Prioritized WAVE-based Parking Assistance with Security and User Anonymity

Subir Biswas
Department of Computer Science, University of Manitoba, Winnipeg MB, Canada R3T 2N2
Email: bigstan@cs.umanitoba.ca

Jelena Mišić
Department of Computer Science, Ryerson University, Toronto ON, Canada M5B 2K3
Email: jmisic@scs.ryerson.ca

***Abstract*—We present a secure, privacy-preserving car parking assistance application using priority-based vehicular communications. The proposed technique utilizes two modified Elliptic Curve algorithms for vehicular message authentication between parking vehicles and corresponding infrastructure. We address the major challenges associated with VANET security and user privacy. The standard Wireless Access in Vehicular Environment (WAVE) protocol suite has been used to construct our signature mechanism, application data secrecy, and message integrity. Security analysis proves the resilience of the scheme against all known attacks. We also investigate network performances in order to justify the effectiveness of the scheme for practical use.**

I. Introduction

Wireless Access in Vehicular Environment (WAVE) is turning out to be an intriguing avenue of research and innovation with traffic-safety and other user-friendly applications that assist the driver of a vehicle, providing a better traffic atmosphere for safe and efficient driving. As such, the primary intention of a Vehicular Ad hoc Network (VANET) was to reduce the number of traffic accidents caused due to potential driver errors on roads and highways.

A VANET consists of two major components. An on-board unit (OBU) is mounted in a vehicle, while road-side units (RSUs) are normally installed at the roadside locations. An OBU is used for transmitting and receiving data or other application messages to/from other OBU(s), or an RSU. OBUs and RSUs are IEEE 802.11p [1] wireless devices enabling WAVE communication schemes in a VANET.

Typical VANET features can be extended to provide unconventional collaborations like parking assistance for registered users. According to the WAVE standards, the DSRC (Dedicated Short Range Communications) enables WAVE devices to communicate with each other within the range. Also, each entity in a VANET broadcasts a periodic (every 100-300ms) safety message where the information about the vehicle's current location, speed, and road-conditions are disseminated. This two forms of communications in VANET can be used as the basis of an efficient parking system for an automated parking facility.

Intuitively, this application would require two-way communications between an RSU and a vehicle's on-board unit (OBU). The corresponding RSU must inform the vehicles about the current status of the parking facility while at the same time, an OBU should be able to ask for a reservation of a parking spot in the parking facility.

Due to the typical wireless features along with ad hoc nature and high volume traffic with variable node density, a VANET is always at the risk of being exposed to several different types of malicious behaviors initiated either by an internal, or an external adversary. Obviously, a VANET scheme for safe parking assistance would not be socially accepted unless it is reliable, secure and scalable. Therefore, potential risks involved with VANETs have to be addressed at least with mutual authentication, confidentiality, and confirmation of message integrity. The scheme should prevent the network from all known security attacks in order to be trusted by a VANET user.

The most common solution to the above requirements is to have a suitable security scheme that can provide confidentiality and authentication of the delivered transmission between any pair of VANET entities. However, identity of a user should not be exposed in the process as nobody would like to be traced later on through their transmitted messages. Moreover, the actual identity of a user should be recoverable by the high-level authority when there is a traffic dispute. Thus, the provided anonymity has to be conditional in our approach.

Lu et al. [2] proposed a scheme that supports intelligent parking of vehicles using the spatiotemporal properties of available parking space and VANET mobility. While the scheme addressed some fundamental security and privacy requirements of a VANET, the approach is basically dependant on bilinear-pairing based signature and verification techniques [3] with some strong assumptions like the accuracy of radio-signal strength and the use of tamper-proof devices. Although, an RSU is installed at the roadside without much physical protection and surveillance, in Lu et al.'s work, RSUs are considered as non-compromisable.

A bilinear-pairing approach is expensive in terms of time and computation complexity, and in most cases it is very difficult to generate the appropriate combination of parameters for pairing-based approaches as the most typical and frequently made assumptions are not feasible in practice [4].

doi:10.4304/jcm.7.8.577-586

A. Our Contribution

In our approach, we use a modified version of Elliptic Curve Digital Signature Algorithm (ECDSA) [5] to provide message authentication and user anonymity in a parking facility. ECDSA is widely accepted, fast, and lightweight mechanism for generation and verification of digital signatures. It is also recommended by the WAVE standards for security services [6]. Unlike other similar PKI-based message authentication approaches (e.g. [7], [8], [9], [10] and [11]), our scheme doesn't depend on trusted third-party certificates to provide source authentication; instead, it uses the geographical location of an entity to validate a received VANET message.

