This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2011 proceedings

Certificateless Secure Upload for Drive-thru Internet

Jun Song[†,‡], Yanyan Zhuang[†], Jianping Pan[†], and Lin Cai[†]
[†]University of Victoria, Victoria, BC, Canada
[‡]China University of Geosciences, Wuhan, China

***Abstract*—Vehicular ad hoc networks have attracted a lot of attention in recent years, in either vehicle-to-vehicle or vehicle-to-infrastructure scenarios. In this paper, we focus on the latter, particularly for vehicles to upload to roadside units, the so-called drive-thru Internet, in a secure and efficient manner. Due to the ad hoc nature and wireless communications, traditional certificate-based security schemes are either infeasible or inefficient in this scenario. Thus we propose a certificateless approach to secure upload in a drive-thru Internet. We discuss the attack model and the desired security properties, and how to achieve these properties through the proposed certificateless scheme. We implement and evaluate the proposed scheme, and also investigate how to mitigate the security overhead through the separation of security association and data transfer in a drive-thru Internet.**

***Index Terms*—Drive-thru Internet, security, certificateless**

I. Introduction

Vehicular Ad-hoc Networks (VANETs) have attracted a lot of attention from both industry and academia in recent years. Dedicated Short Range Communications (DSRC) [1] enable both vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) applications. In this paper, we focus on the V2I scenario. When vehicles drive through the communication range of a roadside unit (RSU), they can access the Internet for various location-based multimedia and intelligent transportation services. It is well known that vehicular networks are different from other types of mobile ad hoc networks. For instance, many VANET applications are location dependent, V2V and V2I wireless communications are broadcast in nature, network connectivity is intermittent, and vehicles often have high but predictable mobility. Considering these properties, simply implementing the existing IEEE 802.11i security standard and online certificate authority (CA)-based authentication schemes for VANET may not be desirable or feasible.

To facilitate vehicular communications, many security and privacy issues need to be addressed, such as identity legitimacy, privacy preserving, data integrity and the non-repudiation of information exchange and storage. Managing and distributing public key certificates is a well-recognized solution to achieving security and privacy. Although there is a remarkable research for public key infrastructure (PKI)-based VANET security solutions in the literature [2], [3], the previous work did not specifically optimize the network performance considering the VANET properties.

VANETs typically operate without full infrastructure support or legacy client-to-server bindings. There are insufficient or low-density RSUs deployed along the roads to cover part of the transportation system, which could be utilized and accessed by nearby vehicles at will. VANET applications need security assurance to authenticate entities and safe-guard information exchange through an insecure network. Similar to the previous work such as [4] for infrastructure-based wireless networks, key agreement protocol is a fundamental but more challenging building block in securing VANETs. It allows entities to implement key agreement and share the session key known to them only, which will be used to ensure data confidentiality and integrity. It also yields trustful authentication in broadcast and high contention environments such as VANETs.

In this paper, we focus on certificateless key agreement for drive-thru Internet services. We assume that the message from RSUs or vehicles can be relayed to locations outside their direct communication range. The RSU piggybacks security session parameters in its periodical beacon messages, and vehicles that are interested in accessing the RSU or willing to help can relay the message to locations further away from the RSU. During the relay process, the relayer adds its identity to the relayed message, so the receiver has the knowledge about the path of the relayed message. To access the services of RSU, a vehicle sends a reply message, which is relayed back to RSU. The relayer may aggregate the reply messages from others and its own whenever possible to reduce overhead.

The fundamental issues to be addressed in this work are 1) whether there exists a reliable, secure key agreement scheme to setup the secure authentication chain, 2) when and where to send authentication messages to the RSU, and 3) if the request can be accepted by the RSU, what is the expected amount of data can be uploaded to the RSU per drive-thru.

The main contributions of this paper are threefold. First, we propose a strong certificateless key agreement protocol following a practical approach and fully addressing the aforementioned security issues under common VANET attacks. Second, we implement and evaluate the proposed protocol, in terms of its computation cost on a commodity platform, showing the feasibility of the scheme. Third, we investigate how to mitigate the security overhead, due to the necessary security requirement of drive-thru Internet, through the separation of security association and data transfer in single and possibly multi-hop scenarios. To the best of our knowledge, this is the first paper to address both security property and network performance for VANET with a certificateless approach.

The rest of this paper is organized as follows. Section II overviews the related work. Section III describes the system model and main security threats for VANETs. Section IV presents our security protocol. Security analysis and performance results are given in Section V, followed by further discussions and conclusions in Section VI.

II. Related Work

Vehicular applications, such as multimedia services, data sharing and wireless communications, need security assur-

978-1-61284-231-8/11/$26.00 ©2011 IEEE

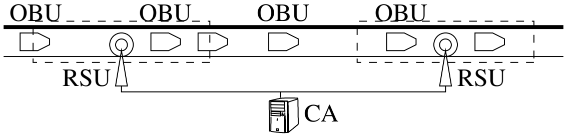

Fig. 1. System model

ance to authenticate entities and secure information exchange through an insecure network. Key agreement protocol is a fundamental building block in securing VANETs, which has to be tailored to special vehicular environments.

