vehicles will share the capability of computing power, Internet access, and storage to form conventional clouds. These researchers have only focused on providing a framework for VC computing, but as already mentioned, the issue of security and privacy has not yet been addressed in the literature. As pointed out by Hasan [8], cloud security becomes one of the major barriers of a widespread adoption of conventional cloud services. Extrapolating from the conclusions of [8], we anticipate that the same problems will be present in VCs.

Recently, vehicular ad hoc network (VANET) security and privacy have been addressed by a large number of papers. Yan *et al.* [9], [10] proposed active and passive location security algorithms. Radar can be employed as a "virtual eye," and onboard radar can detect the location of vehicles. Public Key Infrastructure (PKI) and digital signature-based methods have been well explored in VANETs [11]. A certificate authority (CA) generates public and private keys for nodes. The purpose of digital signature is to validate and authenticate the sender. The purpose of encryption is to disclose the content of messages only to entitled users. PKI is a method that is well suited for security purposes, particularly for roadside infrastructure. GeoEncrypt in VANETs has been proposed by Yan *et al.* [12]. Their idea is to use the geographic location of a vehicle to generate a secret key. Messages are encrypted with the secret key, and the encoded texts are sent to receiving vehicles. The receiving vehicles must be physically present in a certain geographic region specified by the sender to be able to decrypt the message.

Recently, some attention has been devoted to the general security problem in clouds, although not associated with vehicular networks [13]. The simple solution is to restrict access to the cloud hardware facilities. This can minimize risks from insiders [14]. Santos *et al.* [15] proposed a new platform to achieve trust in conventional clouds. A trust coordinator maintained by an external third party is imported to validate the entrusted cloud manager, which makes a set of virtual machines (VMs) such as Amazon's E2C (i.e., Infrastructure as a Service, IaaS) available to users. Garfinkel *et al.* [16] proposed a solution to prevent the owner of a physical host from accessing and interfering with the services on the host. Berger *et al.* [17] and Murray *et al.* [18] adopted a similar solution. When a VM boots up, system information such as the basic input output system (BIOS), system programs, and all the service applications is recorded, and a hash value is generated and transmitted to a third-party Trust Center. For every period of time, the system will collect system information of the BIOS, system programs, and all the service applications and transmit the hash value of system information to the third-party Trust Center. The Trust Center can evaluate the trust value of the cloud. Krautheim [19] also proposed a third party to share the responsibility of security in cloud computing between the service provider and client, decreasing the risk exposure to both. Jensen *et al.* [20] stated technical security issues of using cloud services on the Internet access. Wang *et al.* [21], [22] proposed public-key-based homomorphic authenticator and random masking to secure cloud data and preserve privacy of public cloud data. The bilinear aggregate signature has been extended to simultaneously audit multiple users. Ristenpart *et al.* [23] presented experiments of locating co-residence of other users in cloud VMs.

III. Vehicular Clouds: Paradigm Shift

A. Conceptual Overview

1) Cloud Computing: In recent years, cloud computing and its myriad applications that promise to change the way we think about computing and data storage have received a huge amount of attention. Cloud users do not need to install expensive hardware and software on their local machine. They can subscribe and use both hardware and software *as a service* when they want to use it. In addition, fees are charged based on the usage of the service. The users can access these services through Internet browsers, and no expensive client terminals are needed. Service providers can make good use of *excess* capabilities on the server side including processors, storage, and sensors that can be used to provide services to clients.

2) VANET: In VANETs, the vehicles communicate with each other and/or with the roadside infrastructure using the Federal Communications Commission-mandated DSRC [24], restricting the transmission range to 300–1000 m. There are two types of VANET networks: the zero-infrastructure and the infrastructure-based VANET. The zero-infrastructure VANET is created on-the-fly. There are many challenging security and privacy problems because no infrastructure is used for authentication and authorization. The infrastructure-based VANET can be formed based on the roadside infrastructure. The infrastructure can act as wireless access points for authentication and authorization purposes. By the same token, the vehicles can use the infrastructure to report events and to exchange information.

3) VCs: Similar to VANETs, there are two types of VCs. In the first type called *Infrastructure-based VC*, drivers will be able to access services by network communications involving the roadside infrastructure. In the second type called *Autonomous VC* (AVC) [2], vehicles can be organized on-the-fly to form VC in support of emergencies and other ad hoc events.

VCs provide services at three levels, i.e., application, platform, and infrastructure. Service providers use the levels differently based on what and how the services are offered. The fundamental level is called *Infrastructure as a Service* (IaaS), where infrastructure such as computing, storage, sensing, communicating devices, and software are created as VMs. The next level is *Platform as a Service* (PaaS), where components and services (such as httpd, ftpd, and email server) are provided and configured as a service. The top level is called *Software as a Service* (SaaS), where applications are provided in a "pay-as-you-go" fashion.

VCs provide a cost-efficient way to offer comprehensive services. For example, a cheaper vehicle with network access can access a VM with strong computation, communication, sensing capability, and large storage. Many applications such as traffic news, road conditions, or intelligent navigation systems can be provided by a VM [25].

B. Potential Applications of VC Computing

In this section, we review several possible applications of VCs.

1) *Vehicle maintenance:* Vehicles receive software updates from cloud whenever vehicle manufacturers upload a new version of software.
2) *Traffic management:* Drivers can receive traffic status reports (e.g., congestion) from VCs.
3) *Road condition sharing:* Road conditions such as flooding areas and black ice on the roadway can be shared in VCs. Drivers will be alerted if there are serious road conditions.
4) *Accident alerts at intersections:* Under demanding driving conditions such as fog, heavy storm, snow, and black ice, drivers can order this service to alert them of possible accidents at intersections. Infrastructure, e.g., a tall building, can include high-precision radar to detect car accidents. This infrastructure will cover the whole intersection and frequently scan the intersection. An intelligent algorithm will be applied to each scan result to predict the possibility of accidents.
5) *Safety applications:* Applications related to life-critical scenarios such as collision avoidance and adaptive cruise control require strong security protection, even from surrounding environmental security threats.
6) *Intelligent parking management:* Vehicles will be able to book a parking spot using the VC. All the parking information will be available on clouds without central control. Requests from different physical places can be transferred to the most desired parking lots.
7) *Planned evacuations:* In some disasters such as a hurricanes and tsunamis, VCs will be instrumental in organized evacuations.