We design two different versions of modified ECDSA to incorporate authentication of infrastructure-originated VANET messages, as well as OBU-generated messages with user privacy.

To provide authentication in infrastructure-to-vehicle (I2V) communications, we introduced a vehicular authentication scheme with an identity-based (ID-based) ECDSA proxy signature mechanism in which the corresponding road-side unit signs the message content on behalf of the trusted central authority. The actual identity of the trusted authority and the physical location of the signing road-side infrastructure ensure the authentication of the delivered message.

On the other hand, a vehicle-to-infrastructure (V2I) communication in our work is made trustworthy with a different application of ECDSA signature. Vehicle-generated messages are authenticated by the road-side infrastructures as well as other vehicles. However, the actual identity of a vehicle is not revealed in the process; and hence, user privacy is preserved.

Our contribution in this paper also includes the development of a network simulator using NS-2 with different IEEE 802.11p access categories for investigating the performance of a WAVE-enabled car parking facility.

B. Organization

We organize the rest of the paper in the following fashion. Section II summarizes the fundamental cryptographic primitives like proxy signature mechanism, ECDSA fundamentals for the better understanding of our security scheme. The overview of the proposed system, related assumptions, as well as commonly used notations are listed in Section III. Section IV illustrates our WAVE-enabled parking assistance scheme in details. Security and anonymity issues are analyzed in Section V, while Section VI briefly delineates the simulation results. Section VII contains the concluding remarks of the paper.

II. Preliminaries

In this section, we briefly describe the fundamentals of the proxy signature approach as well as the elliptic curve digital signature algorithm (ECDSA) scheme.

A. Proxy Signature

Definition 1. *Proxy Signature: Proxy signature refers to a variation of digital signature that designates an entity (called a proxy signer) to sign a message on behalf of the original signer.*

Definition 2. *Partial Delegation: The original signer derives a secondary secret key from a primary secret such that it is computationally infeasible to retrieve the primary secret from the knowledge of the secondary secret key. The primary secret is kept with the original signer, while the derived secret key is delivered to the proxy signer in a secure way.*

1) Review of Proxy Signature: An original signer, a proxy signer, as well as a verifier are involved in preprocessing, signature, and verification processes of a proxy signature scheme. The following steps are generally followed:

i. *Proxy derivation:* An original signer generates a proxy key from the original secret key as required by *partial delegation* based proxy signature.
ii. *Proxy delivery and verification:* Original signer delivers the proxy tuple to the proxy signer. A proxy signer can verify the proxy tuple using a verification equation.
iii. *Signing:* A proxy signer uses any ordinary digital signature scheme to sign a message on behalf of the original signer. It uses the proxy key (derived by the original signer) as the secret key for signing the message.
iv. *Verification of the proxy signature:* Upon receiving the proxy signature, an end user derives a new public key from the original signer's public key. This new public key is used for verification of proxy signature using the verification method of the corresponding signature scheme.

B. On ECDSA

We use an elliptic curve over a finite field for our scheme. Below, we discuss some of the preliminary issues with the elliptic curve and ECDSA.

Definition 3. *A finite field $\mathbb{F}_p$ is a finite set of p elements along with addition and multiplication operations on $\mathbb{F}$. The number of elements is denoted as the order of the finite field. There exists a finite field of order q if and only if q is a prime power, and on the other hand, if q is a prime power, then there exists only one finite field of order q denoted by $\mathbb{F}_q$.*

Definition 4. *An Elliptic Curve E over a finite field $\mathbb{F}_p$ is defined in the form of the following equation:*

$$y^2 = x^3 + ax + b, \tag{1}$$

where a prime $p > 3$; $a, b \in F_p$, and $4a^3 + 27b^2 \not\equiv 0 (mod\ p)$. The set of elements of the Elliptic Curve $E(\mathbb{F}_p)$ consists of the points (x, y) where $x \in \mathbb{F}_p$ and $y \in \mathbb{F}_p$. A point at infinity $\mathcal{O}$ together with the set of points $E(\mathbb{F}_p)$ identifies an elliptic curve.

Note that the addition, multiplication, and inversion operations on an elliptic curve points are different from ordinary binary operations. Please refer to [5] for the detailed description of the mentioned operations.