In the survey of certificateless public key encryption (CL-PKE) schemes [5] and the survey of identity-based key agreement (IB-PKE) schemes [6], twenty certificateless encryption schemes and twenty identity-based key agreement schemes are investigated, respectively. These schemes adopt different security models and thus have different efficiency. In Lippold's security model [7], there are three main secrets per entity, including the partial private key from the key generator, the private key from the user, and the ephemeral key for each session. It is shown that when there is at least one secret not compromised at each entity, the key agreement sessions are still fresh. In recent work, authentication and the subsequent download process between moving vehicles and RSUs have been investigated by Zhang [8] and Hao [9]. However, these implemented schemes still rely on CA-based online authentication, which has their limitations in VANET scenarios. Inspired by the previous work, we design and implement a novel and strong multi-secret, certificateless key agreement protocol in this paper, particularly for drive-thru Internet.

III. System Model and Security Threats

A. System Model

In this section, the system model and problem formulation are presented. As shown in Fig. 1, the system under consideration consists of three network entities: the root certificate authority (CA), stationary RSUs along the roadside, and moving vehicles equipped with on-board units (OBUs). According to the public-key infrastructure (PKI), the top-level authority can be integrated into CA, which is in charge of the registration of RSUs and vehicles, and issuing and verifying the certificates when needed, and is assumed to be fully trusted by all entities. RSUs are important part of a VANET infrastructure. Every RSU is connected by wired links to other RSUs and the CA; meanwhile, it has a wireless access point (AP) for all OBUs in its communication range. Despite of their security issues, RSUs are often trusted in VANET. They are deployed in an optimized way for high utilization due to their high cost. Vehicles equipped with OBUs can communicate with each other for local traffic condition, driving experience and related services in VANET. All OBUs should resist against reverse engineering attacks. Some short-term or ephemeral or one-time pseudonymous certificates are installed in OBUs to protect the privacy of vehicles.

B. Attack Model

In this paper, we assume that adversaries have the capability to launch any common active or passive attacks. Thus, for the V2V or V2I applications, neither trusted entities nor secure communications could be taken for granted. In the following, we list some additional threats due to the features of VANET.

First, it must be noted that, in previous security authentication and privacy-preserving schemes for vehicular networks, it is difficult to detect and resist against the *man-in-the-middle attack*, especially the *impersonation attack*. That is, since wireless channels are broadcast in nature, all communications can be overheard and all authentication should be anonymous. Hence, there are higher security requirements to achieve credible authentication in VANET.

Second, for VANETs, it is naturally relying on *certificateless* and *CA-oblivious* schemes due to the loosely-coupled network structures and intermittent coverage of the infrastructure. However, most of the previous work on VANETs mainly focuses on traditional PKI, i.e., using public-key certificates and CA-centered authentication directly or indirectly. This security requirement from the traditional PKI schemes is difficult to be satisfied, especially in open and wireless vehicular networks.

Third, VANET is subject to *time* or *location-based replay attack* due to its intermittent connectivity and broadcast nature. It can be carried out either by the originator or by an adversary who intercepts the requests during the relay or forward process and retransmits them with masqueraded identity. Thus, if the timestamp or location of communication entities are not embedded into the session parameters , an adversary could be able to impersonate other users through the reuse of outdated session information, or the replay in different regions via other high-speed communication links.

Besides the above security requirements, there are other threats that could result in considerable damage to network availability. For example, in reality, there may exist some malicious or selfish users that discard the relay message or refuse to carry-and-forward any messages for others. In this paper, we do not consider these issues specifically.

IV. Security Protocol Design

A. Preliminaries

The notations used in the rest of the paper are listed in Table I with their meaning explained. Due to the page limit, we only review part of the notations and theorems that are closely related to our proposed protocol.

Definition. $\mathbb{Z}_q \overset{def}{=} \{0, 1, \ldots, q-1\}$, and $\mathbb{Z}_q^*$ is relatively prime (co-prime) to q, where q is a prime integer. For the consistency of modular exponentiations, in the rest of paper, random variables are independently and uniformly chosen from $\mathbb{Z}_q^*$ or $\mathbb{Z}_q$, respectively.

Twin Diffie-Hellman (TDH) Trapdoor Theorems [12]: Using the above notations, suppose $X_1 \in \mathbb{G}$, $r, s \in \mathbb{Z}_q^*$, and $X_2 := g^s/X_1^r$. $\hat{Y}, \hat{Z}_1, \hat{Z}_2$ are random variables in $\mathbb{G}$ and defined as functions of X_1 and X_2. Then, 1) X_2 is uniformly distributed over $\mathbb{G}$; 2) X_1 and X_2 are independent;