IV. Analyzing Security in a Vehicular Cloud

In this section, we introduce a set of security analyses that are specially associated with VCs.

A. Security and Privacy Attacks in VC

1) Attacker Model: Traditional security systems are often designed to prevent attackers from entering the system. However, security systems in the VC have a much harder time keeping attackers at bay, because multiple service users with high mobility can share the same physical infrastructure. In the VC environment, an attacker can equally share the same physical machine/infrastructure as their targets, although both of them are assigned to different VMs. To this point, attackers can have more advantages than the attackers on traditional systems. In addition, the attackers are physically moving from place to place as vehicles are mobile nodes. It is much harder to locate the attackers.

The main targets of an attacker are given as follows:

1) confidentiality, such as identities of other users, valuable data and documents stored on the VC, and the location of the VMs, where the target's services are executing;
2) integrity, such as valuable data and documents stored on the VC, executable code, and result on the VC;
3) availability, such as physical machines and resources, privileges, services, and applications.

One possible form of attack is given below:

1) Find the geographic location of the target vehicle and physically move close the target machine;
2) Narrow down the possible areas where the target user's services are executing by mapping the topology of VC;
3) Launch multiple experimental accesses to the cloud, and find out if the target user is currently on the same VM;
4) Request the services on the same VM where the target user is on;
5) Use system leakage to obtain higher privilege to collect the assets [23].

Due to the features of the VC, there are several challenges for attackers as well. High mobility of vehicles is like a double-edged sword. It makes it hard for attackers to harm a specific target vehicle. First, the vehicle's access of each virtual machine can be transitory as vehicles constantly move from one district to another one, if each district is associated with a virtual machine. Additionally, attackers need to locate on which machine/infrastructure a specific target is located because all users in the VC are distributed on virtual machines. However, it is possible to locate the co-residence of other users. Experiments have been done to catch and compare the memory of processors, and users can find co-residence in the same physical machine [23]. Third, the attackers must be physically co-located with the target user on the same physical machines. This will require attackers to be physically present at the same region with the target vehicles or shadow with the target vehicles at the same speed. These challenges make attacking extremely difficult because coexistence is hard to achieve and is temporary. Finally, the attackers have to collect valuable information with certain privileges or with security tokens.

2) Threats: The threats in the VC can be classified using STRIDE [26]: a system developed by Microsoft for classifying computer security threats. The threat categories are given here.

1) *Spoofing user identity:* The attackers pretend to be another user to obtain data and illegitimate advantages. One classic example is the "man-in-the-middle attack," in which the attackers pretend to be Bob when communicating with Alice and pretend to be Alice when

communicating with Bob. Both Alice and Bob will send decryptable messages to the attackers.

2) *Tampering:* The attackers alter data and modify and forge information.
3) *Repudiation:* The attackers manipulate or forge the identification of new data, actions, and operations.
4) *Information disclosure:* The attackers uncover personally identifiable information such as identities, medical, legality, finance, political, residence and geographic records, biological traits, and ethnicity.
5) *Denial of Service:* The attackers mount attacks that consume system resources and make the resources unavailable to the intended users.
6) *Elevation of privilege:* The attackers exploit a bug, system leakage, design flaw, or configuration mistake in an operating system or software application to obtain elevated access privilege to protected resources or data that are normally protected from normal users.

B. Authentication of High-Mobility Nodes

Security authentication in the VC includes verifying user identity and message integrity. To conduct authentication, there are some metrics that can be adopted [27].

1) Ownership: A user owns some unique identity (e.g., identity card, security token, and software token).
2) Knowledge: A user knows some unique things [e.g., passwords, personal identification number and human challenge response (i.e., security questions)].
3) Biometrics: These include the signature, face, voice, and fingerprint.

However, it is challenging to authenticate vehicles due to high mobility. First, high mobility makes it hard to authenticate messages with a location context. For example, accident alert message associated with locations and events at a specified time are hard to verify because the locations of vehicles are constantly changing. Second, high mobility and a short transmission range may result in the recipient being out of reach. It is likely that a vehicle at the border of access point can change its access point when the authentication message is transmitted back. Third, the security token (security key pairs) is hard update. Some vehicles can even park for years without starting a single time. These situations will make the updating tasks of the security token significantly difficult.

In addition, it is challenging to authenticate a vehicle's or driver's identity in the VC. To protect privacy, these identities are often replaced by pseudonyms. The authentication of identity can be complex and makes Sybil attacks possible [28].

C. Establishing Trust Relationships

Trust is one of the key factors in any secure system. A trust relationship can exist in several ways. The network service providers and the vehicle drivers have access to trust. There will be a large number of government agents, e.g., the Department of Motor Vehicles (DMV) and the Bureau of Motor Vehicles (BMV) are trusted organizations. The relationship between the BMV and vehicle drivers is identity uniqueness and legitimacy.

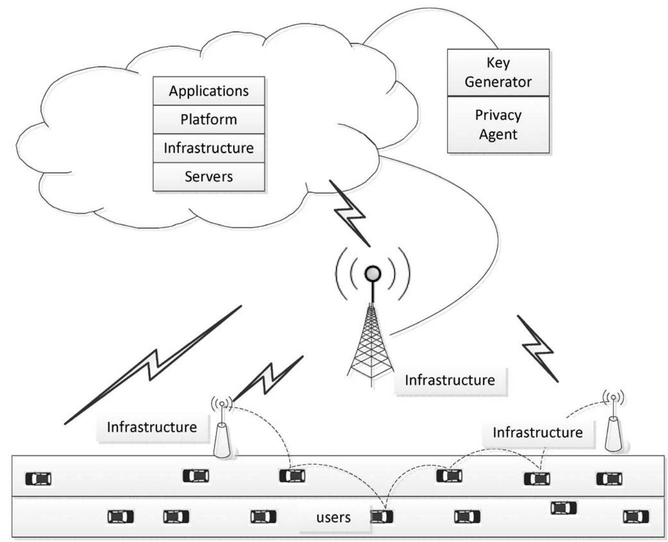

Fig. 2. Vehicles often communicate through multihop routing. A request response will include multiple participants, including users, infrastructure, servers, platform, application, and key generator and privacy agent.

