REGULAR PAPER

An Enhanced Security Protocol for VANET-based Entertainment Services

Jung-Yoon Kim[†], ***Student Member and*** **Hyoung-Kee Choi**[††], ***Member***

SUMMARY Multimedia transactions between vehicles are expected to become a promising application in VANETs but security is a fundamental issue that must be resolved before such transactions can become practical and trusted. Existing certificate-based digital signature schemes are ineffective for ensuring the security of multimedia transactions in VANETs. This ineffectiveness exists because there is no guarantee that (1) vehicles can download the latest certificate revocation lists or that (2) vehicles can complete a multimedia transaction before leaving their communication range. These two problems result, respectively, from a lack of infrastructure and from the inconsistent connectivity inherent in VANETs. In this paper, we propose a digital signature approach that combines a certificateless signature scheme and short-lived public keys to alleviate these problems. We then propose a security protocol that uses the proposed signature approach for multimedia transactions between vehicles. The proposed protocol enables vehicles to trade in multimedia resources without an online trust authority. We provide an analytical approach to optimizing the security of the proposed protocol. The security and performance of our protocol are evaluated via simulation and theoretical analysis. Based on these evaluations, we contend that the proposed protocol is practical for multimedia transactions in VANETs in terms of security and performance.

key words: *VANET security, multimedia transaction, security protocol, nonrepudiation, digital signature*

1. Introduction

Most popular Internet applications rely on the existence of a contemporaneous end-to-end link between the source and destination. However, for many networks such an "existence" is invalid. In these networks, devices can constitute a collaborative network to establish an opportunistic link between the source and destination. These networks characterized by long propagation delays and/or intermittent connectivity are often referred to as delay tolerant networks (DTNs). Recently, we have witnessed the placement of ad hoc networks in a primary position to represent an opportunistic collaborative network. Forms of this emergent communication paradigm are wide ranging and include low-cost Internet service provision in remote, social-based networks to allow humans to communicate without network infrastructure, pocket-switched networks, underwater networks, or other situations that impose gatekeepers. In particular, we are interested in vehicular ad hoc networks (VANETs). In VANETs, vehicles can communicate with nearby vehicles to gather traffic information (referred to as V2V communications), and also with fixed roadside units (RSUs) as a way to connect to the Internet (denoted as V2I communications). Both of these types of communications can be used (1) to enhance drivers' safety and efficiency with traffic information (safety applications) and (2) to make travel comfortable and enjoyable with infotainment (nonsafety applications). Because many works on safety applications in VANETs are already available [1], [2], [3], [4], we focus on nonsafety applications, especially on multimedia services in automotive environments.

In-vehicle multimedia services can be provided via existing mediums, such as satellite radio. These traditional mediums cannot provide on-demand multimedia services to users because these mediums are based on one-way broadcasts. However, the trend in multimedia services is away from one-way broadcasts to two-way on-demand services. This transition in multimedia services can be found in existing multimedia services, such as TV. Unlike conventional TV services, the latest TV services (e.g. IPTV and smart TV) provide a lot of on-demand multimedia resources via two-way communication networks, such as the Internet. Based on this new trend in multimedia services, we can expect that the trend for in-vehicle multimedia services will be from one-way broadcasts to on-demand services based on two-way communications.

As typical two-way communication systems, cellular networks and VANETs can be used for on-demand services in vehicular environments. In cellular networks, a vehicle can access on-demand multimedia services over the Internet via a base station. However, cellular networks are not a preferred option for in-vehicle multimedia services because they have a lower data rate and higher costs than WLAN-based VANETs (more specifically, 802.11p-based VANETs). Furthermore, this use of a base station to access multimedia services conflicts with the current trend to reduce the overloading of base stations. This trend can be found in the latest standardization activities for machine-to-machine

Manuscript received October 28, 2011.
Manuscript revised October 28, 2011.
†The author is with the Department of Mobile Systems Engineering, Sungkyunkwan University, Suwon, 440-746 Korea.
††The author is with the School of Information and Communication Engineering, Sungkyunkwan University, Suwon, 440-746 Korea. (Corresponding author)

(M2M) communications in 3rd Generation Partnership Project (3GPP) and 3GPP2; mobile nodes within their communication range can communicate directly with each other via ad hoc communications, instead of relying on base stations, to reduce the traffic load on base stations and to improve the efficiency. In VANETs, vehicles typically are equipped to access on-demand multimedia services over the Internet via V2I communications. However, vehicles cannot always exercise this capability because the costs of RSU installation and maintenance mean that there are too few of them along roads to allow a connection to every vehicle. When a vehicle is beyond the coverage range of RSUs, it can access an RSU through opportunistic contacts in collaborative networks or directly purchase multimedia resources from nearby vehicles via V2V communications.[1] As a result, VANETs are a better option than cellular networks for on-demand multimedia services.

Because VANETs are a special implementation of opportunistic collaborative networks, VANET-based multimedia services rely on a trustworthy, secure, and collaborative network infrastructure to provide correct data and information. Further, vehicles in the networks are expected to behave honestly and beneficially with other vehicles. In multimedia transactions between vehicles, however, some buyers and sellers may try to misrepresent themselves or act dishonestly. Our goal is to solve these security issues effectively and efficiently. It appears that we can achieve our goal by using existing security mechanisms based on digital certificates. A certificate can be used as a license for a vehicle to perform secure and trustworthy multimedia transactions. More specifically, a certificate is used to bind a security material (i.e., a public/private key pair) with its owner.

Unfortunately, it is not easy to use existing certificate-based security mechanisms in VANETs. This is because for every multimedia transaction a certificate should be validated by a centralized authority over the Internet; when a transaction is performed, a vehicle should be able to connect to a centralized authority and receive from this authority the latest information about the validity of a certificate. However, a vehicle beyond the coverage of RSUs cannot access a centralized authority, so existing certificate-based security mechanisms cannot be used for multimedia transactions in VANETs.

Our approach is to remove the need for vehicles to connect to an RSU for secure multimedia transactions by removing the need to check the validity of certificates. Because a certificate requires invalidation if the corresponding security material (i.e., a private key) has been compromised, we can remove the need for this validity check by periodically updating security materials

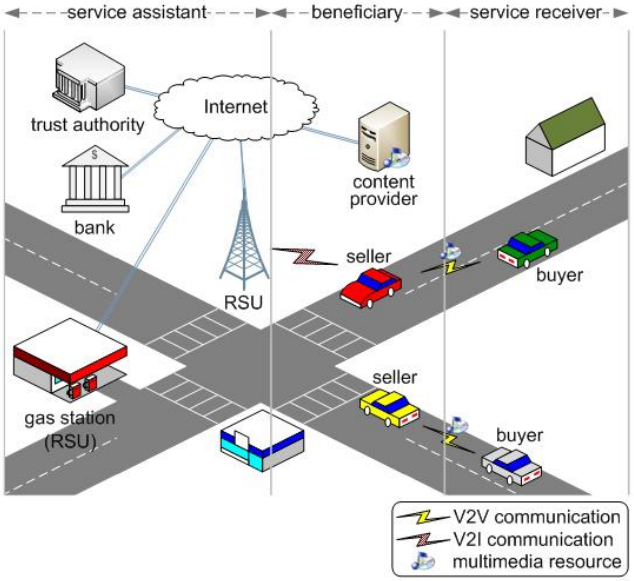

Fig. 1 System architecture for multimedia transactions in VANETs

at short intervals. The short lifetime of security materials leaves too little time to threaten them, so they appear fresh and trustworthy. We enhance the efficiency of our approach by using a security scheme that protects communications without recourse to a certificate.

The main contributions of this paper are threefold: (1) We alleviate the limitations of existing security mechanisms and propose a security protocol for multimedia transactions in VANETs. (2) We evaluate the security of the proposed protocol through a simulation study and compare overhead between the proposed protocol and other proposals through a theoretical analysis. (3) We provide an analytical approach to optimizing the security of the proposed protocol.

The rest of the paper is organized as follows. In Section 2, we define the system model for multimedia transactions in VANETs. In Section 3, we survey the features of many security proposals that can be used for multimedia transactions in VANETs. Section 4 describes the background of the proposed protocol. Section 5 provides a solution to alleviate the limitations of existing security mechanisms and then proposes a security protocol for multimedia transactions in VANETs; this is followed in Section 6 with an analytical approach to optimizing the security of our protocol. Section 7 covers the performance and security analysis of the proposed protocol and simulation results that were undertaken to verify our analytical approach and to evaluate the security of our protocol. Section 8 presents our conclusions.