TABLE I
NOTATIONS AND DEFINITIONS

Notations	Definitions
$\mathbb{Z}_q, \mathbb{Z}_q^*, \mathbb{Z}_q\backslash\{0\}$	$\mathbb{Z}_q$ is the set of integers, and $\mathbb{Z}_q^* = \mathbb{Z}_q\backslash\{0\}$ is the set of nonzero integers
$\mathbb{G}, q, g$	let $\mathbb{G}$ be a cyclic group of prime order q, generated by $g \in \mathbb{G}$
$\mathcal{A}, \mathcal{B}, \vec{\mathcal{S}}, \vec{\mathcal{R}}$	let $\mathcal{A}, \mathcal{B}$ be the two parties in communication, and $\vec{\mathcal{S}}, \vec{\mathcal{R}}$ denote the *Sender* and *Receiver* in one session
$s, S; a, A; b, B$	$s, a, b \in_R \mathbb{Z}_q^*$, s, S are the private/public key from KGC, and a, A are long-term private/public keys for user $\mathcal{A}$. Similarly b, B for user $\mathcal{B}$
$x, X; y, Y$	x, X is defined as the ephemeral private/public key for user $\mathcal{A}$, and y, Y for user $\mathcal{B}$, and $x, y \in_R \mathbb{Z}_q^*$
$\{D_{A_1}, D_{A_2}\}$	$\{D_{A_1}, D_{A_2}\} \in \mathbb{G}^2$ are the independent pseudonymous ID-based private key of $\mathcal{A}$. Similarly for $\mathcal{B}$.
T_A^*, L_A^*	denote the timestamp-based function and location-based function, values of $\mathcal{A}$. Similarly for $\mathcal{B}$.
$\mathcal{I}_A$, $\mathcal{M}_A$	denote the session identifier and session tuples of $\mathcal{A}$. Similarly for $\mathcal{B}$.
$\mathit{\Delta T}$	denote the time difference in VANET which is assigned by KGC
γ	the security parameters for hash functions

3) if $X_1 = g^{x_1}$ and $X_2 = g^{x_2}$, the probability that the value of $\hat{Z}_1^r \hat{Z}_2 = \hat{Y}^s$ does not agree with the value of $\hat{Z}_1 = \hat{Y}^{x_1} \wedge \hat{Z}_2 = \hat{Y}^{x_2}$ is at most $1/q$ (if the latter holds, the former certainly holds).

Dual (exponential) Challenge-Response (DCR) signature [13]: Let public keys $A = g^a$ and $X = g^x$, $B = g^b$ and $Y = g^y$. The DCR signature (DS) of $\mathcal{A}$ and $\mathcal{B}$ on message m_1, m_2 is a tuple of values X, Y, and $DS_{\mathcal{A},\mathcal{B}}$, respectively. Here, the same signature can be exchanged to compute (and verify) as follows: $DS_{\mathcal{A},\mathcal{B}}(m_1, m_2, X, Y) \overset{def}{=} g^{(x+da)(y+eb)} = (YB^e)^{x+da} = (XA^d)^{y+eb}$, where d and e denote $H(X, m_1)$ and $H(Y, m_2)$.

B. Protocol Design

In the following, we present the design of our certificateless authenticated key agreement protocol. It consists of both the optimized PK-AKE and ID-AKE schemes based on discrete logarithm, inspired by Lippold [7] and Hou [10]. This scheme is tailored to applications in a dynamic and insecure vehicular communication environment. There are five main secrets per entity in our proposed scheme, including the partial private key from KGC, the private key from the user, the ephemeral key, the time-based and location-based tags for each session. For each party, the proposed key agreement protocol stays secure as long as one of secrets has not been compromised.

System Initialization: The KGC selects four cryptographic hash functions, $H : \{0,1\}^* \rightarrow \{0,1\}^{\gamma} \times \{0,1\}^{\gamma}$, $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$, $H_2 : \mathbb{G} \rightarrow \mathbb{G}$, and $H_{MAC} : \{0,1\}^* \rightarrow \{0,1\}^{\gamma}$. In addition, H_{MAC} would be used as a MAC algorithm with secret key k', such as the SHA or MD5 algorithm.

Key extraction: Users $\mathcal{A}$ and $\mathcal{B}$ obtain their independent pseudonymous public keys, $d_{A_1} = H_1(r_A, ID_A)$, $d_{A_2} = H_2(H_1(r_A, ID_A))$, $d_{B_1} = H_1(r_B, ID_B)$, and $d_{B_2} = H_2(H_1(r_B, ID_B))$, and their corresponding pseudonymous private keys, $D_{A_1} = g^{sH_1(r_A, ID_A)}$, $D_{A_2} = g^{sH_2(H_1(r_A, ID_A))}$, $D_{B_1} = g^{sH_1(r_B, ID_B)}$, and $D_{B_2} = g^{sH_2(r_B, H_1(ID_B))}$. Here, r_A and r_B are secret random nonce values.

Key agreement: there are four sessions in the proposed certificateless key agreement protocol.