However, the large population of vehicles creates challenges to building trust relationships to all the vehicles at any time. There will be occasional exceptions. In addition, drivers are increasingly concerned about their privacy. Tracking vehicles/drivers will cause worries in most cases. As a result, pseudonyms are often applied to vehicles. On the other hand, a certain level of trust of identity is needed. Some applications such as accident reliability investigation by law enforcement or insurance companies require the driver's identity to be responsible for accidents. Therefore, we assume that a low level of trust relationship exists in VANETs. To obtain a high-level trust relationship, the security scheme discussed in Section IV needs to be executed.

In VCs, it is far more challenging to build trust relationships than in vehicular networks and conventional cloud computing. Fig. 2 shows an example of multiple participants in a VC. The VC is often based on DSRC. Many applications need multihop routing, with multiple nodes involved in communication. Therefore, the VC has inherited the challenge of establishing trust relationships among multiple vehicles, roadside infrastructure, service providers, network channels, and even the secret key generator.

In this paper, we assume that the VC cloud infrastructure is trusted, the VC service providers are trusted, the vast majority of VC users are trustworthy, and the attackers have the same privileges as normal users.

D. Location Validation and Pseudonymization

Most, if not all, VC applications rely on accurate location information. Therefore, location information must be validated. There are two approaches to validate location information: active and passive. Vehicles or infrastructure with radar (or camera, etc.) can perform active location validation. Radar input can be used to validate location information. Vehicles or infrastructure without radar, or in a situation where radar detection is not possible, can validate location information by applying statistical methods [9], [29].

A vehicle's identity is often tangled with owner's identity. Because of legal and insurance issues, a vehicle's unique identity (such as vehicle identity number, Internet Protocol address, and hostname) is often linked to the owner's identity. Therefore, tracking a vehicle can often invade its owner's privacy. To protect privacy, one can replace vehicular identity by a pseudonym. The real identity can only be discovered by the Pseudonymization Service Center, which is a secured and trusted entity. The pseudonym is subject to timeout. After expiration, a new pseudonym will be assigned. Digital license plates (DLPs) or electronic license plates, which are a wireless device periodically broadcasting a unique identity string, have been proposed. Temporary public keys as DLPs can protect privacy and can be broadcast [11].

E. Scalability

Security schemes for VCs must be scalable to handle a dynamically changing number of vehicles. Security schemes must handle not only regular traffic but special traffic as well, e.g., the large volume of traffic caused by special events (e.g., football games, air shows, etc.)

The dynamics of traffic produces dynamic demands on security. For example, imagine a downtown area with several supermarkets and stores that take orders from vehicles in traffic, complete with credit card information. To protect credit card information, comprehensive cryptographic algorithms must be applied. However, the comprehensive algorithms decrease the efficiency of communication response time. Therefore, better algorithms and, perhaps, less comprehensive security schemes are needed to speed up the response time.

F. Single-User Interface

Single-user access interface is another challenge to VCs. When the number of service accesses in a cloud increases, the number of VMs that provide the service will increase to guarantee quality of service. More VMs will be created and assigned. With the increase in VMs, security concerns grow as well. When the number of service accesses decreases, the number of VMs that provide the service will decrease to improve resource utilization. Some VMs will be destroyed and recycled. These procedures are transparent to vehicles. Vehicles only see one access interface and do not need to know the changing of VMs. To achieve scalability, a simple solution is to clone and expand the service in a different cloud. However, a single interface obviously makes scalability even more difficult.

G. Heterogeneous Network Nodes

Conventional cloud computing and fixed networks often have homogeneous end users. As it turns out, vehicles have a large array of (sometimes) vastly different onboard devices. Some high-end vehicles have several advanced devices, including a Global Positioning System (GPS) receiver, one or more wireless transceivers, and onboard radar devices. In contrast, some economy models have only a wireless transceiver. Some other vehicles have different combinations of GPS receivers, wireless transceivers, and radar. Different vehicle models have different device capabilities such as speed of processor, volume of memory, and storage. These heterogeneous vehicles as network nodes create difficulties to adapting security strategies. For example, PKI encryption and decryption algorithms will require vehicles to meet certain hardware conditions.

H. VC Messages

1) Safety Messages: The initial motivation of VANET was the dissemination of traffic safety messages. Based on the emergency level, there are three types of safety messages.

1) Level one: public traffic condition information. Vehicles exchange traffic information (e.g., traffic jam) that indirectly affects other vehicles' safety, e.g., a traffic jam increases the likelihood of accidents. This type of message is not sensitive to communication delay, but privacy needs to be protected.
2) Level two: cooperative safety messages. Vehicles exchange messages in cooperative accident avoidance applications. These messages are often time critical, and privacy needs to be protected.
3) Level three: liability messages. After accidents happen, there will be liability messages generated by law enforcement authorities. These messages contain important evidence for liability claims and are bonded by a certain time range. Privacy information is naturally protected.

A common format of safety messages is timestamp, geographic location, speed, percentage of speed change since the last message, direction, acceleration, and percentage of acceleration change since last message. The safety message will append information such as public traffic condition and accidents. The appended message can help determine liability. Driver identity information is not necessary to be part of the safety message. Pseudonyms can be applied to protect the driver's identity. The signature of the safety message can be described as follows: Following the ElGamal signature scheme [30], we define three parameters.

1) H: a collision-free hash function;
2) p: a large prime number that will ensure that computing discrete logarithms modulo p is very difficult;
3) $g < p$: a randomly chosen generator out of a multiplicative group of integers modulo p.

Each vehicle has long-term PKI public/private key pairs:

- private key: S;
- public key: $\langle g, p, T\rangle$, where $T = g^S \bmod p$.

It should be noted that a message m can be combined as $m|T$, where T is the timestamp. The timestamp can ensure the freshness of the message. For each message m to be signed, three steps are followed.

1) Generate a per-message public/private key pair of S_m (private) and $T_m = g^{S_m} \bmod p$ (public).
2) Compute the message digest $d_m = H(m|T_m)$ and the message signature $X = S_m + d_m S \bmod (p-1)$, where mod is the modulo operation and $|$ is the concatenation operator.
3) Send m, T_m, and X.