[1] Using opportunistic links provides worse performance because of the long propagation delay, so we focus on a direct transaction between nearby vehicles for multimedia transactions.

2. System Model

In this section, we provide descriptions of the system architecture, the attack model, and the system requirements for multimedia transactions in VANETs.

2.1 System Architecture

Fig. 1 depicts the general system architecture for multimedia transactions in VANETs. As shown in Fig. 1, the participating entities can be divided into three groups based on their roles: a beneficiary, a service receiver, and a service assistant.

A beneficiary means the entities that receive money from the sale of multimedia resources. An entity holding the copyright on multimedia resources can sell them directly to consumers or transfer the sales rights to agents who get a predetermined commission from their sales. We refer to the copyright holders and recipients of sales rights, respectively, as content providers and sellers. Anyone can be registered as a seller by getting permission from a content provider to sell multimedia resources. A seller can use his or her own vehicle to sell the resources to other vehicles via V2V communications as shown in Fig. 1.

The service receivers category includes vehicles purchasing multimedia resources from beneficiaries. Because content providers sell multimedia resources through the Internet, a service receiver can purchase these resources via a nearby RSU within its communication range through V2I communications. Note that a vehicle cannot always connect to an RSU. This is because the number of RSUs may be restricted because of the cost of their installation and management. If a service receiver is unable to connect to the Internet, it can download multimedia resources from sellers near itself through V2V communications as shown in Fig. 1.

A service assistant means the group of organizations and facilities required for operating multimedia transaction systems in VANETs. A trust authority (TA), an RSU, and a bank are representative service assistants. A TA helps beneficiaries and service receivers to participate in multimedia transactions by authorizing them to digitally sign multimedia transactions. An RSU serves as the gateway to the Internet. A bank is responsible for the transfer of money for the multimedia resources between the bank accounts of the service receivers and beneficiaries. Service assistants can use Transport Layer Security to establish a secure channel between them.

2.2 Threat Model

Various threats are possible against multimedia transaction systems in VANETs. The differing goals of potential attacks permit their classification into three categories that also reflect the vulnerabilities of these systems. The three general forms of attack are fraudulent transactions, violations of drivers' privacy, and denial of service.

Fraudulent transactions: A malicious beneficiary may try to sell an invalid multimedia resource. A malicious nonbeneficiary may also commit this fraud by masquerading as an authorized beneficiary. Failure to pay is another way for an attacker to exploit the system. There are two possible methods for a malicious service receiver to obtain a multimedia resource without paying. The first is by intercepting the packets of multimedia resources as they are transmitted from a beneficiary to another service receiver. The second is by repudiating a contract to purchase a multimedia resource after having received it.

Violations of drivers' privacy: Violations of privacy include disclosing information about multimedia transactions and tracking movement of vehicles. By eavesdropping on transactions, an attacker can learn who buys a particular multimedia resource. An attacker can also trace a vehicle's movement by consistently eavesdropping on the victim's messages; linkage between messages makes it possible for an attacker to identify the source of the messages and ultimately trace the originator of a specific message by tracking messages sent from a specific source. Note that breaking this linkability is out of scope of this paper.

Denial of service: An attacker can interfere with other vehicles' multimedia transactions by jamming the communication channels [5]. Prevention of this type of denial-of-service attack is not considered in this paper because it cannot be fully thwarted in the application layer, and our focus is exclusively on application-layer approaches. Note, however, that some countermeasures proposed in [5] can be used to protect VANETs from this type of attack.

2.3 System Requirements

Defense against attacks on multimedia transaction systems in VANETs has several security requirements.

Nonrepudiation is used to preclude fraudulent transactions. A digital signature signed with a private key is used as evidence of nonrepudiation. A private key can be compromised by a malfunction or by an attack that extracts the private key file from the hard disk. After compromising a key, an attacker can use it to buy multimedia resources in the name of the owner of the key. Avoiding this form of system exploitation requires immediate revocation of compromised private keys, their corresponding public keys, and their certificates. Up-to-date and frequently distributed certificate revocation lists (CRLs) are necessary so that vehicles in the network can be protected from the use of compromised or revoked private keys. The distribution of recent CRLs is an

important system requirement for multimedia transactions.

Efficiency is another important property of transaction systems in VANETs, because of the need to overcome the inconsistent connectivity between vehicles that results from their speed and mobility [2]. Delays in completion of transactions must be minimal so that they can be completed while two vehicles are within each other's range. One way to gain the necessary efficiency is to reduce the computational overhead related to digital signatures.

Mutual authentication, data integrity, and protection against replay attack are needed, respectively, to prevent fraud by protecting against impersonation, message forgery, and message reuse. Confidentiality and access control are used to prevent illegal access to multimedia resources. Traceability of vehicles is needed for a TA to penalize wrongdoing, such as attempts to buy multimedia resources without sufficient funds. Privacy violations can be mitigated by preserving confidentiality and anonymity, both of which will protect disclosure of information about multimedia transactions. Some of these security requirements can be efficiently guaranteed by using symmetric encryption instead of asymmetric encryption. Sharing a symmetric key between two entities requires adoption of an authenticated key agreement algorithm.

3. Related Works

In transaction systems, a transaction involves the exchange of payment and delivery information between a buyer and a seller. An important element of such systems is preventing either the buyer or seller from repudiating a transaction. Historically, these systems have relied on TA-based [6], [7], [8] and certificate-based [9], [10], [11], [12] methods to ensure nonrepudiation. In the TA-based nonrepudiation protocols, a buyer and a seller allow a TA to store information about the multimedia transactions. The TA can arbitrate disputed transactions by using the stored information. This TA approach assumes that all the entities in the system trust the TA and can always connect with it. Certificate-based nonrepudiation relies on digital certificates. A buyer and a seller accept a transaction only if each other's signature on the transaction is verifiable through the certificate. This signature can be used as evidence to settle disputed transactions.

Despite the widespread use of these two nonrepudiation methods, neither is acceptable for multimedia transactions in VANETs. They are unsuitable because vehicles cannot always connect to a TA because of too little infrastructure in VANETs and certificate management imposes excessive overhead [1], [2]. Proposals to overcome these disadvantages of traditional means of nonrepudiation fall into three groups that are defined by their methodology. The three groups are certificateless signature-based, token-based, and delegation-based methodologies.

Certificateless signature methods reduce the excessive delays of certificate transmission, validation, and management in certificate-based signature approaches [13], [14], [15]. Reduction of this overhead is important because the inconsistent connectivity in VANETs does not guarantee buyers and sellers enough time to complete a transaction. In these approaches, the need for certificates is removed by using identities as public keys or making them self-certifying. Although most of these nonrepudiation approaches remove the need for certificates, they retain the burden of managing public-key revocation lists[2] to prevent the use of compromised private keys. Although our proposed signature solution belongs to this group, it is distinguished from the others in eliminating the need for certificates and management of revocation lists. This distinguishment makes the proposed solution suitable for VANETs.

Despite the reduction in the overhead of certificate management that is possible through the use of certificateless signature approaches, several researchers have gone a step further and concluded the generation and verification of a signature for every transaction is another unnecessary element of overhead. Their token-based signature schemes are a nonrepudiation technique that reduces the number of signatures generated and verified when several transactions occur between the same buyer and seller [16], [17], [18]. For instance, in the first transaction, a buyer hashes a random number n times, signs the nth hash result, and then sends the nth hash result and its signature to a seller. In the next transaction between them, this buyer generates the message authentication code (MAC) of the transaction using the $(n-1)$th hash result as a key. The buyer then sends the $(n-1)$th hash result and the MAC to the seller without its signature. This $(n-1)$th hash result, the MAC, and the signature on the nth hash result can be used as evidence for this transaction. Despite the gains in efficiency in repeated transactions with these token-based signature approaches, they do not detect revoked certificates efficiently. The buyer's certificate is verified only in the first transaction and not in subsequent transactions, which opens the way for security breaches in the later transactions.