1. User $\mathcal{A}$ (sender) does the following steps. 1) Select an ephemeral private key $\hat{x} \in_R \{0,1\}^{\gamma}$, and compute $x = H_1(a, \hat{x})$, $X = g^x$. 2) Compute $M = S^a$, $N = S^x$, and $\alpha = {D_{A_1}}^x {D_{A_2}}^a$. 3) Destroy x, D_{A_1}, and D_{A_2}. 4) Initialize the session identifier to $\mathcal{I}_A$=$(\mathcal{A}, \mathcal{B}, \vec{\mathcal{S}}, X, M, N, \alpha, T_A^*, L_A^*)$. 5) Send $\mathcal{M}_A$=$(\mathcal{B}, \mathcal{A}, X, M, N, \alpha, T_A^*, L_A^*)$ to $\mathcal{B}$.

2. Upon receiving $\mathcal{M}_A$, user $\mathcal{B}$ (receiver) does the following steps. 1) Verify $X, M, N,\ \alpha \in \mathbb{G}^*$, and check whether $(T_B^* - T_A^*) \geqslant \mathit{\Delta T}$ or $L_A^* \not\equiv L_B^*$. If so, indicating the session has expired or the region is invalid, $\mathcal{B}$ cancels this session; otherwise, proceeds to the next steps. 2) Select an ephemeral private key $\hat{y} \in_R \{0,1\}^{\gamma}$, and compute $y = H_1(a, \hat{y})$ and $Y = g^y$. 3) Verify that $\alpha = N^{d_{A_1}} M^{d_{A_2}}$, and compute $U = S^b, V = S^y, \beta = {D_{B_1}}^y {D_{B_2}}^b$. 4) Compute $\lambda_1 = A^y, \lambda_2 = X^b, \lambda_3 = X^y, \lambda_4 = \alpha^{yd_{B_1}+bd_{B_2}}, \lambda_5 = {D_{B_1}}^{d_{A_1}}$, and $\lambda_6 = {D_{B_2}}^{d_{A_2}}$. 5) Compute (k, k') = $H(\mathcal{A}, \mathcal{B}, X, Y, M, N, U, V, \alpha, \beta, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6)$ and B_{MAC} = $H_{MAC}(k', \vec{\mathcal{R}}, \mathcal{B}, \mathcal{A}, Y, X, N, M, V, U, \beta, \alpha, L_B^*)$. 6) Destroy $\hat{y}, y, M, N, \alpha, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$ and λ_6, and then initialize the session identifier $\mathcal{I}_B$ = $(\mathcal{B}, \mathcal{A}, \vec{\mathcal{R}}, X, Y, M, N, U, V, \alpha, \beta, B_{MAC}, T_B^*, L_B^*)$. 7) Send $\mathcal{M}_B$ = $(\mathcal{A}, \mathcal{B}, X, Y, M, N, U, V, \alpha, \beta, B_{MAC}, T_B^*, L_B^*)$ to $\mathcal{A}$.

3. Upon receiving $\mathcal{M}_B$, user $\mathcal{A}$ checks his active session with identifier $\mathcal{I}_A$. If found, $\mathcal{A}$ does the following steps. 1) Verify that $Y \in \mathbb{G}^*$, and check whether $(T_A^* - T_B^*) \geqslant \mathit{\Delta T}$ or $L_B^* \not\equiv L_A^*$. If so, indicating the session has expired or the region is invalid, $\mathcal{A}$ cancels this session; otherwise proceeds to next steps. 2) Compute $x = H_1(a, \hat{x})$ and verify that $\beta = V^{d_{B_1}} U^{d_{B_2}}$. 3) Compute $\lambda_1 = Y^a$, $\lambda_2 = B^x$, $\lambda_3 = Y^x$, $\lambda_4 = \beta^{xd_{A_1}+ad_{A_2}}$, $\lambda_5 = {D_{A_1}}^{d_{B_1}}$, and $\lambda_6 = {D_{A_2}}^{d_{B_2}}$. 4) Compute $(k, k') = H(\mathcal{A}, \mathcal{B}, X, Y, M, N, U, V, \alpha, \beta, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6)$. 5) Destroy $\hat{x}, x, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$, and λ_6. Then verify that B_{MAC}= $H_{MAC}(k', \vec{\mathcal{R}}, \mathcal{B}, \mathcal{A}, Y, X, N, M, V, U, \beta, \alpha, L_B^*)$. Compute $A_{MAC} = H_{MAC}(k', \vec{\mathcal{I}}, \mathcal{A}, \mathcal{B}, X, Y, M, N, U, V, \alpha, \beta, L_A^*)$. Destroy k'. 6) Send $\mathcal{M}_A = (\mathcal{B}, \mathcal{A}, X, Y, M, N, U, V, \alpha, \beta, B_{MAC}, A_{MAC}, T_A^*, L_A^*)$ to $\mathcal{B}$. 7) Update the session identifier to $\mathcal{I}_A$ = $(\mathcal{A}, \mathcal{B}, \vec{\mathcal{S}}, X, Y, M, N, U, V, \alpha, \beta, B_{MAC}, A_{MAC}, T_A^*, L_A^*)$, and complete the key agreement session by accepting k as the session key for data transfer.