To verify the message, three steps are followed.

1) Compute the message digest $d_m = H(m|T_m)$.
2) Compute $Y_1 = g^X$ and $Y_2 = T_m T^{d_m}$.
3) Compare $Y_1 = Y_2$. If $Y_1 = Y_2$, then the signature is correct.

The reason is

$$Y_1 = g^X = g^{S_m + d_m S} = g^{S_m} g^{d_m S} = T_m g^{S d_m} = T_m T^{d_m} = Y2.$$

2) Confidential Messages: To ensure the confidentiality of a sensitive message, the message will be both signed and encrypted. Suppose that vehicle A sends a sensitive message m to vehicle B. Each vehicle has its own PKI public/private key pairs. Thinking of the overhead of PKI processing time, we can adapt a symmetric encryption algorithm. However, to exchange a secret key, we still need to use PKI support. The handshake of exchanging the secret key is defined as follows:

$$A \rightarrow B : B|K|T_{\text{pub}_B}, SigB|K|T_{\text{pri}_A}$$

where A and B are the identities of vehicles A and B, respectively; K is the secret key shared by A and B; m is the sensitive message; T is the timestamp; pub_B is the public key of B; and pri_A is the private key of A.

Once A and B both know the secret key K, they can communicate by using a well-known message authentication code (MAC or HMAC). Hashing the sensitive message is done as follows:

$$A \leftrightarrow B : m, \text{MAC}_K m.$$

There are potential problems with this approach. As a drawback of symmetric encryption, nonrepudiation (i.e., integrity and origin of data) cannot be ensured, although the likelihood of data being surreptitiously changed is extremely low. This is a compromise solution between efficiency and security. To achieve a higher level of security for sensitive messages, one can apply active security mechanisms [9] or adopt PKI encryption at the cost of losing a certain amount of efficiency. In multihop networks, the key handshake in this scheme does not scale well in zero-infrastructure VANET, but it can scale well with the aid of roadside infrastructure.

I. Key Management

1) Key Assignment and Rekeying: In VANETs, some organizations can serve as CAs: governmental transportation authorities, vehicle manufacturers, or nonprofit organizations.

Initially, a vehicle will receive a key pair from the manufacturer or some governmental authority. Key assignment is on the basis of a unique ID with a certain expiration time. Upon expiration, the key pair has to be renewed at the local DMV/BMV. The renewal/expiration period can be the same period of vehicular state inspection, e.g., mandatory annual state inspection in many U.S. states.

2) Key Verification: To verify key pairs, we assume that every vehicle trusts CAs and that CAs are tamper-proof. Key validation can be done at the CAs or sub-CAs. Let pub_i of vehicle i be the public key issued by a CA j, i.e., CA_j. Vehicle i will have a certificate $\text{cert}_i[\text{pub}_i]$ assigned by CA_j when CA_j assigns the public key. The process of validating public key will compute the following certificate at CA_j:

$$\text{cert}_i[\text{pub}_i] = \text{pub}_i|\text{sig}_{\text{pri}_{\text{CA}_j}}(\text{pub}_i|\text{ID}_{\text{CA}_j})$$

where pri_{CA_j} is the private key of CA_j, and ID_{CA_j} is the identity of CA_j. The idea is to sign the special message $\text{pub}_i|\text{ID}_{\text{CA}_j}$ using the private key of CA_j. The digital signature algorithm has been discussed in Section IV-H1.

3) Key Revocation: Key revocation is an important and effective way to prevent attacks. There are certain cases when key pairs will be exposed to attackers. It is obvious that an exposed key pair needs to be disabled. One of the advantages of PKI is that PKI can revoke a key pair. Vehicles will be aware that the exposed key pair has been revoked and refuse to communicate with vehicles with invalid key pairs. PKI uses certificate revocation lists (CRLs) to revoke keys. CRLs include a list of the most recently revoked certificates and are instantly distributed to vehicles. In VANETs, the infrastructure can serve as CRL distributors.

The CAs can revoke key pairs by using onboard tamper-proof devices. Suppose that CAs want to revoke the key pairs of vehicle V. CAs will send out the revoke message signed by public key of V to the tamper-proof devices. After receiving this revoking message, the tamper-proof device will validate the message and revoke the key pairs. The tamper-proof device will also send back an ACK to the CA to confirm the operation. To improve communication between V and CA, the vehicle's location is retrieved to select the closest CA. If the latest vehicle location failed to be retrieved, the last location will be used to select the closest CA. In this case, the CA will use a broadcasting message to revoke the key pairs. The broadcasting message can be sent out by using several media such as FM, Internet, and satellite.

To avoid attackers reporting other vehicles to CA to revoke the key pairs of other vehicles, revocation will be triggered by a certain number of neighboring vehicles. There is another risk that attackers can launch planned attacks. For example, several attackers can surround a well-behaved vehicle and report the well-behaved vehicle as a misbehaving vehicle. Prevention of this risk is very challenging. Due to the dynamics of traffic, it is costly to launch such an attack. One possible solution is to build behavior history records and credit the past behavior into values, just like the bank credit system. A similar solution has been discussed as Map History [9].

V. Research Approach

In this section, we offer a first attempt to addressing several of the challenges previously discussed. We begin by describing the two VC models, i.e., infrastructure- and ad-hoc-based models. We then demonstrate algorithms to enhance authentication of high-mobility vehicles, configure customized security schemes, and improve scalability of security schemes.