In the two groups of signature just discussed, a buyer and a seller directly generate and send messages for transactions. A vehicle should save its computational and communication resources for higher priority applications, such as road safety applications [2], [3]. Delegation-based signature approaches reduce the data and computational loads imposed on vehicles by delegating signature authority for transactions to a proxy [3], [19],

[2] In these approaches, lists of revoked public keys are distributed instead of CRLs because of the absence of certificates.

[20]. The proxy instead of a vehicle performs computational intensive operations, such as validation of certificates/public keys and signature generation/verification. Hence, a vehicle can reserve its resources for higher priority applications. However, delegation-based signature schemes are impractical for VANETs. The reason once again is the inconsistent connectivity inherent in opportunistic networks, such as VANETs, which means that vehicles cannot always be sure of a connection to a proxy.

4. Background and Preliminary

Two important considerations in the design of a digital signature technique for multimedia transactions in VANETs are reduction of the overhead resulting from certificate management and efficient distribution of revocation lists.

Traditional public key cryptography (PKC) carries the burden of validating public keys and certificates. Identity-based cryptography (IBC) [21] has been proposed as a way to remove this disadvantage of PKC by using a user's identity as his or her public key. However, IBC is unacceptable because of an inherent key-escrow problem [22] in that a user's private key is known to a TA because the TA generates it. It is essential to protect private keys from disclosure simultaneously with removing the necessity of certificates for efficient real-time transactions.

Despite the importance of distributing up-to-date revocation lists, their dissemination in VANETs is not straightforward. The problem again lies with inconsistent connectivity, which means a vehicle cannot count on a TA connection to the Internet that will ensure dissemination of revocation lists. Without access to up-to-date lists, a vehicle risks acceptance of illegal signatures signed with revoked or compromised private keys.

In this section, we describe two security schemes that solve these two important problems. They are a certificateless signature scheme [13] and Kounga *et al.*'s scheme [11], [12].

4.1 Certificateless Signature Scheme

A certificateless signature scheme has no need for a certificate because anyone can recover a valid public key of a signer with only a few materials provided by a TA. Because a TA does not know the user's private key, this scheme does not suffer from the key-escrow problem.

A signer chooses a random number *PRI* as his or her private key and calculates his or her public key *PUB* as an irreversible function of *PRI* based on the discrete logarithm problem, that is $PUB = g^{PRI} \bmod n$. Note that g and $n = p \cdot q$ are a public base and a public modulus, respectively, where p and q are strong prime numbers secretly generated by a TA. The TA publishes $RECOVER = (PUB - ID)^{ID^{-1}} \bmod n$, where ID^{-1} is a multiplicative inverse of *ID* modulo $\phi(n) = (p-1)(q-1)$. *RECOVER* is used to recover a valid public key. A verifier recovers the signer's public key by using *ID* and *RECOVER* as follows: $PUB = RECOVER^{ID} + ID \bmod n$. Note that no one can generate *RECOVER* except the TA because of the integer factorization problem. A valid public key is recovered using *RECOVER* and *ID*, so a verifier does not need to validate the public key.

This digital signature scheme uses less overhead than PKC for certificate management because it removes the need for certificates. However, this approach cannot be used in VANETs because of the problem of distributing revocation lists.

Table 1 Notations used in the description of Kounga *et al.*'s scheme in Section 4.2

Notation	Description
$H_0(\cdot)$, $H_1(\cdot)$	Cryptographic one-way hash functions, such as SHA-256
SECRET	Secret derived from a strong passphrase chosen by a user
$TIME_0$	Issue time of a user's certificate
$INTERVAL_i$	ith time interval after $INTERVAL_0$, where $INTERVAL_0$ is the first time interval that starts at $TIME_0$
LEN	Length of each time interval
NUM	Total number of time intervals
$PUB_{NUM-i-1}$ / $PRI_{NUM-i-1}$	User's public/private key for a time interval $INTERVAL_i$
CHECK	Value used to validate $PUB_{NUM-i-1}$, where $0 \le i < NUM$
CERT	User's certificate including the user's identity *ID*, *LEN*, *NUM*, $TIME_0$, and *CHECK*

4.2 Kounga *et al.*'s Scheme

In Kounga *et al.*'s scheme [11], [12], a user does not have to revoke a public/private key pair before its expiration. This is because attackers do not have enough time to compromise a private key during its short lifetime. Furthermore, users can check whether a key pair has expired by using their local time. Therefore, revocation lists do not have to be managed.

Kounga *et al.*'s scheme is composed of three phases: registration, key generation, and key validation. Table 1 summarizes the notations.

Registration: When a user registers a device at a TA, the device obtains system parameters: two hash functions $H_0(\cdot)$ and $H_1(\cdot)$ and a large prime g. The device synchronizes its clock with the TA's clock and generates a secret *SECRET* by using the user's strong passphrase. Note that the device does not store *SECRET*. Then this device generates the check value *CHECK* as follows:

$$CHECK = H_0(g^{\prod_{j=0}^{NUM} H_1^{\,j}(SECRET)}), \quad (1)$$

where *NUM* is an integer chosen by the TA. *CHECK* is

later used to check the validity of the device's public key. $H_1^j(SECRET)$ means that $H_1(\cdot)$ is applied j times on *SECRET*. The device then sends the device's identity *ID* and the check value *CHECK* to the TA in order to obtain a certificate binding these two values. The TA sends the device the TA's public key and the certificate *CERT*, including *ID*, *LEN*, *NUM*, $TIME_0$, and *CHECK*, where *LEN* is an integer chosen by the TA and $TIME_0$ is the issue time of *CERT*. This certificate is verifiable with the TA's public key. After receiving *CERT*, the device divides time into intervals $INTERVAL_i$ of the equal length *LEN*, where $0 \le i < NUM$. The first time interval $INTERVAL_0$ starts at the issue time $TIME_0$ of *CERT*.

Key generation: In a time interval $INTERVAL_i$, where $0 \le i < NUM$, the device generates the private key $PRI_{NUM-i-1}$ and the public key $PUB_{NUM-i-1}$ for the time interval $INTERVAL_i$ as follows:

$$PRI_{NUM-i-1} = \prod_{j=0}^{NUM-i-1} H_1^j(SECRET), \quad PUB_{NUM-i-1} = g^{PRI_{NUM-i-1}} = g^{\prod_{j=0}^{NUM-i-1} H_1^j(SECRET)}. \tag{2}$$

Key validation: To validate a public key $PUB_{NUM-i-1}$, a verifier receives $PUB_{NUM-i-1}$, $H_1^{NUM-i}(SECRET)$, and *CERT* from the owner of the public key. After validating *CERT* with the TA's public key, the verifier determines *i* from its local time, $TIME_0$, and *LEN* as follows:

$$i = \left\lfloor \frac{\text{local time} - TIME_0}{LEN} \right\rfloor, \tag{3}$$

where $\lfloor x \rfloor$ is the floor function of x. $TIME_0$ and *LEN* are contained in *CERT*. The verifier then checks if

$$CHECK = H_0(PUB_{NUM-i-1}^{\prod_{j=NUM-i}^{NUM} H_1^j(SECRET)}) \tag{4}$$

holds, where *NUM* and *CHECK* are contained in *CERT*, to validate the public key. Each key pair expires automatically after *LEN* time units, when the next time interval $INTERVAL_{i+1}$ is reached. This is because the key pair is validated based on the current time. Obviously, a shorter *LEN* reduces the possibility of compromise of a private key.

The verifier can then use $PUB_{NUM-i-1}$ to verify signatures during $INTERVAL_{i+1}$. We have omitted explanations of how to generate and verify signatures.

Kounga *et al.*'s scheme has a problem with certificate management. The overhead and delays imposed may render this approach unsuitable for VANETs because of the need for speed so that vehicles can complete transactions before they move out of their communication range.

5. Proposed Protocol

Our design goal is to enable vehicles to securely trade in multimedia resources via V2V communications. Achieving this goal requires solving the problems of distributing revocation lists and validating certificates. In this section, we propose a signature approach that solves both problems by combining Kounga *et al.*'s scheme and a certificateless signature scheme. We then propose a security protocol for multimedia transactions in VANETs.

5.1 Combining Kounga *et al.*'s Scheme and Certificateless Signature Scheme

Applying the certificateless signature scheme described in Section 4.1 removes the need for certificates in Kounga *et al.*'s scheme. The proposed signature solution consists of registration, key generation, and key validation phases. It works as follows.