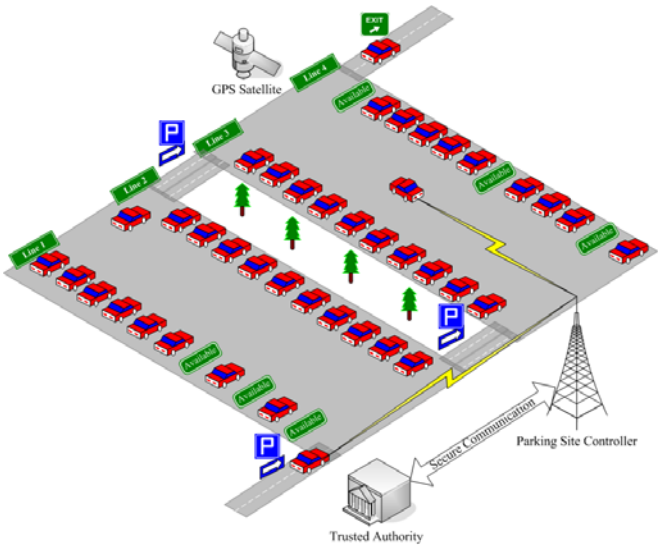


Fig. 1. System model for our proposed WAVE-enabled parking facility.

1) ECDSA Domain Parameters: The domain parameters of Elliptic Curve Digital Signature Algorithm (ECDSA) require an Elliptic Curve E over a finite field of size q, and a base point $G \in (\mathbb{F}_q)$. Value q is chosen as a prime power p^t where p is a prime number, and t is a positive integer. We choose, $t = 1$, thus $p = q$. Also, as indicated in eqn.(1), two field elements a and b are chosen, where, $a, b \in (\mathbb{F}_q)$. All these parameters could be shared by the entities or by some specific user depending upon the ECDSA configuration.

2) ECDSA Summary: A signer of message m follows the steps:

i. *Key Pair Generation:* Select a random number $d \in_R \mathbb{Z}_q^*$ to compute $Q = dG$; where G is a base point of the elliptic curve $E(\mathbb{F}_p)$.

ii. *Signature Generation:* The signer computes $(x_1, y_1) = kG$; where k is a random number and $1 \leq k \leq q$. The signer then computes $r = x_1 mod\ q$ and $s = k^{-1}(SHA1(m) + dr) mod\ q$; where if $r = 0$ or $s = 0$, the signer aborts the current operation and restarts the procedure. (r, s) represents the signature for message m.

iii. *Verification:* A verifier first checks if r and s are in the interval $[1, q-1]$. It then does the following computations:
$w = s^{-1} mod\ q$
$u_1 = SHA1(m)w\ mod\ q$
$u_2 = rw\ mod\ q$
$X = u_1 G + u_2 Q$
If $x \neq \mathcal{O}$ and $v = r$, the verifier accepts the signature, otherwise the delivered message is rejected.

III. System Design

A. System Overview

As indicated earlier, the ad hoc nature and frequently changeable topology along with variable node density contribute to the security risk of a vehicular ad hoc network.

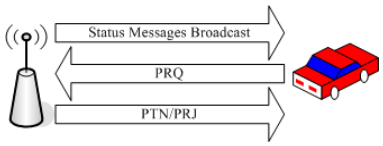


Fig. 2. System overview of WAVE-enabled parking facility.

Exposure of application data, data inconsistency in communication, identity forgery, and unleashing the original identity of a user may jeopardize the prospect of a vehicular network. Therefore, we take into account, the fundamental security and privacy aspects during the design process of our secure parking assistance scheme which provides user-data confidentiality, message integrity, and message authentication with conditional user-anonymity.

Our system design includes a globally trusted authority (TA), a parking site RSU called *parking site controller* or PSC, and a secure network layer communication between the TA and the PSC. We assume that all the vehicles are equipped with global positioning systems (GPS), so that their OBUs are aware of their corresponding geographical coordinates. Figure 1 illustrates our proposed model for a WAVE-enabled car parking facility.

When in operation, a PSC disseminates periodic status messages which include the current status of the corresponding parking facility. A recipient OBU would accept or reject a periodic message following a signature verification process.

Algorithm 1 provides an identity-based [3] proxy signature [12], [13] mechanism for the authentication of infrastructure originated messages in VANET. The location information of the PSC in a parking facility is used as the identity for the signer. TA generates a message m, produces a delegation proxy key $s_{i,m}$ associated to the message m and PSC_i; using the generated message, expiry information (t_m) and the geographical tolerance (a_m). TA then securely deliver the key along with other credentials to the proxy signer PSC_i which would sign and deliver the message on behalf of TA.

The mechanism also allows a recipient OBU to verify the received signature using its own GPS location, and other received credentials.

While delivering on periodic safety messages containing a vehicle's speed, location, and road-safety information, an OBU nearby a parking facility may request for a parking space by sending a *parking reservation request* (PRQ) to the PSC. Up on successful reception of a PRQ, PSC stores a copy of it, and if a parking space is available, it would grant a *parking token* (PTN) to the requesting OBU. A parking token is an approval of the request made by the vehicle, which can be used during the admission and toll collection process in the parking facility with some suitable applications.