4. Upon receiving $\mathcal{M}_A$, user $\mathcal{B}$ checks his active session with identifier $\mathcal{I}_B$. If found, then $\mathcal{B}$ does the following steps. 1) Verify that A_{MAC} = $H_{MAC}(k', \vec{\mathcal{I}}, \mathcal{A}, \mathcal{B}, X, Y, M, N, U, V, \alpha, \beta, L_A^*)$. Check whether $(T_B^* - T_A^*) \geqslant \mathit{\Delta T}$ or $L_A^* \not\equiv L_B^*$. If so, the session could be expired or the region be invalid, and the session should be canceled; otherwise proceed. 2) Destroy k'. 3) Update the session identifier to $\mathcal{I}_B$ = $(\mathcal{B}, \mathcal{A}, \vec{\mathcal{R}}, X, Y, M, N, U, V, \alpha, \beta, B_{MAC}, A_{MAC}, T_B^*, L_B^*)$ and complete the key agreement session by accepting k as the session key for data transfer.

Correctness: The correctness of above computations can be proved as follows: $\lambda_4 = \alpha^{yd_{B_1}+bd_{B_2}}$, where α = $N^{d_{A_1}} M^{d_{A_2}}$ $= {D_{A_1}}^x {D_{A_2}}^a$, β = $V^{d_{B_1}} U^{d_{B_2}} = {D_{B_1}}^y {D_{B_2}}^b$. λ_5 = ${D_{A_1}}^{d_{B_1}}$

= ${D_{B_1}}^{d_{A_1}}$, and λ_6 = ${D_{A_2}}^{d_{B_2}}$ = ${D_{B_2}}^{d_{A_2}}$.

Remarks: λ_1, λ_2, and λ_3 originate from the *NAXOS-C* protocol, which is also a combined security model and proved to be secure in both pre and post-specified peer models [11], even though the adversary can obtain the knowledge about the ephemeral public keys and secret information (See [11] for the proof and details). Furthermore, these three encrypted values combine both ephemeral session keys and certificateless long-term keys. M, N, U, V are based on the encapsulated Boneh-Franklin session keys [14], and α, β are derived from the Krawczyk DCR signature and strong twin Hashed DDH assumption. It should be pointed out that λ_4 comes from the TDH problem proposed by Cash [12] and the Krawczyk DCR signature [13]. In this paper, the design of λ_4 has two main purposes: one is that the DCR signature is embedded in it in order to self-compute (verify) signatures from two parties $\mathcal{A}$ and $\mathcal{B}$; the other is to setup the combination and binding of all elements of a subgroup, such as ephemeral session keys, certificateless long-term keys, non-interactive ID-based keys, and the master public key from PKC. λ_5 and λ_6 are based on the non-interactive ID-based key agreement scheme of discrete logarithm, which is similar to the bilinear pairing scheme proposed by Lippold [7], but more computation efficient.

C. Secure Upload for Drive-thru Internet

With the proposed certificateless key agreement protocol, vehicles approaching an RSU can initiate the establishment of session keys after receiving the RSU's (relayed) beacon messages. Once the session key is accepted, vehicles and the RSU can ensure the authenticity of each other and the confidentiality and integrity of their communication. In this paper, we focus on the scenario where vehicles mainly upload to the RSU over a shared broadcast channel and have to compete with each, thus attracting possible attacks. Furthermore, we view the vehicle arrivals of a highway segment as a Poisson process, which is based on realistic traffic characteristics and verified by statistical analysis of empirical data [17].

Since each of the four sessions during the key agreement process will introduce additional computation overhead at vehicles and the RSU, as well as relayers, secure communication will be delayed until the session key is accepted, thus reducing the available communication time window for each vehicle to the RSU. Our first goal is to find out how much performance degradation, in terms of the amount of data uploaded to the RSU, is due to the proposed security scheme. In order to do so, we have to implement the proposed scheme in a commodity platform reasonable for VANETs.

Our next goal is to investigate whether it is possible to mitigate the security overhead with the properties of VANETs. Since vehicles can relay beacon and key agreement messages, vehicles can possibly initiate the key agreement process before they actually reach the RSU's direct communication range, thus leaving the short-range, high-speed communication time window intact with the RSU. However, the further away vehicles are from the RSU, the less likely messages can be relayed successfully even with vehicle mobility, and even if likely, more relays and thus security overhead are involved to reach vehicles further away from the RSU. We will investigate these two questions in the next section.

V. Security Analysis and Performance Results

In this section, we demonstrate that the proposed protocol is secure, practical and feasible by analyzing its security property, computation overhead and communication overhead. The experimentation platform used in this paper is a commodity PC with a dual-core 3.4 GHz CPU and 2 GB RAM.

A. Security Analysis

Recently, there are a number of new security threats emerged in the domains of CL-PKE and IB-PKE. Due to the page limit, in this section we only analyze some security properties closely correlated to our design objectives. In our scheme, when the adversary attempts to impersonate $\mathcal{A}$ to another user, we can detect and replace those keys, and some temporary intermediate results are destroyed and recomputed when they are needed again, in spite of the increase in computation cost. Furthermore, session identifiers and MAC values are updated regularly, which is an extra layer of security and can further minimize the damage of falsifying, altering or modifying metadata parameters as soon as possible. T_A^* and T_B^* can resist against time difference-based reply attacks so as to preserve the freshness of the key session, and can be treated as X and Y in encrypted forms (we do not give details here due to the page limit). On the other hand, L_A^* and L_B^* aim to resist against location difference-based reply attacks.