Registration: The registration phase has the same purpose as that of Kounga *et al.*'s scheme. The vehicle receives $H_0(\cdot)$, $H_1(\cdot)$, g, n, *LEN*, and *NUM* from a TA over a secure channel, such as an offline communication. g and n are the same as those in the certificateless signature scheme as described in Section 4.1. *LEN* and *NUM* have the same purpose as those used in Kounga *et al.*'s scheme and are chosen by the TA. After receiving them, the vehicle synchronizes its clock with the TA's clock and generates a secret *SECRET* in the same way as in Kounga *et al.*'s scheme. Then, it calculates

$$CHECK = H_0(g^{SECRET \cdot \sum_{j=1}^{NUM} H_1^j(SECRET)} \bmod n), \tag{5}$$

and transfers *ID*, *CHECK*, and $g^{SECRET} \bmod n$ to the TA, where *CHECK* is used to validate the vehicle's public keys. The TA chooses the issue time of *RECOVER* for this vehicle, which is called $TIME_0$, and computes

$$RECOVER = (g^{SECRET})^{H_0(ID\|CHECK\|TIME_0)^{-1}} \bmod n, \tag{6}$$

where *RECOVER* is used for recovering the valid *CHECK*. Note that $TIME_0$ for this vehicle should be equal to $TIME_i$ for other vehicles, where $0 \le i < NUM$, for security optimization given in Section 6. After receiving *RECOVER* and $TIME_0$ from the TA, the vehicle checks whether *RECOVER* is valid as follows:

$$RECOVER^{H_0(ID\|CHECK\|TIME_0)} \equiv g^{SECRET} \bmod n. \tag{7}$$

The TA and the vehicle then divide time into intervals $INTERVAL_i$ of the equal length *LEN*, where $0 \le i < NUM$ and $INTERVAL_0$ starts at $TIME_0$. The vehicle then stores *ID*, *CHECK*, $TIME_0$, *RECOVER*, g, n, $H_0(\cdot)$, $H_1(\cdot)$, *LEN*, and *NUM* for later use.

Key generation: The public/private key pair for a time interval $INTERVAL_i$ is calculated as follows:

$$PRI_{NUM-i-1} = SECRET \cdot \sum_{j=1}^{NUM-i-1} H_1^j(SECRET), \quad PUB_{NUM-i-1} = g^{PRI_{NUM-i-1}} \bmod n. \tag{8}$$

Key validation: The vehicle transfers *ID*, *CHECK*, $TIME_0$, *RECOVER*, $H_1^{NUM-i}(SECRET)$, and $PUB_{NUM-i-1}$ to a verifier in $INTERVAL_i$. After receiving them, the verifier determines *i* by using (3) and calculates

Table 2 Notations used in the description of the proposed protocol in Section 5.2

Notation	Description
$SIG_K(M)$	Message M and its signature signed with a private key K
$ENC_K(M)$	Message M encrypted with a symmetric key K
LIST	List of metadata for the seller's multimedia resources
EXR	Buyer's expected multimedia resource
DEK	Data encryption key for the confidentiality of multimedia resources
SK	Session key between a buyer and a seller used to securely transmit *DEK*
ENR	Encrypted multimedia resource

$$m = RECOVER^{\sum_{i=NUM-i}^{NUM} H_1^j(SECRET)}, \quad CHECK' = H_0\left(PUB_{NUM-i-1} \cdot \left(m^{H_0(ID\|CHECK\|TIME_0)}\right)\right). \tag{9}$$

If $CHECK' = CHECK$ holds, the verifier believes that the public key is valid during the time interval $INTERVAL_i$.

After the validation, a signer and a verifier can use the Schnorr signature method for signature generation and verification with $PRI_{NUM-i-1}$ and $PUB_{NUM-i-1} = g^{PRI_{NUM-i-1}} \bmod n$, respectively.

5.2 Security Protocol for Multimedia Transactions in VANETs

The proposed security protocol has three phases: sign-up, transaction, and billing. In the sign-up phase, a vehicle communicates with a TA over a secure channel to receive materials for the multimedia transaction system. After the registration, two vehicles within range of each other can trade in multimedia resources through the transaction phase. Later, when connected to an RSU, the seller bills the TA for payment by sending digital signatures on the transaction as a way to verify the sale. Table 2 describes the notations used in this section.

Sign-up: Before participating in multimedia transactions a user must sign up with the multimedia transaction system. A user submits to the TA offline his or her private information, bank account information, and a pseudo identity, *PID*, that the user has randomly chosen. *PID* is used as the vehicle's identity *ID* instead of its real identity to confer greater anonymity and improve security. The bank account is used later to transfer money in transactions. The TA completes the sign-up by storing and registering the information on the vehicle and driver.

Transaction: Two vehicles perform this phase to trade in multimedia resources. Fig. 2 depicts the procedure.

The seller periodically broadcasts a list of multimedia resources, say *LIST*. msg_1 in Fig. 2 includes *LIST*, TS_1, and the seller's signature, where TS_1 is a timestamp. The buyer then requests an expected multimedia resource *EXR* contained in *LIST* by sending msg_2 in Fig. 2 to the seller. msg_2 includes *EXR*, TS_2, and their signature, where TS_2 is the buyer's timestamp.

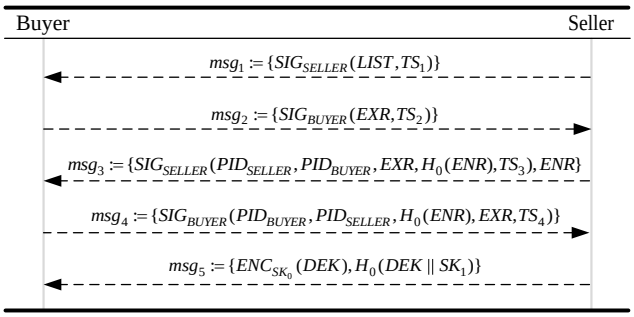

Fig. 2 System architecture for multimedia transactions in VANETs

After receiving msg_2, the seller sends msg_3 in Fig. 2 to the buyer. msg_3 contains (PID_{SELLER}, PID_{BUYER}, *EXR*, $H_0(ENR)$, TS_3), their signature, and the encrypted multimedia resource *ENR* corresponding to *EXR*. *ENR* is encrypted with a data encryption key (*DEK*). TS_3 is the seller's timestamp. Note that each multimedia resource is encrypted with a different *DEK*, and each *DEK* is randomly generated by a content provider. The seller receives *DEK* from a content provider in advance via V2I communications when connected to an RSU. The signature in msg_3 can later be used as evidence that the seller sent *ENR* corresponding to *EXR* to the buyer at TS_3.

After receiving msg_3, the buyer checks if the two *EXR*s in msg_2 and msg_3 are equal, and validates the received *ENR* by using $H_0(ENR)$ signed by the seller in msg_3. The buyer then confirms the purchase of *ENR* by sending msg_4 to the seller as shown in Fig. 2. msg_4 includes (PID_{BUYER}, PID_{SELLER}, $H_0(ENR)$, *EXR*, TS_4) and their signature, where TS_4 is the buyer's timestamp. This signature can later be used as evidence that the buyer bought *ENR* corresponding to *EXR* from the seller at TS_4.

After receiving msg_4, the seller checks whether the two *EXR*s in msg_2 and msg_4 are equal. After that, the buyer and the seller calculate a session key $SK = PUBKEY_{SELLER}^{PRIKEY_{BUYER}} = PUBKEY_{BUYER}^{PRIKEY_{SELLER}} = g^{PRIKEY_{SELLER} \cdot PRIKEY_{BUYER}}$ based on the Diffie-Hellman algorithm, where $PRIKEY_X$ and $PUBKEY_X$ denote X's private key $PRI_{NUM-i-1}$ and public key $PUB_{NUM-i-1} = g^{PRI_{NUM-i-1}} \bmod n$, respectively. The buyer and the seller then calculate $SK_0 = H_0(SK \| 0)$ and $SK_1 = H_0(SK \| 1)$. The seller encrypts *DEK* with SK_0, computes $H_0(DEK \| SK_1)$ as a MAC of *DEK*, and sends msg_5 to the buyer as shown in Fig. 2. msg_5 includes the encrypted *DEK* and the MAC.