A parked vehicle's OBU periodically updates its parking status by sending PRQs to the PSC on a regular interval. PSC acknowledges the OBU's update by sending a fresh PTN each time it receives the periodic update from an OBU.

If there is no parking space available, PSC notifies the

requesting vehicle with a *parking rejection* (PRJ) message which may also contain information like the other nearest parking lot, or an estimated waiting time for a probable parking spot allocation. Figure 2 outlines the communications needed by the proposed parking assistance scheme.

Anonymity in vehicular communications is vital since a user would not like to be traced by the messages it sends and receives over the period of time. We provide an anonymous authentication mechanism for VANET users in Algorithm 2. This mechanism would be used for OBU's periodic safety message signature, as well as other application message signature. This mechanism proves conditional anonymity for VANET users.

B. Notations

Table I lists the notations used in the rest of our work.

TABLE I
NOTATIONS

Component	Description
TA	global trusted authority
Q	master public key
q	a large random prime number
x	system's master secret; $1 < x < q$
G	base point on the elliptic curve $E(\mathbb{F}_p)$
k_o	random secret associated to the TA
k_p, k_c	session parameters
k_i, k_g	random secrets $< q$
$H(.)$	hash function $H(.): \{0,1\}^* \rightarrow \mathbb{Z}_q^*$
loc_z	GPS position information of entity z
m	a message to be signed and delivered
a_m	geographical tolerance
t	current system time (rounded up)
t_m	message m's expiry information

C. Geographical Tolerance

Upon receiving a message m, an OBU compares its own location (obtained from GPS) with the origin of m taking into account the tolerance value a_m specified in the message.

Let the current GPS position data include l-bits long x_{gps} and y_{gps} for latitude and longitude information respectively of an OBU. The validation of geographical scope is done at an OBU in the following manner:

i. Take $l - a_m$ most significant bits from both x_{gps} and y_{gps} and replace the remaining bits with 0s. The outcome would be x'_{gps} and y'_{gps}.
ii. Return (x'_{gps}, y'_{gps}) as the new location information.
iii. Verify if location (x'_{gps}, y'_{gps}) matches with the source location of the received message.

IV. PROPOSED SCHEME

Vehicles come across each other for a very short period of time. Again, the chance of having a specific vehicle in other OBU's communication range is quite low. Hence, pre-authentication of users is not feasible in VANETs. We deploy two different algorithms to sign periodic safety, and other application messages for two categories of vehicular message authentication (i.e. I2V and V2I message authentications).

A. An ID-based Proxy Signature in VANET With ECDSA

In an anticipated VANET model, an emergency/road-safety application message is generated by a trusted central authority (e.g. department of transportation), while the issued message is signed and delivered to end users (OBUs) by corresponding road-side units (RSUs) on behalf of the originator of the message.

The following steps illustrate ID-based proxy signature with ECDSA for VANET applications.

1) Key Setup: TA generates a system secret x; where $1 < x < q$ and computes

$$Q = xG. \tag{2}$$

This Q is a public parameter, and is preloaded to all possible verifiers in the network.

TA then randomly picks k_o; where $1 < k_o < q$ for the original signer to compute

$$R_o = k_o G. \tag{3}$$

For each RSU_i under a particular TA decides a random number k_i, and determines R_i using the following equation.

$$R_i = k_i G. \tag{4}$$

The actual identity of the original signer TA (ID_o) and the location info for proxy signer RSU_i (loc_i) are public, and all the verifier OBUs in the vicinity of the RSU are aware of their own locations from GPS. TA generates message m along with its expiry information t_m, and the geographic tolerance a_m. It then computes,

$$h_{i,m} = H(loc_i || ID_o || m || t_m || a_m). \tag{5}$$

The proxy key of RSU_i for a message m is computed at the TA as

$$s_{i,m} = (k_i + h_{i,m} x) k_o^{-1} \, mod \, q. \tag{6}$$

$(s_{i,m} || m || t_m || a_m)$ is delivered to the RSU_i (or, to the PSC_i) in a secure way. The proxy signer verifies the proxy key $S_{i,m}$ by checking if $R_i = (s_{i,m} R_o - h_{i,m} Q) \, mod \, q$ holds. If it does not, the proxy signer requests for a fresh proxy key from the original signer TA.