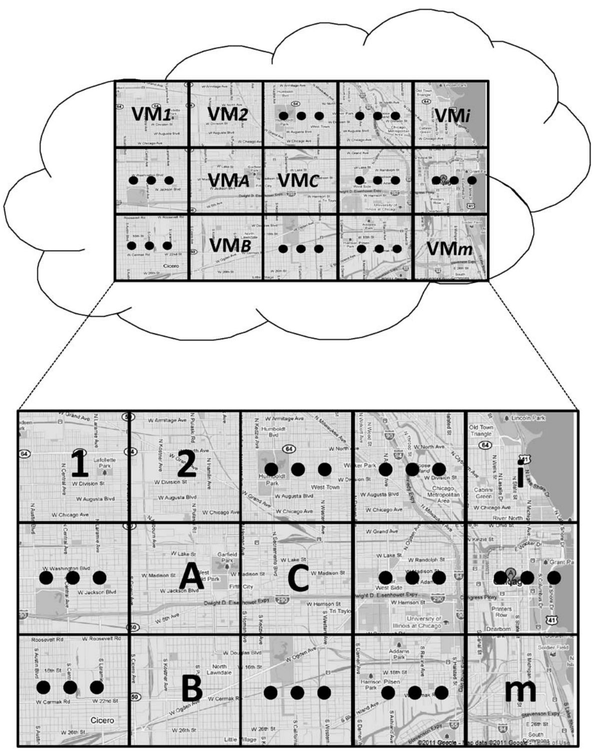

Fig. 3. Downtown area partitioned into cells, each mapped to a virtual machine.

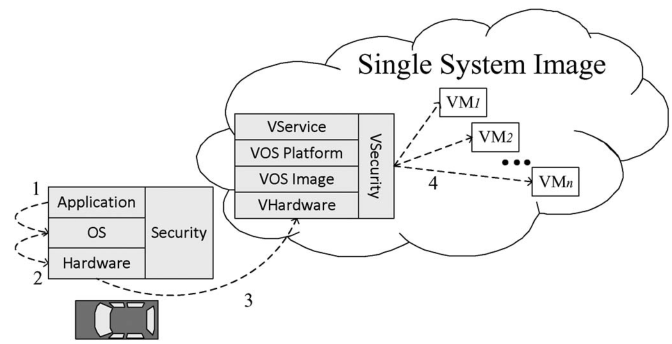

Fig. 4. Vehicle node in a cell can communicate with a virtual machine that is responsible for the cell.

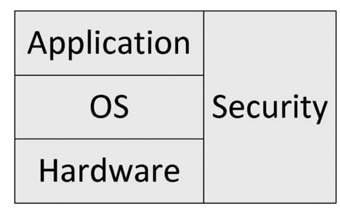

Fig. 5. Vehicle node image is located on each individual vehicle.

A. The Cloud Model

The cloud in this proposal is associated with a number of grids. A city or a traffic area is partitioned into grids. The grid size is predefined, e.g., 700 m^2 and with two GPS coordinates. The grid of a city is shown in Fig. 3. Each cell is associated with a virtual machine in the cloud. The virtual machine can dynamically request resources from cloud. For example, when the grid is congested, the corresponding virtual machine will request more communicating, storage, and computing resources. The cloud will be able to borrow these resources from the idle virtual machine, which is associated with sparse traffic grid. Therefore, the traffic of the whole city can be mapped to the cloud.

This cloud model provides high capability in customizing cloud services and the security scheme. For example, a downtown area is often queried about vacant parking spots and congestion status. The corresponding virtual machine can be specially configured and optimized in the smart parking and congestion control services. At a busy intersection, a collision-warning service can be specialized and optimized in the virtual machine. A possible solution is to collect and sort all the vehicles' mobility information at the intersection. When vehicles are too close to each other by considering the headway distance and relative speed, the vehicles will receive an alarm from the cloud. Even cheaper cars that have no radar cruise control system can get benefits from the cloud collision warning system.

What distinguishes vehicles from standard nodes in a conventional cloud is *autonomy* and *mobility*. Indeed, large numbers of vehicles spend substantial time on the road and may be involved in dynamically changing situations; we argue that, in such situations, the vehicles have the potential to cooperatively solve problems that would take a centralized system an inordinate amount of time, rendering the solution useless [2]. Vehicles automatically form a cloud by connecting virtual cells, which can be a group of vehicles. Each virtual cell is associated with a virtual machine in which vehicles rent or contribute their spare computing, storage, and sensing resource. The group of vehicles moves at almost the same speed. Since vehicles are cloud constructors and cloud users, all vehicles inside a cell can directly receive packets from each other. A cell leader can be elected to communicate with other clouds [9].

1) Virtual Machines of VCs: This objective concerns how a cloud is formed and how the service can be provided. We first consider the basic modules of the VC and then introduce the process of a service request and response.

The communication between a vehicle and the cloud is through a unique entry. The cloud provides a single system image to each individual virtual machine shown as Fig. 4. Each vehicle has a node image, which includes hardware drivers, operating system image, security system, and applications, as shown in Fig. 5. When the applications of the vehicle send a request to the cloud, the request will be forwarded to the operating system and, then, the hardware (network driver). The request will be sent by the wireless network and received by the cloud single system image. The allocator of the cloud will locate which virtual machine should be responsible for the request and forward the request to the virtual machine. If the request needs to access other virtual machines, e.g., to check the traffic congestion status of a city in a remote state, the virtual machine can communicate with other virtual machines as well.

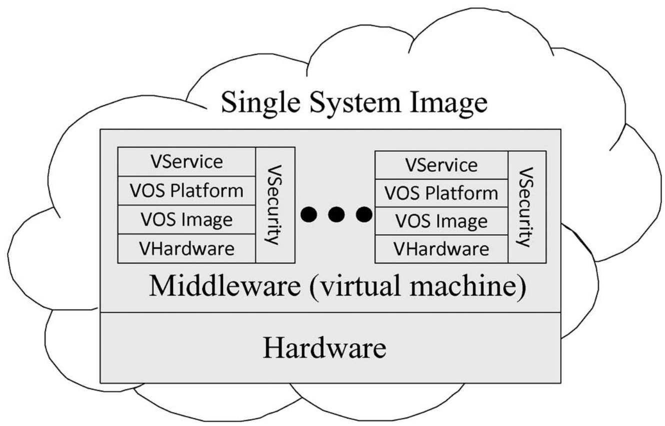

Fig. 6. Cloud provides a single system image and is composed by a number of virtual machines.

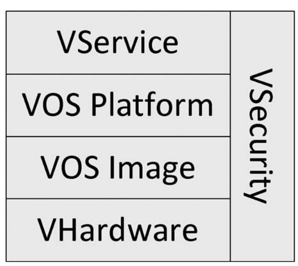

Fig. 7. Single virtual machine located in the cloud.