The buyer uses *SK* to obtain *DEK* and then can enjoy the multimedia resource by decrypting *ENR* with *DEK*. If the buyer fails to receive msg_5, the buyer can later download msg_5 from the content provider via an RSU by sending him or her msg_3 as evidence of the transaction.

Billing: When connected to an RSU, the seller transmits msg_4 to the TA via V2I communications as evidence of the sale. The TA verifies the signature on (PID_{BUYER}, PID_{SELLER}, $H_0(ENR)$, *EXR*, TS_4) in msg_4 and

then uses PID_{BUYER} and PID_{SELLER} to search the driver's bank account from its database. The TA requests a bank to transfer a certain amount of money from the buyer's account to the content provider's one as a sales margin and to the seller's account as an incentive. Determining these amounts is beyond the scope of this paper.

6. Analytical Approach: Security Optimization

Because the proposed protocol does not manage lists of revoked public/private key pairs, it is difficult to protect against the use of compromised private keys before key pairs are updated. In this section, we will provide an analytical approach to minimizing the probability that a vehicle of interest communicates with *malicious vehicles*, say $P_{insecure}$, where *malicious vehicles* are defined as vehicles using compromised private keys. We refer to a vehicle of interest as *VEHICLE*. We can derive $P_{insecure}$ as follows.

Let us define the two counting processes $\{N_{comm}(t)\}_{t \geq 0}$ and $\{N_{comp}(t)\}_{t \geq 0}$. They count the number of vehicles communicating with *VEHICLE* by a time t and the number of vehicles using compromised private keys by a time t, respectively. Hence, the rate at which *VEHICLE* communicates with other vehicles and the rate at which vehicles use compromised private keys are respectively given by

$$R_{comm} = \lim_{t \to \infty} \frac{N_{comm}(t)}{t}, \quad R_{comp} = \lim_{t \to \infty} \frac{N_{comp}(t)}{t}. \tag{10}$$

We assume that $TIME_0$ for *VEHICLE* is zero, where $TIME_0$ is the starting time of the first time interval $INTERVAL_0$ described in Section 5.2 (that is, *VEHICLE* generates the first public/private key pair at the time t = 0). Just after t = 0, *VEHICLE* has confidence that no private key is compromised because all vehicles' key pairs have just been updated at t = 0. As time passes, however, the number of *malicious vehicles* and their chances of communicating with *VEHICLE* increase. Let us assume that $N_{vehicle}$ is the number of total vehicles in the system. The proportion of *malicious vehicles* among the total vehicles by a time t is given by

$$P_{comp}(t) = \frac{R_{comp} \cdot t}{N_{vehicle}}. \tag{11}$$

The number of *malicious vehicles* communicating with *VEHICLE* by a time t, say $N_{mal}(t)$, is

$$\begin{aligned} N_{mal}(t) = P_{comp}(t) \cdot N_{comm}(t) &= \frac{R_{comp} \cdot t}{N_{vehicle}} \cdot \int_{x=0}^{t} R_{comm} dx \\ &= \frac{R_{comm} \cdot R_{comp} \cdot t^2}{N_{vehicle}}. \end{aligned} \tag{12}$$

Because *VEHICLE* does not recognize that these *malicious vehicles* use compromised private keys, it will accept all signatures from *malicious vehicles*. We assume that the inter-update times (time between successive updates of public/private key pairs) come from a sequence of independent and identically distributed random variables $\{T_i\}_{i \geq 1}$. Let $E[T] := E[T_i]$ and $E[T^2] := E[T_i^2]$. During T_1, at which the vehicles' public/private key pairs are updated for the first time, *VEHICLE* will accept signatures from the $N_{mal}(T_1)$ *malicious vehicles*.

Table 3 Measured delays to perform operations and functions used in [12], [14], [15], and our protocol

Description	Delay (ms)
Delay to perform a cryptographic one-way hash function (SHA-256)	0.006
Delay to perform a symmetric en/decryption algorithm (AES-128)	0.004
Delay to calculate a 1024-bit modular exponentiation	1.007
Delay to calculate a 1024-bit modular multiplication	0.004
Delay to calculate a 1024-bit modular addition/subtraction	0.0005

$N_{mal}(t)$ is a renewal process that renews itself at time instants $\{T_i\}$ so the rate of communicating with *malicious vehicles*, say R_{mal}, is given by

$$R_{mal} = \lim_{t \to \infty} \frac{NUM_{mal}(t)}{t} = \frac{E[N_{mal}(T_1)]}{E[T_1]}, \tag{13}$$

where $NUM_{mal}(t)$ is the sum of $N_{mal}(T_i)$ and i = 1, 2, ⋯, $m(t)$. $m(t)$ is the number of updates of key pairs by a time t. Note that (13) follows from the elementary renewal reward theorem [23].

For given R_{comm} and R_{mal}, the proportion of *malicious vehicles* among vehicles communicating with *VEHICLE* is given as R_{mal} / R_{comm}. Hence by definition

$$\begin{aligned} P_{insecure} = \frac{R_{mal}}{R_{comm}} &= \frac{E\left[\frac{R_{comm} \cdot R_{comp} \cdot T^2}{N_{vehicle}}\right]}{R_{comm} \cdot E[T]} \\ &= R_{comp} \cdot \left(\frac{Var(T)}{N_{vehicle} \cdot E[T]} + \frac{E[T]}{N_{vehicle}} \right). \end{aligned} \tag{14}$$

According to (14), $P_{insecure}$ increases as a function of $Var(T)$ for the given $N_{vehicle}$, R_{comp}, and $E[T]$. T_i is equal to the update interval $INTERVAL_i$ of vehicles' public/private key pairs. All vehicles should simultaneously update their key pairs at the uniform time interval of *LEN* to minimize $Var(T)$. Hence, $TIME_0$ for a vehicle should be equal to one of $TIME_i$ for others, where $TIME_i$ is the starting time of $INTERVAL_i$ and $0 \leq i < NUM$, and each key pair should be updated at the same interval of *LEN*. In this case, although a vehicle should wait until $TIME_0$ to join the system for the very first time, $Var(T)$ is zero and $P_{insecure}$ is minimized for the given $E[T]$, R_{comp}, and $N_{vehicle}$. This makes the proposed protocol more secure than others against the use of compromised private keys.

7. Performance and Security Analysis

This section provides the simulation results and performance and security analyses of the proposed protocol. For the simulation and performance analysis, we measured the times taken to operate SHA-256, AES-128, modular exponentiation, multiplication, and addition on a Pentium IV 3.0GHz with 2GB random access memory using the MIRACL [24] as shown in

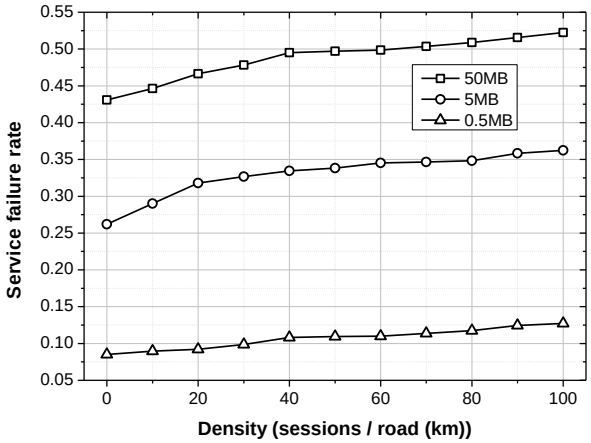

Fig. 3 The service failure rate increases as a function of session density and the size of a multimedia resource.

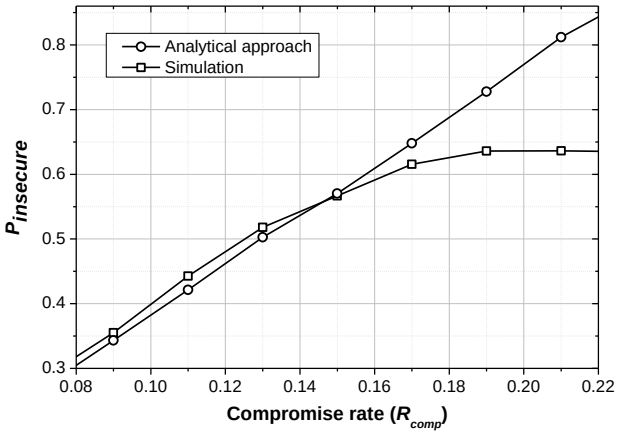

Fig. 4 Comparison of the two $P_{insecure}$ from the simulation result and the analytical approach in Section 6

Table 3.