2) Payload Preprocessing: RSU_i (or, PSC_i) computes a session parameter k_p from its location info (loc_i) and system time (t).

$$\begin{aligned} k_p &= H(loc_i || t) \\ (x_p, y_p) &= k_p R_o \, mod \, q. \end{aligned} \tag{7}$$

3) Proxy Signature: Once the payload is processed, proxy signer RSU_i (or, PSC_i) generates the proxy signature using the following equation.

$$s_{p,i,m} = k_p^{-1} (H(m) + s_{i,m} x_p) \, mod \, q. \tag{8}$$

The proxy signature $(s_{p,i,m} || R_i)$ along with the tuple $(m || t_m || a_m)$ is delivered to the end user (an OBU).

4) Verification: Upon receiving the signature along with a message, a receiver OBU_j computes $h_{j,m} = H(loc_j||ID_o||m||t_m||a_m)$; where loc_j denotes the location information of OBU_j.

The following verification equation is checked.

$$(x_p, y_p) = (H(m)R_o + x_p(R_i + h_{j,m}Q))s_{p,i,m}^{-1} \bmod q. \quad (9)$$

The above equation holds if the signature is valid. Value x_p is independently computed at the verifier end using verifier OBU_j's location information loc_j and current system time t over eqn. (7). We assume that the rounded up location information of RSU_i (or, PSC_i) is same as the rounded up location information of OBU_j within the RSU's communication range. That is, $loc_i = loc_j$.

5) Correctness: We examine the proxy-key generation—, and proxy signature verification equations (eqn. (6) and (8)) in order to prove the correctness of our scheme.

Theorem 1. *If the proxy key verification equation holds, the proxy key is valid.*

Proof: The proxy key (eqn. (6)) is given as:
$s_{i,m} = (k_i + h_{i,m}x)k_o^{-1} \bmod q$.
Multiplying both sides by k_oG gives,
$s_{i,m}k_oG \bmod q = (k_iG + h_{i,m}xG) \bmod q$;
using eqn. (10), (3) and (4), yields: $s_{i,m}R_o = (R_i + h_{i,m}Q) \bmod q$
or, $R_i = (s_{i,m}R_o - h_{i,m}Q) \bmod q$.

This is proxy key verification equation used by proxy signer RSU_i (or, PSC_i). Hence, if the verification equation holds, the proxy key is valid. ■

Theorem 2. *If a proxy signature $(s_{p,i,m}||R_o||R_i)$ on a given message content $(m||t_m||a_m)$ is generated by a valid proxy signer PSC_i, it would be accepted by a verifier OBU_j.*

Proof: We investigate the proxy signature equation (eqn. 8). $s_{p,i,m} = k_p^{-1}(H(m) + s_{i,m}x_p) \bmod q$;
or, $s_{p,i,m} = (k_p^{-1}H(m) + s_{i,m}x_pk_p^{-1}) \bmod q$;
using eqn.(6) yields:
$s_{p,i,m} = (k_p^{-1}H(m) + (k_i + h_{i,m}x)k_o^{-1}x_pk_p^{-1}) \bmod q$;
or, $s_{p,i,m} = (k_p^{-1}H(m) + k_ik_o^{-1}k_p^{-1}x_p + h_{i,m}xk_o^{-1}k_p^{-1}x_p) \bmod q$;
dividing both sides by $s_{p,i,m}$, we get:
$1 = (k_p^{-1}H(m)s_{p,i,m}^{-1} + k_ik_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1} + h_{i,m}xk_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1}) \bmod q$;
multiplying by G on both sides, we get:
$G = (k_p^{-1}H(m)s_{p,i,m}^{-1}G + k_ik_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1}G + h_{i,m}xk_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1}G) \bmod q$
or, $G = (k_p^{-1}H(m)s_{p,i,m}^{-1}G + (k_iG)k_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1} + h_{i,m}(xG)k_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1}) \bmod q$;
replacing (xG) and (k_iG) using eqn. (10) and (4) yields:
$G = (k_p^{-1}H(m)s_{p,i,m}^{-1}G + R_ik_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1} + h_{i,m}Qk_o^{-1}k_p^{-1}x_ps_{p,i,m}^{-1}) \bmod q$
or, $k_pk_oG \bmod q = (k_oH(m)s_{p,i,m}^{-1}G + R_ix_ps_{p,i,m}^{-1} + h_{i,m}Qx_ps_{p,i,m}^{-1}) \bmod q$;
using again, eqn. (3), we get:
$k_pR_o \bmod q = (R_oH(m)s_{p,i,m}^{-1} + R_ix_ps_{p,i,m}^{-1} + h_{i,m}Qx_ps_{p,i,m}^{-1}) \bmod q$
or, $(x_p, y_p) = (R_oH(m)s_{p,i,m}^{-1} + R_ix_ps_{p,i,m}^{-1} + h_{i,m}Qx_ps_{p,i,m}^{-1}) \bmod q$
or, $(x_p, y_p) = (R_oH(m) + x_p(R_i + h_{i,m}Q))s_{p,i,m}^{-1} \bmod q$.