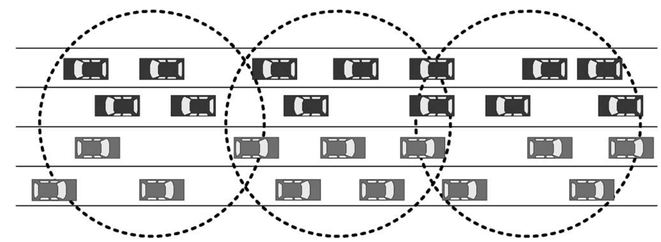

Fig. 8. Trust relationship in AVCs can be built on the basis of a group of vehicles. The behavior of a vehicle can be monitored by all members.

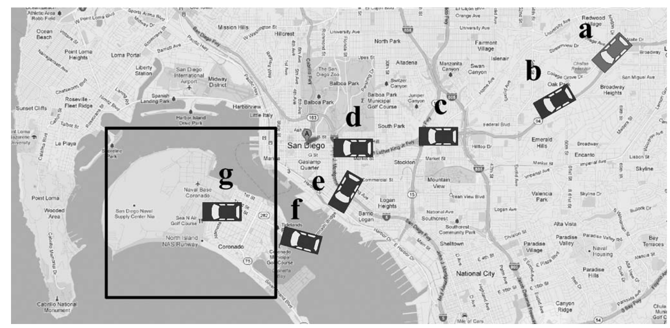

Fig. 9. Geographic location-based security mechanism. The shaded square is the naval base. Only the vehicles in the shaded rectangle region (i.e., vehicle $\boldsymbol{g}$ can decrypt and access the received ciphertext sent by vehicle $\boldsymbol{a}$).

The VC is a single system image composed of a number of virtual machines. A single image can be created by a layer of middleware between the hardware manager system and a number of virtual machines, as shown in Fig. 6. The middleware is a cloud operating system and a platform to allocate a large number of virtual machines. Each virtual machine is composed of virtual hardware, virtual operating system image, virtual operating system platform, virtual security system, and virtual services, as shown in Fig. 7. The virtual hardware is composed of several real computers that virtually act as real hardware and provide the interface of the hardware. The virtual operating system image can be any current operating system, such as Linux/Unix or Windows. The virtual operating system platform includes not only the operating system but system applications such as web server and databases. The virtual security system is a set of complete security solutions, including hardware and software. The customized security protocols can be configured and replaced in this module. The virtual services are actual services that are configured for the related traffic area/grid.

B. Securing VCs

1) Trust Relationship: For infrastructure-based VC, trust relationships can be built by infrastructures that are constructed by authorities such as BMV/DMV or other transportation agencies. Infrastructure will be authenticated and assigned with security key pairs. Infrastructure stores the key pairs in tamper-proof devices. As shown in Fig. 2, vehicles communicate with infrastructure as access point to the VC. The infrastructure is sufficiently capable to handle large numbers of accesses in its transmission range. The scalability of trust relationships can be achieved because the infrastructure is connected to each other by fixed networks.

For AVCs, trust relationships can be built as well. A cell leader can be elected to represent the members in the cell to communicate with other cells. For security reasons, the cell leader is monitored by its neighbors. When the leader sends and receives aggregated position packets, all the members in the cell will compare the positions in the packets based on their knowledge. By remaining silent, they confirm that the packets have not been altered. Otherwise, they broadcast protest packets against the leader. The other neighbors will put the leader and the protestor vehicle into the question table after receiving the protest packet. Then, the opinion of the other neighbors is taken into account. If the majority of vehicles regard the leader as malicious, the record of the leader is moved to the distrust table, as discussed by Yan *et al.* [9]. Otherwise, the records sent by the leader are placed in the trust table (see Fig. 8).

2) Authentication and Confidentiality: To provide authentication and confidentiality, we propose a geographic location-based security mechanism to ensure physical security on top of conventional methods. Messages are encrypted with a geographic location key that specifies a decryption region. This provides *physical* security because a vehicle has to be physically present in the decryption region to decrypt ciphertext encrypted with this geographic location key. As an example, Fig. 9 shows a shaded square that is a location-based security region. Sender vehicle a specifies the region, creates the location key, encrypts the message, and sends ciphertext to vehicles in this region. Vehicles outside this region such as b, c, d, and e cannot decrypt the message. Only vehicle f can decrypt the message because it is physically inside the decryption region. Since the decryption region can be dynamically specified, attacks are extremely expensive and difficult to mount.

C. *Configuring Security Strategies*

It is important to allow the VC to dynamically configure the security protocols and to independently replace security strategies. We will start with the configuration of security protocols and then describe an intelligent task management method.

1) More Vehicles Involved, More Secure Cloud Needed: The cloud will provide vehicles a single system image that is transparent of details of security scheme changes. As vehicles are dynamically moving in and out of a cell, the security protocols of a cell in its virtual machine need to be dynamically adjusted. We observe the fact that the more vehicles are involved, the more secure and the stricter a protocol should be. Similar facts can be found in daily life. Airports are often crowded, and security is often stricter than that in many other places. Events such as football games, auto races, and air shows often attract more people, as well as more policemen who patrol the area more often to ensure the security of attendees.

Therefore, it is important to know the expected volume of vehicles at any time to dynamically switch security protocols. We are interested in the following problem to evaluate the expected number of vehicles at any given time. Consider a cell with finite capacity N. At time $t = 0$, the cell contains $n_0 \geq 0$ cars. After that, cars arrive and depart at time-dependent rates, as described next. If the cell contains k, $(0 \leq k \leq N)$ cars at time t, then the car arrival rate $\alpha_k(t)$ is

$$\alpha_k(t) = \frac{N-k}{N}\lambda(t)$$

and the car departure rate $\beta_k(t)$ is

$$\beta_k(t) = k\mu(t)$$

where, for all $t \geq 0$, $\lambda(t)$ and $\mu(t)$ are *integrable* on $[0, t]$. It is worth noting that both $\alpha_k(t)$ and $\beta_k(t)$ are functions of both t and k. In particular, it may well be the case that, for $t_1 \neq t_2$, $\alpha_k(t_1) \neq \alpha_k(t_2)$, and similarly for $\beta_k(t_1)$ and $\beta_k(t_2)$, giving a mathematical expression to the fact that, at different times of the day, for example, the departure rate depends on not only the number of cars present in the cell but on the time-dependent factors as well.