This model combines the above properties for better security, is a balanced approach with efficiency considerations, and is able to resist against possible attacks in insecure and intermittently connected environments such as VANETs. For the sack of page limit, we omit the theoretical proof of the security protocol, which will be discussed in the full version.

In the following, we also show that our proposed scheme is secure against ten common types of security attacks [7], [15], as the desired security properties for certificateless key agreement protocols. Due to the page limit, we list possible attacks and corresponding resilience properties in Table II (please see [7] and [15] for the details of the attacks).

Privacy issues: As discussed previously, VANET privacy is a conditional privacy-preserving issue. Users $\mathcal{A}$ and $\mathcal{B}$ can utilize their independent pseudonymous keys embedded with secret random nonce values, r_A and r_B, to setup the session keys and form chains of authentication. During the key agreement process, no information related to their identities is leaked. In addition, ephemeral keys and other one-time values, such as x, k', A_{MAC}, L_A^*, are produced and chosen independently. Hence, it is very difficult to identify which vehicle has ever sent which message, or where or when it has downloaded data [3]. However, once it is necessary to track down the true identity of one vehicle, CA can utilize the session receipts received by RSUs or other users to identify and verify the true ID rather than the pseudonymous ID.

TABLE II
ATTACKS AND RESILIENCE PROPERTIES

Types	Resilience Properties
Impersonation attack [7]	long-term private key a or b
KCI attack [15]	e.g., N, α, CDH or DDH etc.
Known session key attack [7]	ephemeral key and one-time values, e.g., x, k', A_{MAC}, T_A^*, etc.
Perfect forward secrecy [15]	ephemeral key and one-time values, e.g., same as above.
Partial forward secrecy [15]	ephemeral key and one-time values, e.g., same as above.
Weak perfect forward secrecy [15]	ephemeral key and one-time values, e.g., same as above.
Unknown key-share attack [7]	session identifier and MAC values, CDH or DDH, etc.
Leakage of ephemeral keys [7]	e.g., $M, U, \alpha, \beta, \lambda_4$, CDH, etc.
KGC forward secrecy [7]	ephemeral key and long-term values, e.g., x, α, λ_4, etc.
KGC leakage of eph. keys [7]	long-term private keys, a and b

TABLE III
COMPUTATION OVERHEAD

Operations	Length (bits)	Timing (ms)		Forms
		Naive	*Comba*	
Add/Mod$_{1024}$	1024	0.0003		$a \pm b$ / $a \oplus b \mod N$
Mul$_{1024/\text{prime}}$	1024	0.0053		$a \cdot b \bmod N$ / $p \cdot q$
Mod$_{\text{exp}^1}$	1024	11.7627	2.6209	$a^b \bmod N$
Mod$_{\text{exp}^2}$	1024	11.6608	2.6206	g^e, $g \in \mathbb{G}$
	512	5.8933	2.0469	
	160	1.9777	0.6436	
SHA	512	0.0328		HASH with γ
	256	0.0069		
	160	0.0066		
Point$_{\text{mul}}$	512	4.7719	1.1031	aP, P over elliptic curve $E(\mathbb{F}_p)$ or $E(\mathbb{F}_{2^m})$
	256	2.5328	0.5594	
	160	1.6375	0.3546	
$E(\mathbb{F}_p)$ Tate	155	62.5125		MNT $k = 4$
	160	155.2660		low-rho $k = 4$
	160	233.3280		Cocks-Pinch $k = 4$
$E(\mathbb{F}_p)$ Ate	155	165.9516		MNT $k = 4$
	160	156.9070		Ate by Freeman etc.
$E(\mathbb{F}_{2^m})$ η_T	*fastest*	70.8590		η_T by Berreto etc.

B. Computation Overhead

Without loss of generality, we evaluate the computation efficiency through the original source code in the Miracl library [16], which is a well-known free software for non-commercial use and implements efficient Big Number Cryptography. We select some representative examples, with a 160-bit group, 1024-bit security and the embedding degree of $k = 4$, to compare the computation overheads. We obtain the timing costs listed in Table III as the average value of 1,000 computations. Note that precomputations are adopted in the $Comba$ method and there is no precomputation in the $Naive$ method in Miracl [16]. In addition, three typical elliptic curves are adopted in our experiment, which are $E(\mathbb{F}_p)$ Ate pairing, the $E(\mathbb{F}_p)$ Tate pairing of Cocks-Pinch $k = 4$ curve, and $E(\mathbb{F}_{2^m})$ η_T the fastest known pairing [16], respectively. In order to simplify the implementation, we can set ΔT to 100 ms, which it is almost half of the estimated total computation time in the worst case, without further considering the media access delay and packet loss.