7.1 Simulation Results

In this section, we use the ns-2 simulator to conduct a simulation to evaluate the performance and security of the proposed protocol and to verify the result of our analytical approach.

Simulation setup: We include in the overall delay the processing delays required to compute the operations denoted in Table 3. To improve the accuracy of the simulation results, we generate a realistic vehicular mobility model that is based on road topology extracted from a real street map in the TIGER database [25]. The model is created by using the Mobility Model Generator for Vehicular Networks (MOVE) [26]. The map used in the simulation is a realistic urban traffic environment that corresponds to Manhattan, NY. The map is scaled down to 6km x 6km in size. We set a vehicle's transmission range as 250m in accordance with Wu *et al.*'s measurement study [27]. We set an RSU's transmission range as 500m, because an RSU has a larger transmission range than vehicles. There are 100 vehicles moving on the simulation map. Each vehicle moves from a randomly chosen starting point to a randomly chosen destination. Two vehicles within communication range of each other can randomly start to perform multimedia transactions. When a destination is reached, the vehicle terminates all communication because we assume that the driver turns off the engine and gets out of the car. The simulation time is 500s; the output data collected during the first 50s is excluded from the simulation results because initialization bias occurs in the warm-up period of the simulation.

Simulation result: If two vehicles exceed their communication range before they complete a multimedia transaction, the transaction fails. We define a *failed transaction* as one that failed before completion. The *service failure rate* is defined as a ratio of the number of *failed transactions* to the number of transactions that occurred during the simulation. The number of ongoing transactions on the same channel may have an effect on the *service failure rate* because of congestion in the channel. We refer to the number of ongoing transactions (sessions) per kilometer as the *session density*. Fig. 3 depicts the *service failure rate* for each size of a multimedia resource as a function of the *session density*. When *session density* increases, the time required to successfully complete a transaction may lengthen because of increased channel congestion. Because many vehicles are moving toward different destinations, they have limited residence time within their communication range. Hence, if the time necessary to complete a transaction lengthens, the *service failure rate* increases. Consequently, the *service failure rate* increases as a function of the *session density* as shown in Fig. 3. By analogy, the size of a multimedia resource has an effect on the *service failure rate* as shown in Fig. 3; a larger multimedia resource causes a longer communication delay.

We compare $P_{insecure}$ measured via the simulation with another $P_{insecure}$ derived from the analytical approach in Section 6 as a function of compromise rate R_{comp}. Note that $P_{insecure}$ is the probability of accepting digital signatures from vehicles using compromised private keys and R_{comp} is the rate at which compromised private keys are used by vehicles as described in Section 6. To measure $P_{insecure}$ in the simulation, we assume that private keys are compromised according to an exponential distribution with R_{comp}. Fig. 4 illustrates the comparison of these two $P_{insecure}$. As shown in Fig. 4, when R_{comp} increases beyond a certain value ($\approx$ 0.17), $P_{insecure}$ measured via the simulation no longer increases, which

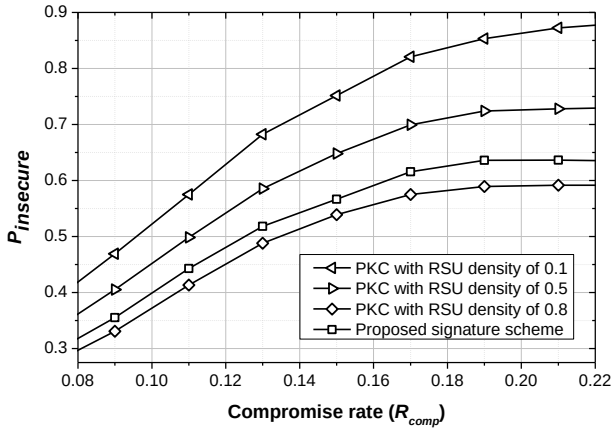

Fig. 5 Comparison of the four $P_{insecure}$ in our signature approach and PKC with the *RSU density* of 0.1, 0.5, and 0.8

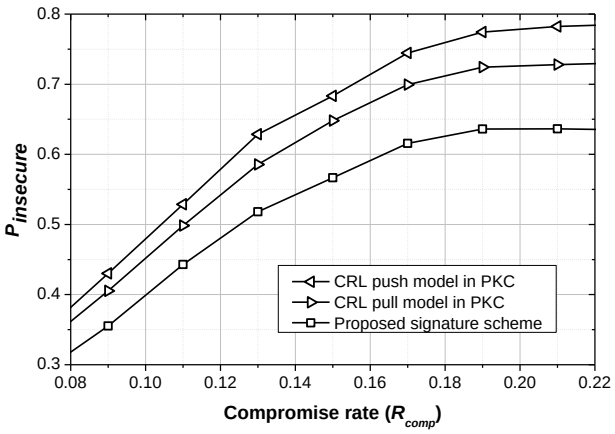

Fig. 6 Comparison of the two CRL update models in PKC and the key update model in our signature approach

is unlike $P_{insecure}$ derived from our analytical approach. This is because in the simulation mobility patterns can mean that some vehicles do not encounter each other. In addition, as mentioned earlier in this section, a vehicle does not communicate with vehicles that have already reached their destination. Hence, a vehicle may avoid communication in the simulation with some vehicles that are using compromised private keys. In the analytical approach, $P_{insecure}$ increases to one as an increase of R_{comp}, because all vehicles communicate with each other as time goes to infinity (in a steady-state).

We compare the proposed signature approach and PKC in terms of $P_{insecure}$ to evaluate the security of our proposal. The times between successive updates of public/private key pairs in the proposed signature approach, say T_i in Section 6, can be redefined as the times between successive updates of CRLs in PKC. This is possible because these two types of updates have the same purpose: to convince all vehicles that no private key is compromised or revoked. We refer to the times between successive CRL updates by a specific vehicle, say *VEHICLE*, in PKC as *inter-CRL update times*. The *inter-CRL update times* $\{T_i\}$ varies as a function of the residence time of *VEHICLE* within the coverage of RSUs, because *VEHICLE* cannot update the recent CRLs while out of the coverage of RSUs. We refer to the ratio of the coverage of RSUs to the total area of the simulation map as the *RSU density*. The *RSU density* has an important effect on $P_{insecure}$ in PKC, because $P_{insecure}$ varies according to the mean and variance of the *inter-CRL update times* $\{T_i\}$ as proved in Section 6.

Fig. 5 compares the four $P_{insecure}$ in the proposed signature approach and PKC with the *RSU density* of 0.1, 0.5, and 0.8, measured via the simulation. We call them $P_{insecure\text{-}ours}$, $P_{insecure\text{-}PKC\text{-}0.1}$, $P_{insecure\text{-}PKC\text{-}0.5}$, and $P_{insecure\text{-}PKC\text{-}0.8}$, respectively. We set the mean of the times between successive updates of public/private key pairs in the proposed signature approach, say $E[T]$ in Section 6, to be the same as the *inter-CRL update times* in PKC with an *RSU density* of 0.5. Hence, a fair comparison of $P_{insecure\text{-}ours}$ and $P_{insecure\text{-}PKC\text{-}0.5}$ is possible. As shown in Fig. 5, our approach is more secure than PKC against the use of compromised private keys under the same $E[T]$. This is because of the difference between the two $Var(T)$ of our approach and PKC as proved in Section 6. $P_{insecure\text{-}ours}$ is larger than $P_{insecure\text{-}PKC\text{-}0.8}$ as shown in Fig. 5. This is because $E[T]$ of PKC with an *RSU density* of 0.8 is smaller than that of our approach.