A verifying OBU receives a message within the communication range of the corresponding RSU. Since, OBU_j is in the communication range of RSU_i, we can say $loc_j = loc_i$ which implies $h_{i,m} = h_{j,m}$ from eqn. (5).

Therefore, the above equation yields as: $(x_p, y_p) = (R_oH(m) + x_p(R_i + h_{j,m}Q))s_{p,i,m}^{-1} \bmod q$.

We derived the verification equation from the proxy signature equation (eqn. (8)). Hence, if a proxy signature is generated by a legitimate proxy signer, the signature passes the verification process. ■

Algorithm 1 Location-based I2V message signature algorithm using ECDSA [5]

i. Key Initialization Module (at TA): Initialized with public parameters $\{q, G, ID_o, loc_i\}$, and secret parameters $\{x, k_o, k_i\}$. Following computations take place:
   $Q = xG$
   $R_o = k_oG$
   for $i = 1 \rightarrow n$ **do**
   $R_i = k_iG$
   end for
ii. Application Message Generator (at TA): Generates the status message m along with the expiry time t_m, and geographical tolerance a_m.
iii. Key Generator (at TA):
   $h_{i,m} = H(loc_i, ID_o, m, t_m, a_m)$
   $s_{i,m} = (k_i + h_{i,m}x)k_o^{-1} \bmod q$
   $(s_{i,m}, m, t_m, a_m)$ is delivered to PSC_i.
iv. Preprocessing (at PSC_i and OBU_j):
   $k_p = H(loc_{i,orj}, t)$
   $(x_p, y_p) = k_pR_o \bmod q$
v. Signature Generation (at PSC_i): PSC_i computes the following signature equation.
   $s_{p,i,m} = k_p^{-1}(H(m) + s_{i,m}x_p) \bmod q$
   Signature $(s_{p,i,m}, R_i)$, and (m, t_m, a_m) are transmitted to the OBUs.
vi. Verification (at OBU_j): An OBU_j computes-
   $h_{j,m} = H(loc_j, ID_o, m, t_m, a_m)$
   IF $(x_p, y_p) = (H(m)R_o + x_p(R_i + h_{j,m}Q))s_{p,i,m}^{-1} \bmod q$
   Accept m, ELSE Reject.

B. Location-based Anonymous Message Signature Algorithm

This anonymous message authentication mechanism is again based on four functional steps as detailed below.

1) Key Setup: TA chooses the system secret x, where $1 < x < q$; and computes

$$Q = xG. \quad (10)$$

Algorithm 2 Location-based anonymous signature algorithm using ECDSA

i. Key Initialization Module (at TA): Initialized with public parameters $\{q, G\}$, and secret parameters $\{x, k_g\}$.
   $Q = xG$
   $R_g = k_g G = k_{g+1} G = k_{g+2} G = ...$
   $s_g = (1 + xH(R_g)k_g^{-1}) mod\, q$
   $s_{ug} = s_g \oplus Password$
ii. Pre-processing (both at signer and receiver):
   $k_c = H(loc_c, t)$
   $(x_c, y_c) = k_c R_g \, mod\, q$
iii. Signature Generation (at signer): A signer g computes the following equations.
   $s_g = s_{ug} \oplus Password$
   $s_{c,g} = k_c^{-1}(H(m) + s_g x_c) mod\, q$
   Signature $(s_{c,g}, R_g)$, and m are transmitted.
iv. Verification (at recipient):
   IF $(x_c, y_c) = (H(m)R_g + x_c(R_g + H(R_g)Q))s_{c,g}^{-1} mod\, q$
   Accept m, ELSE Reject.

In order to provide anonymity during message delivery, TA chooses for each individual OBU a random number k_g (again, $1 < k_g < q$). A common identifier R_g is calculated as:

$$R_g = k_g G. \tag{11}$$

For example, if $g, g+1, g+2, ...$ are the registered vehicles; TA computes $R_g = k_g G = k_{g+1} G = k_{g+2} G = ...$. Hash function $H(.)$ is used for computing $H(R_g)$. TA derives a partial delegation key [12] for each vehicle g from the the master secret x using the equation below.

$$s_g = (1 + xH(R_g)k_g^{-1}) mod\, q. \tag{12}$$

The delegation key s_g is XOR-ed with the corresponding user's password to provide protection against a potential OBU compromise attack.

$$s_{ug} = s_g \oplus Password \tag{13}$$

s_{ug} is then delivered to the corresponding vehicle's OBU in a secure manner. When the OBU want's to deliver a signed message, the user must input the password which would be XOR-ed again with the s_{ug} value to reproduce the actual delegation key s_g.