Here, we evaluate and compare the performance of our protocol with the Lippold's protocol [7], which offer similar security and privacy properties even though different schemes were adopted, i.e., discrete logarithm in our protocol and elliptic curve cryptosystem in theirs. We measure the computation overhead of the following seven different operations based on the Miracl library [16]. Denoted by $a, b, e \in \mathbb{Z}_q^*$, large prime number p and q, P over elliptic curve $E(\mathbb{F}_p)$ or $E(\mathbb{F}_{2^m})$, the operations are 1) addition (Add$_{1024}$) or modular (Mod$_{1024}$) of 1024-bit big numbers; 2) multiplication (Mul$_{1024}$) of 1024-bit big numbers or multiplication (Mul$_{\text{prime}}$) of large prime numbers; 3) modular exponentiation (Mod$_{\text{exp}^1}$) of big numbers; 4) modular exponentiation (Mod$_{\text{exp}^2}$) of large prime numbers; 5) SHA-1 HASH functions, and γ is its security parameter; 6) point multiplication (Point$_{\text{mul}}$) of a variable point on elliptic curves; 7) pairing implementation (Pairing) over elliptic curves. The overall computation overheads obtained on a commodity platform are listed in Table III.

With respect to computation overhead, our protocol shows some desired performance advantages and reflects the practical feasibility of the proposed approaches. According to our protocol and the results in Table III, the total time required for the key agreement process, excluding all precomputation time, includes the time for the three-way handshake, sixteen 1024-bit modular exponent Mod$_{\text{exp}^1}$ operations, four 1024-bit big number multiplication Mul$_{1024/\text{prime}}$ operations, and six 512-bit hash operations. The total computation time, which is estimated in worst case, is 188.42 ms according to Table III. On the other hand, for the $Comba$ method, the total computation time is 42.15 ms. The experimentation results suggest that the workload of Lippold [7] takes nearly 16, 25 and 7 times (with the fastest known pairing) more than our protocol with the $Naive$ method, indicating our protocol is much faster than Lippold's. The tradeoff is desired, due the short contact time between vehicles and RSUs in VANETs.

C. Communication Overhead

We adopt the proposed key agreement protocol for a drive-thru Internet along a highway segment. The communication range is $R = 30$ m at 11 $Mbps$, a reasonable setting for IEEE 802.11-based VANETs. The vehicles reach the RSU in a Poisson arrival process with rate $\lambda \in [0.005, 0.12]$ vehicles per meter, reflecting free-flow to jammed scenarios, at speed $v_f(1 - \lambda/\lambda_{\text{jam}})$, where free-flow speed $v_f = 120$ km/hr or 33.33 m/s and $\lambda_{\text{jam}} = 0.12$. The drive-thru time is $2R/v$.

According to our key agreement protocol, the three-way handshake has to be initiated by the vehicle and finished before the secure upload can happen. With the model we built in [17], we can obtain the frame service time, including both media access delay and transmission and propagation delay, of each key agreement message, given the number of competing vehicles accessing the RSU at the same time. Adding this to the computation overhead obtained above, we can calculate the total overhead due to the key agreement protocol, thus obtaining the amount of data updated with or without the security measures introduced and the properties achieved. In this paper, we mainly investigate the one-hop scenario and show

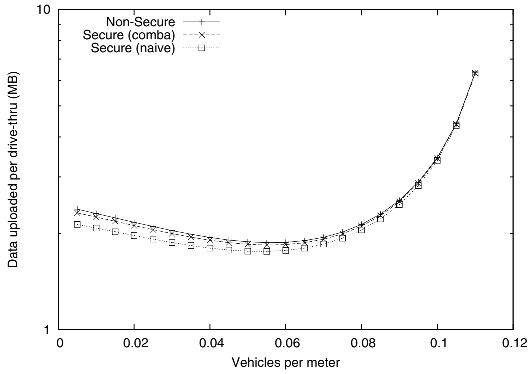

Fig. 2. Data uploaded per drive through with and without security

the possible extension to multiple hops. By first showing the basic idea and the feasibility of multi-hop relaying with some preliminary results, we will continue to determine how to form a secure and dynamic authentication chain and how to improve their practical and achievable performance in intermittently connected VANETs through multi-hop communications.

Figure 2 shows the performance results. When there are a few vehicles on the road, the drive-thru speed is fast and the time is short, so less upload contention encountered by vehicles for a high throughput; when there are a lot of vehicles, the drive-thru speed is much lower, so vehicles have more time to upload, but at a lower throughput. Thus when vehicle densities are very low or high, the amount of uploaded data per drive-thru is larger. With the *Naive* method, vehicles will upload about 10% less data at very low vehicle density (or high vehicle speed), and with the *Comba* method, that gap is reduced to only 2.5%, but with high density, the performance degradation due to security is negligible. We believe this is a reasonable cost for the security properties achieved here. Note that although with a high vehicle density more data can be uploaded to an RSU, the drive time to reach the RSU is also increased significantly due to a very low vehicle speed, which actually indicates an undesired operating region.