The above simulation is performed under the assumption that CRLs are disseminated based on the *CRL pull model* [28]. In this model, a vehicle regularly requests a TA to send the latest CRLs via RSUs. If a vehicle is within the coverage of RSUs, it can obtain the CRLs. Otherwise, the vehicle receives the recent CRLs once it returns to coverage by RSUs. There is another CRL update model: the *CRL push model* [28]. In this model, recent CRLs are regularly disseminated by a TA via RSUs by using multicast, which makes this model more efficient than the *CRL pull model* in terms of communication overhead. However, if a vehicle is beyond the coverage of RSUs when a TA distributes the CRLs, the vehicle must wait until the next CRL dissemination to update the recent CRLs. Hence, as shown in Fig. 6, the *CRL pull model* is more secure than the *CRL push model* against the use of compromised or revoked private keys. Fig. 6 shows that our key update model in the proposed signature approach outperforms the two CRL update models in PKC in terms of security against the use of compromised private keys. Furthermore, our key update model is more efficient than the two CRL update models in PKC in terms of communication because it does not need to download revocation lists.

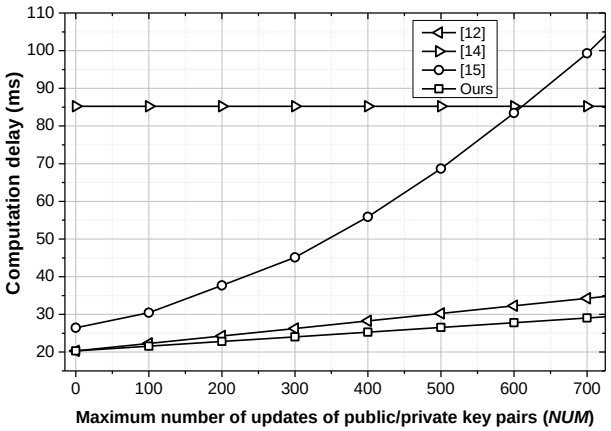

Fig. 7 Comparison between computation delays of the four protocols over the number of updates of key pairs, *NUM*

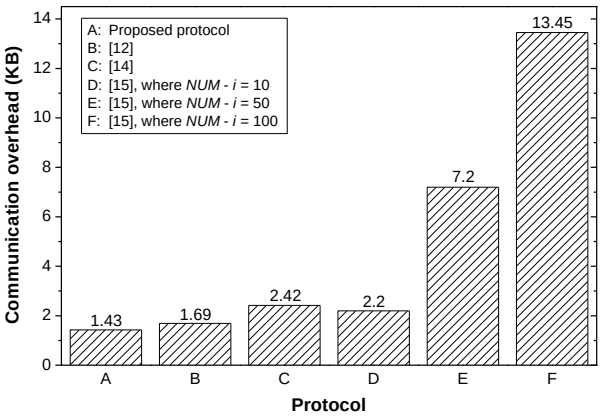

Fig. 8 Comparison between communication overhead of the four protocols

7.2 Performance Analysis

We compare the computation and communication overhead of our protocol with those of [12], [14], and [15] that have been proposed for transactions in VANETs.

Computation delay: In [15], because a private key is calculated based on repetitive nonmodular multiplications, the computation delay required to generate a private key increases as a function of the number of multiplications and the size of the operand. This operand size may also increase linearly with the number of multiplications. Hence, in a worst case scenario, the delay to generate a private key in [15] increases exponentially as a function of the number of multiplications. Our protocol is based on additions. Because the size of the operand of an addition increases at most one bit, the delay to generate a private key in our protocol increases almost linearly with the number of additions. As a result, the overall performance of [15] is more degraded than that of our protocol, according to the increase in the number of repetitive operations, say *NUM*.

Fig. 7 compares the overall computation delays of the four protocols based on the above analysis and the measured delays depicted in Table 3. For a more accurate comparison, we measured the delay required to perform repetitive multiplications in [15] for a given *NUM*. As shown in Fig. 7, the time complexity of [15] is O(NUM^2). Although [15] has several advantages, such as the absence of certificates and their revocation lists, this lack of scalability renders [15] impractical. The time complexity of the proposed protocol is O(*NUM*). Our protocol has better performance in terms of computation than the other three protocols when $NUM \geq 5$.

Communication overhead: To compare communication overhead, we compare the total length of messages used for a multimedia transaction between vehicles in [12], [14], [15], and our protocol. Note that we do not consider the overhead resulting from the transmissions of a multimedia resource and its information, say *ENR*, *EXR*, and *LIST*. This is because all four protocols have the same overhead for these transmissions. The length of values used in [12], [14], [15], and our protocol is summarized in Table 4. We can calculate the length of messages exchanged between vehicles in the proposed protocol by using Table 4. The calculated length (in bytes) of msg_1, msg_2, msg_3, msg_4, and msg_5 are 496, 496, 212, 212, and 48, respectively. The length of messages used in [12], [14], and [15] can also be calculated by using Table 4.

Fig. 8 compares the communication overhead of [12], [14], [15], and the proposed protocol. In Fig. 8, *A*, *B*, and *C* denote the proposed protocol, [12], and [14], respectively. *D*, *E*, and *F* refer to [15] where $NUM - i$ are 10, 50, and 100, respectively. In [15], the size of a signature increases linearly as a function of $NUM - i$, where the current time belongs to the ith time interval $INTERVAL_i$. Hence, the variation in *NUM* and *i* cannot be ignored in [15]. As shown in Fig. 8, our protocol outperforms the others in terms of communication.

7.3 Security Analysis

Table 5 compares [12], [14], [15], and the proposed protocol with respect to fulfillment of security requirements. As shown in Table 5, [12], [15], and our protocol guarantee **partial confidentiality**; this means that multimedia resources are encrypted, whereas transactions are not encrypted. Nevertheless, in [15] and our protocol, an attacker cannot learn the real identity of a specific vehicle that buys or sells a specific multimedia resource. This is because **identity anonymity** is preserved in these two protocols by using a pseudo

Table 4 Summary of the length of values used in [12], [14], [15], and the proposed protocol

Description of values	Notation in the proposed protocol	Length in bytes
Identities	PID_{BUYER}, PID_{SELLER}	8
Values generated based on modulus n	$RECOVER$, $PUB_{NUM-i-1}$	128
Timestamps	TS_1, TS_2, TS_3, TS_4, $TIME_0$, LEN	4
Random numbers	NUM	16
Hash results (SHA-256)	$H_0(DEK \| SK_1)$, $CHECK$, $H_1^{NUM-i}(SECRET)$	32
Results of symmetric encryption (AES-128)	$ENC_{SK_0}(DEK)$	16

Table 5 Comparison of the four protocols with respect to fulfillment of security requirements

Security requirements	[12]	[14]	[15]	Ours
Nonrepudiation	O	O	O	O
Mutual authentication	O	O	O	O
Confidentiality	Partial	O	Partial	Partial
Access control	O	O	O	O
Data integrity	O	O	Partial	O
Authenticated key agreement	X	N/A	X	O
Protection against replay attack	O	O	O	O
Traceability of misbehaving vehicles	O	O	O	O
Identity anonymity	X	O	O	O

identity, say *PID*, instead of the real one. **Data integrity** of some messages in [15] is not guaranteed because the MACs or signatures on these messages are not generated. Further, an **authenticated key agreement** for symmetric cryptography is not provided in [12] and [15]. The proposed protocol preserves data integrity by using digital signatures on msg_1, msg_2, msg_3, and msg_4 and a MAC on msg_5. The proposed protocol also provides an authenticated key agreement to prevent man-in-the-middle attacks by signing all the messages used in a key agreement. An authenticated key agreement is unavailable in [14], because a symmetric key does not need to be agreed upon in [14]. Although [14] guarantees all the security requirements for multimedia transactions, [14] is unsuitable for VANETs because it assumes a seller can connect to a TA for every transaction. This assumption is impractical for VANETs because of the lack of infrastructure in VANETs.

Other security requirements are fulfilled in the proposed protocol as follows.

Nonrepudiation: A vehicle calculates its private key by using a driver's strong passphrase and materials received from a TA. Without a valid passphrase and materials, no one can generate a valid private key, so the message signed with a private key can be used as evidence for liability.

Mutual authentication: A seller and a buyer are able to authenticate themselves to each other by sending signatures in msg_1 and msg_2, respectively.

Access control: A seller can control access to each multimedia resource by sending *DEK* only to an authorized buyer, because each multimedia resource is encrypted with a different *DEK*.