Consider the counting process $\{X(t)|t \geq 0\}$ of continuous parameter t, where, for every positive integer k, $(1 \leq k \leq N)$, the event $\{X(t) = k\}$ occurs if the cell contains k, cars at time t. We let $P_k(t)$ denote the probability that the event $\{X(t) = k\}$ occurs. In other words

$$P_k(t) = \Pr\left[\{X(t) = k\}\right].$$

In addition to $P_k(t)$, of interest are the expected number $E[X(t)]$ and the variance $Var[X(t)]$ of the number of cars in the cell at time $t > 0$, as well as the limiting behavior of these parameters as $t \to \infty$, whenever such a limit exists and/or makes sense.

To make the mathematical derivations more manageable, we set $P_k(t) = 0$ for $k < 0$ and $k > N$. Thus, $P_k(t)$ is well defined for all integers $k \in (-\infty, \infty)$ and for all $t \geq 0$. In particular, the assumption about the cell containing n_0 cars at $t = 0$ translates into $P_k(0) = 1$ if $k = n_0$ and 0 otherwise.

Let t, $(t \geq 0)$, be arbitrary, and let h be sufficiently small such that, in the time interval $[t, t+h]$, the probability of two or more arrivals or departures, or of a simultaneous arrival and departure, is $o(h)$. With h chosen as stated, the probability $P_k(t+h)$ that the cell contains k, $(0 \leq k \leq N)$ cars at time $t+h$ has three components.

1) $P_k(t)[1 - h(N - k/N)\lambda(t) - kh\mu(t) + o(h)]$.
2) $P_{k-1}(t)[h(N - k + 1/N)\lambda(t) + o(h)]$.
3) $P_{k+1}(t)[(k+1)h\mu(t) + o(h)]$.

Here, by assumption, $P_k = 0$ for $k < 0$ and $k > N$.

The expression of probability $P_k(t)$ can be derived by

$$P_k(t) = 1 - e^{-h(t)} \int_0^t \mu(u) e^{h(u)} \mathrm{d}u$$

where

$$h(x) = \int_0^x \left[\frac{\lambda(s)}{N} + \mu(s)\right] \mathrm{d}s.$$

We can write the linearity of expectation as

$$E\left[X(t)\right] = e^{-h(t)} \left[n_0 + \int_0^t \lambda(u) e^{h(u)} \mathrm{d}u\right].$$

D. *Enhancing Scalability of Security Schemes*

When vehicle population increases in a certain area, not only the scalability of the VC but also the scalability of security schemes becomes a tough problem. In our cloud model, the scalability of the security scheme can be enhanced by a virtual machine division algorithm, a highly scalable algorithm. When the number of access of a virtual machine grows sufficiently large, compared to an empirical threshold, the virtual machines (as a super-VM) will divide itself into multiple subvirtual machines (as sub-VMs). Each virtual machine will obtain the same amount of resources as the original super VM. The middleware of the super VM can randomly forward request to subvirtual machines to load balance. The middleware of the super VM also caches the most recently accessed and frequent information. It caches and executes information such as frequently asked questions (FAQs) and answers. If access from a vehicle hits the FAQ, the middleware directly sends back the answer. If the access misses the FAQ, the middleware then forwards access to a relatively idle VM. This can further reduce the workload of sub-VMs (see Fig. 10).

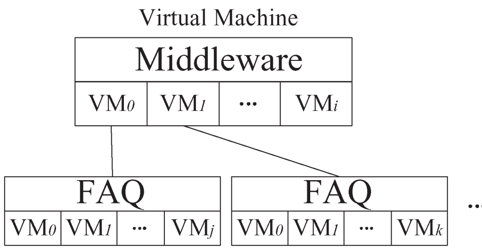

Fig. 10. Virtual machine can be divided into multilayers of VMs. Each layer is composed by multiple VMs. The middleware can also be deployed with a cache of frequently accessed information.

VI. Concluding Remarks

In this paper, we have addressed the security challenges of a novel perspective of VANETs, i.e., taking VANETs to clouds. We have first introduced the security and privacy challenges that VC computing networks have to face, and we have also addressed possible security solutions. Although some of the solutions can leverage existing security techniques, there are many unique challenges. For example, attackers can physically locate on the same cloud server. The vehicles have high mobility, and the communication is inherently unstable and intermittent. We have provided a directional security scheme to show an appropriate security architecture that handles several, not all, challenges in VCs. In future work, we will investigate the brand-new area and design solutions for each individual challenge. Many applications can be developed on VCs. As future work, a specific application will need to analyze and provide security solutions.

Extensive work of the security and privacy in VCs will become a complex system and need a systematic and synthetic way to implement intelligent transportation systems [32], [33]. Only with joint efforts and close cooperation among different organizations such as law enforcement, government, the automobile industry, and academics can the VC computing networks provide solid and feasible security and privacy solutions.

Acknowledgment

The authors would like to thank three anonymous referees for their constructive comments and criticism that helped us improve the organization of this paper.

References

[1] S. Arif, S. Olariu, J. Wang, G. Yan, W. Yang, and I. Khalil, "Datacenter at the airport: Reasoning about time-dependent parking lot occupancy," *IEEE Trans. Parallel Distrib. Syst.*, 2012, [Online]. Available: https://csdl2.computer.org/csdl/trans/td/preprint/ttd2012990021-abs.html, to be published.

[2] S. Olariu, M. Eltoweissy, and M. Younis, "Toward autonomous vehicular clouds," *ICST Trans. Mobile Commun. Comput.*, vol. 11, no. 7–9, pp. 1–11, Jul.–Sep. 2011.

[3] S. Olariu, I. Khalil, and M. Abuelela, "Taking VANET to the clouds," *Int. J. Pervasive Comput. Commun.*, vol. 7, no. 1, pp. 7–21, 2011.

[4] L. Li, J. Song, F.-Y. Wang, W. Niehsen, and N. Zheng, "IVS 05: New developments and research trends for intelligent vehicles," *IEEE Intell. Syst.*, vol. 20, no. 4, pp. 10–14, Jul./Aug. 2005.