To further improve the performance with security, one approach is to separate the security association and data transfer, by allowing the key agreement process to start before reaching the direct communication range of the RSU, through multi-hop relaying. Since the key agreement messages are relatively much smaller than the actual data to be uploaded, the extra communication overhead is minimum, but vehicles can fully utilize their drive-thru time for high-speed secure data transfer. For example, according to [17], when $\lambda = 0.05$ (i.e., inter-vehicle distances are exponentially distributed and on average vehicles are $20\ m$ apart), vehicles traveling at $19.44\ m/s$ have 47% chance to reach the RSU $40\ m$ away in 3 hops on average (note that due to the exponential distribution, the number of hops needed to reach the RSU is larger than that with a uniform distribution or fixed spacing among vehicles). The frame service time is $4.448\ ms$, even through there are only 3 vehicles covered by the RSU on average. This translates into a $166.5\ ms$ relayed key agreement overhead with the *Comba* method over three hops and $605.3\ ms$ with the *Naive* method, respectively. If the relay is successful, it allows vehicles to finish the relayed key agreement with the *Comba* method before they are entering the $30\ m$ direct communication range of the RSU at $514.3\ ms$, but this process cannot finish with the *Naive* method, and definitely cannot finish with Lippold's protocol due to its high overhead. Due to the page limit, we omit the detailed results here.

VI. Conclusions

In this paper, we presented some important security issues for drive-thru Internet services, proposed a secure and practical key agreement protocol, and implemented and evaluated it on a commodity platform. The performance results showed that the proposed protocol achieves our objectives with a reasonably low cost, and we also investigated the ways to further reduce the cost. Our future work will focus on the fast re-authentication, reduced two-way handshake, and other crypto protocols to make the computation even more efficient.

Acknowledgment: This work is supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada, and Jun Song is also supported in part by a grant from China University of Geosciences to visit the University of Victoria where the work is done.

References

[1] Dedicated Short Range Communications (DSRC), [online]. http://www.leearmstrong.com/DSRC/DSRCHomeset.htm

[2] IEEE Std. 1609.2-2006, IEEE Trial-Use Standard for Wireless Access in Vehicular Environments- Security Services for Applications and Management Messages, 2006.

[3] R. Lu, X. Lin, H. Zhu, P. Ho, X. Shen, "ECPP: Efficient Conditional Privacy Preservation Protocol for Secure Vehicular Communications," in Proc. IEEE INFOCOM, 2008, pp. 1229-1237.

[4] M. S. Bargh, R. J. Hulsebosch, E. H. Eertink, a. Prasad, H. Wang, and P. Schoo, "Fast authentication methods for handovers between IEEE 802.11 wireless LANs," in Proc ACM WMASH'04, 2004, pp. 51.

[5] A. W. Dent, "A survey of certificateless encryption schemes and security models," Int. J of Information Security, vol. 7, 2008, pp. 349-377.

[6] L. Chen, C. Kudla, "Identity based authenticated key agreement protocols from pairings", Int. J of Information Security. 2007. pp. 213-241.

[7] G. Lippold, C. Boyd, N. Gonzalez, "Strongly Secure Certificateless Key Agreement", In: Pairing 2009, pp.12-14 August, Palo Alto, CA, USA

[8] L. Zhang, Q. Wu, A. Solanas, and J. Domingo-Ferrer, "A Scalable Robust Authentication Protocol for Secure Vehicular Communications," IEEE Trans on Vehi Tech, vol. 59, 2010, pp. 1606-1617.

[9] Y. Hao, J. Tang, Y. Cheng, and C. Zhou, "Secure data downloading with privacy preservation in vehicular ad hoc networks", in Proc. IEEE ICC, Cape Town, South Africa, May. 23-27, 2010.

[10] M. Hou, Q. Xu, "Secure and Efficient Two-Party Authenticated Key Agreement Protocol from Certificateless Public Key Encryption Scheme", 5th Int'l Joint Conf on INC, IMS and IDC, 2009, pp. 894-897.

[11] A. Menezes, B. Ustaoglu: "Comparing the Pre- and Post-specified Peer Models for Key Agreement", Technical Report CACR 2008-07, University of Waterloo (2008), http://www.cacr.math.uwaterloo.ca.

[12] D. Cash, E. Kiltz, and V. Shoup, "The Twin Diffie-Hellman Problem and Applications". Proceedings of IACR EUROCRYPT, 2008, pp. 28-40.

[13] H. Krawczyk, "HMQV: A High-Performance Secure Diffie-Hellman Protocol". Crypto 2005, pp. 546-566.

[14] D. Boneh, M. Franklin, "Identity-Based Encryption from the Weil Pairing," Computer, vol. 32, 2003, pp. 586-615.

[15] C. Swanson, D. Jao, "A Study of Two-Party Certificateless Authenticated Key-Agreement Protocols," INDOCRYPT, 2009, LNCS 5922, Springer Berlin/Heidelberg, 2009, pp. 57-71.

[16] MIRACL: Multiprecision Integer and Rational Arithmetic C/C++ Library, [online]. http://http://www.shamus.ie.

[17] Y. Zhuang, V. Viswanathan, J. Pan, and L. Cai, "Upload Capacity Analysis for Drive-Thru Internet," Technical Report, VT-2010-09-4, Waterloo Engine, 2010.