2) Signature Preprocessing: When a vehicle wants to deliver a message— either triggered by a VANET safety application on a road-traffic accident, or from any other application source, it associates the signature with a timestamp, and the vehicle's current position as session parameters. Similar approximated session parameters are obtained and used by a verifying entity (OBU and/or RSU) using the verifier's location information and the current timestamp. This is to defeat the replay attack of expired safety messages in the VANET. Also, an adversary is unable to broadcast the old message in a different geographical location. The steps are given as:

The signing vehicle uses its GPS information as the location information loc_c, and a timestamp t that refers to the expiry information of the corresponding message m.

$$k_c = H(loc_c || t) \tag{14}$$

Signing OBU computes the session parameter x_c as

$$(x_c, y_c) = k_c R_g (mod\, q). \tag{15}$$

3) Signature Generation: Once the sending vehicle is ready to deliver the message m, it reproduces the delegation secret, $s_g = s_{ug} \oplus Password$; and signs the message using the following ECDSA formula:

$$s_{c,g} = k_c^{-1}(H(m) + s_g x_c)(mod\, q). \tag{16}$$

The signature payload and the message are combined as $(R_g || s_{c,g} || m)$ to be delivered to the neighboring vehicles (OBUs) and RSUs within the communication range of the sending vehicle's OBU.

4) Verification: For a receiver OBU or RSU, it is important to verify the integrity of the received message, as well as the source authentication. The received signature components are utilized for the verification of the signature as illustrated in the following manner.

A receiving entity computes k_c from its own location information and the current timestamp using the relationship given in eqn. (14). If the verifying node and the sender of the message are in the communication range of each other, we assume that their operational locations would be the same, and also, if the message is received within the same time-frame as the received payload's, used timestamp value t for verification is same as the received timestamp. Eqn. (15) is used to obtain (x_c, y_c) values by the verifier.

Finally, the verification equation is checked as

$$(x_c, y_c) = (H(m)R_a + x_c(R_g + H(R_g)Q))s_{p,i}^{-1}(mod\, q). \tag{17}$$

These two VANET authentication mechanisms are summarized in Algorithm 1 and 2 respectively.

C. Parking Assistance With Security and User Anonymity

In our system, periodic status messages from PSC are to disseminate the public information about parking availability, environmental events etc.

On the other hand, parking request (PRQ), parking token (PTN), and parking rejection (PRJ) messages are solely associated to a specific user of the VANET. This implies that data secrecy and message authentication both are mandatory in user specific messages, while only message authentication is required in periodic broadcasts.

Secured WAVE Short Message Protocol (WSMP) [14] format is used to construct a parking status message which is generated by TA, signed and periodically transmitted over the VANET by PSC using the mechanism stated in Algorithm 1. Along with the parking availability information, a periodic status message also contains PSC's ephemeral public key v in

TABLE II
SIGNATURE OVERHEADS OF PROPOSED SCHEME

	Used Elliptic Curve		
Approach	160-bit EC	P-224	P-256
Ordinary ECDSA	166 Bytes	182 Bytes	190 Bytes
Algorithm 1	44 Bytes	60 Bytes	68 Bytes
Algorithm 2	40 Bytes	56 Bytes	64 Bytes

the WSM's application data field. When the current message expires, the verification will fail for m at the receiver end. TA will then generate new status message and deliver it to the corresponding PSC after producing the other necessary credentials as given in step iii of Algorithm 1.

Note that we do not need to include any third party certificates with the status message as we are deploying identity-based signature in Algorithm 1. However, we must add the message expiry information (t_m), and the geographical tolerance (a_m) with the WSM. This signature is unforgeable, verifiable, distinguishable, and undeniable.

Disclosure of contents in a vehicular communication is the major security concern for VANET users. Normally, a user would not like to let anybody know about the application messages its sending and receiving. Therefore, payloads of user specific messages are signed using Algorithm 2. The signed payloads are encrypted using random symmetric keys, while the symmetric keys are again encrypted with individual recipient's public keys.

A PRQ from a requesting vehicle is designed to contain an encrypted signed message. While the signature is produced using Algorithm 2, the encryption is generated using a random symmetric key δ. The key δ is encrypted by PSC's temporary public key v, and included in the PRQ. In addition, an OBU includes a temporary public key β with the PRQ message payload. The private key α of the key-pair is kept within the OBU.