[5] G. Yan and S. Olariu, "A probabilistic analysis of link duration in vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1227–1236, Dec. 2011.

[6] H. Xie, L. Kulik, and E. Tanin, "Privacy-aware traffic monitoring," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 61–70, Mar. 2010.

[7] D. Huang, S. Misra, G. Xue, and M. Verma, "PACP: An efficient pseudonymous authentication based conditional privacy protocol for vanets," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 736–746, Sep. 2011.

[8] R. Hasan, *Cloud Security*. [Online]. Available: www.cs.jhu.edu/~ragib; http://www.ragibhasan.com/research/cloudsec.html AQ1

[9] G. Yan, S. Olariu, and M. C. Weigle, "Providing VANET security through active position detection," *Comput. Commun.*, vol. 31, no. 12, pp. 2883–2897, Jul. 2008, Special Issue on Mobility Protocols for ITS/VANET.

[10] G. Yan, S. Olariu, and M. Weigle, "Providing location security in vehicular ad hoc networks," *IEEE Wireless Commun.*, vol. 16, no. 6, pp. 48–55, Dec. 2009.

[11] J. Sun, C. Zhang, Y. Zhang, and Y. M. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 9, pp. 1227–1239, Sep. 2010.

[12] G. Yan and S. Olariu, "An efficient geographic location-based security mechanism for vehicular ad hoc networks," in *Proc. IEEE Int. Symp. TSP*, Macau SAR, China, Oct. 2009, pp. 804–809.

[13] A. Friedman and D. West, "Privacy and security in cloud computing," *Center for Technology Innovation: Issues in Technology Innovation*, no. 3, pp. 1–11, Oct. 2010.

[14] J. A. Blackley, J. Peltier, and T. R. Peltier, *Information Security Fundamentals*. New York: Auerbach, 2004.

[15] N. Santos, K. P. Gummadi, and R. Rodrigues, "Toward trusted cloud computing," in *Proc. HotCloud*, Jun. 2009.

[16] T. Garfinkel, B. Pfaff, J. Chow, M. Rosenblum, and D. B. Terra, "Virtual machine-based platform for trusted computing," in *Proc. ACM SOSP*, 2003, pp. 193–206.

[17] S. Berger, R. Cáceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn, "VTPM: Virtualizing the trusted platform module," in *Proc. 15th Conf. USENIX Sec. Symp.*, Berkeley, CA, 2006, pp. 305–320.

[18] D. G. Murray, G. Milos, and S. Hand, "Improving XEN security through disaggregation," in *Proc. 4th ACM SIGPLAN/SIGOPS Int. Conf. VEE*, New York, 2008, pp. 151–160.

[19] F. J. Krautheim, "Private virtual infrastructure for cloud computing," in *Proc. Conf. Hot Topics Cloud Comput.*, 2009, pp. 1–5.

[20] M. Jensen, J. Schwenk, N. Gruschka, and L. L. Iacono, "On technical security issues in cloud computing," in *Proc. IEEE Int. Conf. Cloud Comput.*, 2009, pp. 109–116.

[21] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *Proc. IEEE INFOCOM*, San Diego, CA, 2010, pp. 1–9.

[22] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proc. 14th ESORICS*, 2009, pp. 355–370.

[23] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds," in *Proc. 16th ACM Conf. CCS*, 2009, pp. 199–212.

[24] SIRIT-Technologies, White paper. DSRC technology and the DSRC industry consortium (DIC) prototype team.

[25] D. Wen, G. Yan, N. Zheng, L. Shen, and L. Li, "Toward cognitive vehicles," *IEEE Intell. Syst. Mag.*, vol. 26, no. 3, pp. 76–80, May–Jun. 2011.

[26] Microsoft, The stride threat model. [Online]. Available: http://msdn.microsoft.com

[27] Fed. Fin. Inst. Examination Council, Authentication in an Internet banking environment2009. [Online]. Available: http://www.ffiec.gov/pdf/authentication_guidance.pdf

[28] J. Douceur, "The sybil attack," in *Proc. Rev. Papers 1st Int. Workshop Peer-to-Peer Syst.*, 2002, vol. 2429, pp. 251–260.

[29] G. Yan, W. Yang, E. F. Shaner, and D. B. Rawat, "Intrusion-tolerant location information services in intelligent vehicular networks," *Commun. Comput. Inf. Sci.*,, vol. 135, pp. 699–705, 2011.

[30] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inf. Theory*, vol. IT-31, no. 4, pp. 469–472, Jul. 1985.

[31] Nat. Inst. Stand. Technol., Gaithersburg, MD, The NIST Definition of Cloud Computing, 2011.

[32] J. Li, S. Tang, X. Wang, W. Duan, and F.-Y. Wang, "Growing artificial transportation systems: A rule-based iterative design process," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 322–332, Jun. 2011.

[33] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.

Gongjun Yan received the Ph.D. degree from Old Dominion University, Norfolk, VA, in 2010.

He is an Assistant Professor of informatics with Indiana University Kokomo. His research interests include information security and privacy, intelligent vehicles, vehicular ad hoc networks, and wireless communications.

Ding Wen is currently a Professor with the Center for Military Computational Experiments and Parallel Systems Technology, National University of Defense Technology Changsha, Hunan, China. His research interests include intelligent systems and unmanned systems.

Stephan Olariu received the Ph.D. degree in computer science from McGill University, Montreal, QC, Canada, in 1986.

He is currently a Professor of computer science with Old Dominion University, Norfolk, VA. He has held many different roles and responsibilities as a member of numerous organizations and teams. Much of his experience has involved the design and implementation of robust protocols for wireless networks and, particularly, sensor networks and their applications. He is currently applying mathematical modeling and analytical frameworks to the resolution of problems ranging from securing communications to predicting the behavior of complex systems and evaluating the performance of wireless networks.

Michele C. Weigle received the Ph.D. degree in computer science from the University of North Carolina, Chapel Hill, in 2003.

She is currently an Associate Professor of computer science with Old Dominion University, Norfolk, VA. Her research interests include vehicular networks, mobile ad-hoc networks, wireless networking, sensor networks, network simulation and modeling, and Internet congestion control.

AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

AQ1 = Please confirm which of the two provided web address for reference [8] should we use.

END OF ALL QUERIES