Protection against replay attack: Because all the transaction messages include a signed timestamp, an attacker cannot reuse these messages.

Traceability of misbehaving vehicles: A TA can trace the signer of a signature from its database by using *PID* and the signature contained in msg_3 and msg_4.

8. Conclusion

Traditional PKC is unsuitable for VANETs because of the need for certificate validation and CRL distribution. We have proposed a solution that combines two existing security schemes: a certificateless signature scheme and Kounga *et al.*'s scheme. This solution is suitable for VANETs because it eliminates the need for certificates and their revocation lists and enhances the efficiency of multimedia transactions. We also have proposed a security protocol for multimedia transactions in VANETs that uses the same solution. The proposed protocol enables vehicles to trade in multimedia resources without the help of an online TA. To strengthen the security of the proposed protocol, we have used an analytical approach to optimize the update interval of public/private key pairs because of the effect this interval has on precluding the possibility of use of compromised private keys. We also have used the ns-2 simulator to perform a simulation study to verify the result of the analytical approach and to evaluate the performance and security of the proposed protocol and discussed the results of the simulation. Based on the analysis and simulation results, we contend that the proposed protocol outperforms others in terms of computation and communication while guaranteeing the security requirements for multimedia transactions in VANETs.

References

[1] J.J., Haas, Y.-C. Hu, and K.P. Laberteaux, "Efficient Certificate Revocation List Organization and Distribution," *IEEE J. Selected Areas in Comm.*, vol. 29, no. 3, pp. 595-604, Mar. 2011.

[2] A. Wasef, R. Lu, X. Lin, and X. Shen, "Complementing Public Key Infrastructure to Secure Vehicular Ad Hoc Networks," *IEEE Wireless Comm.*, vol. 17, no. 5, pp. 22-28, Oct. 2010.

[3] S. Biswas and J. Mišić, "Deploying Proxy Signature in VANETs," *Proc. IEEE GLOBECOM '10*, pp. 1-6, Dec. 2010.

[4] Y. Hao, Y. Cheng, C. Zhou, and W. Song, "A Distributed Key Management Framework with Cooperative Message Authentication in VANETs," *IEEE J. Selected Areas in Comm.*, vol. 29, no. 3, pp. 616-629, Mar. 2011.

[5] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming Sensor Networks: Attack and Defense Strategies," *IEEE Network*, vol. 20, no. 3, pp. 41-47, May/June 2006.
[6] K.-W. Park, S.S. Lim, and K.H. Park, "Computationally Efficient PKI-Based Single Sign-On Protocol, PKASSO for Mobile Devices," *IEEE Trans. Computers*, vol. 57, no. 6, pp. 821-834, June 2008.
[7] P. Lin, H.-Y. Chen, Y. Fang, J.-Y. Jeng, and F.-S. Lu, "A Secure Mobile Electronic Payment Architecture Platform for Wireless Mobile Networks," *IEEE Trans. Wireless Comm.*, vol. 7, no. 7, pp. 2705-2713, July 2008.
[8] Y. Atif, "Building Trust in E-commerce," *IEEE Internet Computing*, vol. 6, no. 1, pp. 18-24, Jan./Feb. 2002.
[9] G. Wang, "An Abuse-Free Fair Contract-Signing Protocol Based on the RSA Signature," *IEEE Trans. Information Forensics and Security*, vol. 5, no. 1, pp. 158-168, Mar. 2010.
[10] M. Bellare, J.A. Garay, R. Hauser, A. Herzberg, H. Krawczyk, M. Steiner, G. Tsudik, E.V. Herreweghen, and M. Waidner, "Design, Implementation, and Deployment of the iKP Secure Electronic Payment System," *IEEE J. Selected Areas in Comm.*, vol. 18, no. 4, pp. 611-627, Apr. 2000.
[11] G. Kounga, T. Walter, and C. Schaefer, "Generating CA-authenticated Public Keys in Ad Hoc Networks," *Proc. ACM MobiHoc '08*, pp. 1-2, May 2008.
[12] G. Kounga and C. Schaefer, "Selling Multimedia Resources in Ad Hoc Networks," *IEEE Comm. Magazine*, vol. 46, no. 2, pp. 126-131, Feb. 2008.
[13] Y. Chang, C. Chang, and H. Huang, "Digital Signature with Message Recovery Using Self-certified Public Keys without Trustworthy System Authority," *Applied Math. and Computation*, vol. 161, no. 1, pp. 211-227, Feb. 2005.
[14] J.T. Isaac, J.S. Camara, S. Zeadally, and J.T. Marquez, "A Secure Vehicle-to-roadside Communication Payment Protocol in Vehicular Ad Hoc Networks," *Computer Comm.*, vol. 31, no. 10, pp. 2478-2484, June 2008.
[15] K.-E. Shin, H.-K. Choi, and J. Jeong, "A Practical Security Framework for a VANET-based Entertainment Service," *Proc. ACM PM2HW2N '09*, pp. 175-182, Oct. 2009.
[16] T.-F. Lee, T. Hwang, S.-H. Chang, and S.-K. Chong, "Enhanced Delegation-based Authentication Protocol for PCSs," *IEEE Trans. Wireless Comm.*, vol. 8, no. 5, pp. 2166-2171, May 2009.
[17] R. Gennaro and P. Rohatgi, "How to Sign Digital Streams," *Information and Computation*, vol. 165, no. 1, Feb. 2001, pp. 100-116.
[18] K. Ren, W. Lou, K. Kim, and R. Deng, "A Novel Privacy Preserving Authentication and Access Control Scheme for Pervasive Computing Environments," *IEEE Trans. Vehicular Technology*, vol. 55, no. 4, pp. 1373-1384, July 2006.
[19] Y. Yu, C. Xu, X. Huang, and Y. Mu, "An Efficient Anonymous Proxy Signature Scheme with Provable Security," *Computer Standards & Interfaces*, vol. 31, no. 2, pp. 348-353, Feb. 2009.
[20] Z. Shao, "Provably Secure Proxy-protected Signature Schemes Based on RSA," *Computers & Electrical Eng.*, vol. 35, no. 3, pp. 497-505, May 2009.
[21] X. Yi, "An Identity-based Signature Scheme from the Weil Pairing," *IEEE Comm. Letters*, vol. 7, no. 2, pp. 76-78, Feb. 2003.
[22] Z. Shao, "Self-certified Signature Scheme from Pairings," *J. Systems and Software*, vol. 80, no. 3, pp. 388-395, Mar. 2007.
[23] D.R. Cox, Renewal Theory, Methuen & Co., 1970.
[24] Shamus Software, "MIRACL Library 5.5.1," Dec. 2010; http://www.shamus.ie/index.php?page=home.
[25] U.S. Census Bureau, "Topologically Integrated Geographic Encoding and Referencing system," June 2009; http://www.census.gov/geo/www/tiger.
[26] F.K. Karnadi, Z.H. Mo, and K.-C. Lan, "Rapid Generation of Realistic Mobility Models for VANET," *Proc. IEEE WCNC '07*, pp. 2506-2511, Mar. 2007.
[27] H. Wu, M. Palekar, R. Fujimoto, R. Guensler, M. Hunter, J. Lee, and J. Ko, "An Empirical Study of Short Range Communications for Vehicles," *Proc. ACM VANET '05*, pp. 83-84, Sep. 2005.
[28] A. Fongen, "Scalability Analysis of Selected Certificate Validation Scenarios," *Proc. IEEE MILCOM '10*, pp. 2192-2198, Nov. 2010.

Jung-Yoon Kim is a Ph.D. student in Mobile Systems Engineering at Sungkyunkwan University in South Korea. He received his B.S. degree (2006) in Computer Engineering and M.S. degree (2008) in Electrical and Computer Engineering from Sungkyunkwan University in South Korea. He held internship at AhnLab (2004) where he worked for quality assurance of network security systems. He has research interests in network security, especially authentication and key management.

Hyoung-Kee Choi received a Ph.D. degree in electrical and computer engineering from Georgia Institute of Technology in 2001. He is an associate professor and a director of the Education Center for Mobile Communications in Sungkyunkwan University, Korea. He joined Lancope in 2001 and remained until 2004, where he guided and contributed to research in Internet security. His research interests span network security and Internet traffic modeling. He serves as an Associate Editor for ACM Transactions on Internet Technology.