A parking token (PTN), or a parking rejection (PRJ) message is produced by the corresponding PSC to notify the requesting OBU about the approval or rejection of the request respectively. These messages are encrypted by PSC with a random session key γ. Again, the session key γ is encrypted by recipient OBU's temporary public key β. These notifications are valid for a pre-specified time period, and expires afterward. Specially, parking tokens expire and become unusable after the specified time.

D. Signature Overhead

For a 160-bit elliptic curve, the size of an ECDSA signature is 40 bytes. In case of our identity-based periodic message signature (from Algorithm 1), we need to include the expiry time t_m, as well as the geographical tolerance a_m. Assuming 2 bytes of space requirement for each of them, the total size of the signature overhead for periodic status message yields 44 bytes. The other signatures (from Algorithm 2) do not include t_m, and a_m; and hence will have only 40 bytes of signature overhead. Using two different types of NIST [15] curves—P-224 and P-256 suggested by the current security standards for WAVE [6], would have signature overhead of 56 bytes, and 64 bytes respectively with an additional overhead for the third party certificate (126 bytes). Table II summarizes the signature overheads of our proposed secure parking assistance scheme. P-256 is usually chosen for signing third-party certificates, whereas P-224 is commonly used for safety and other application messages.

V. SECURITY ANALYSIS

Security of our scheme depends on the difficulty of elliptic curve discrete logarithm problem (ECDLP). Following malicious behaviors and challenges are among the most anticipated ones in our secure parking assistance scheme.

1) Signature Forging: In order to generate a valid proxy key $s_{i,m}$, a proxy signer PSC_i would require the system secret x, hash value $h_{i,m}$ over the corresponding identities (loc_i and ID_o), and two other random numbers k_i, k_o as indicated in Algorithm 1. The secret x is irreversible from the knowledge of the public key Q, since that involves point multiplications of an elliptic curve $E(\mathbb{F}_p)$. The corresponding identity values indicate the location information of the entities, which are fixed and must not be changed by the proxy signers as they will be used by the verifier OBUs during the verification of the proxy signature. Different hash values on them would result in an unsuccessful verification.

Also, generation of a signature by an OBU_g involves the delegated secret key s_g. As given in Algorithm 2, delegation secret s_i is computed by TA using the system secret x, and individual secret k_g. Thus, forging a signature would require an attacker to have at least two secrets (x, and k_g) which are stored only in TA. Also, associated difficulty in solving an elliptic curve discrete logarithm problem would not allow an OBU to retrieve the system secret x from the knowledge of public key Q. Hence, forging a signature in our scheme would be extremely hard.

2) Replaying Old/Expired Messages: Any change or modification on message content m, or expiry information t_m would result in a different proxy key $s_{i,m}$ for which the generated signature in 1 would be different. This makes sure that a false message or a replay attack with this approach will not be successful.

A signing OBU computes the session parameter x_c from k_c using Algorithm 2. This k_c is derived by hashing the message-originator's position and timestamp which would be same as the receiving node's current position and time assuming that both the signer and the verifier are in close proximity. Therefore, repeating an old and expired signature would not pass the verification process at the receiver OBU/RSU.

3) Message Tunneling: In addition to the current system time, session parameters use a signer's current position information during the signature preprocessing. This prevents an adversary to tunnel the signed message for using it in a different location.

4) Non-repudiation: An adversary is unable to forge a signature in our scheme. Also, an old message from an OBU would not pass the verification process, and since the location

information is embedded with each signature; the message would fail the verification at a different location then the OBU's current position. As a result, once a message is signed and delivered, the sender OBU can not deny the signature for the sent message.

5) OBU Compromise: An attacker may compromise an OBU to obtain its secret s_{ug}. However, the signature generation of OBU requires the corresponding user's password without which the delegated secret s_g can not be obtained (refer to Algorithm 2). Hence, an OBU compromise in our scheme would not let an adversary find either the delegation secret s_g, or the system secret x.

6) Signature Linking: An OBU may sign identical payloads in subsequent time-frames. A timestamp-based session parameter x_c during the signature preprocessing phase ensures the change of signatures in different time frames even if the message contents resemble to the previously sent messages.

A. Identity Dispute and Revocation

On an identity dispute, TA may generate all possible signatures using each individual secret s_g with the same R_g value. If the alleged signature is a valid one, and the time, location information are accurate; it'd match with one of the generated signatures. Secret credentials of the matched signature will then be used to identify the disputed user's identity.

When TA revokes an entity (say, OBU_g), it appends the corresponding secret s_g to the revocation list and sends an update to *PSC*. Using the disclosed s_g, PSC can internally derive a new signature on each received message originated from the revoked OBU. If a received signature matches with the derived one, PSC identifies the sender as a revoked entity.

VI. Network Simulation

We develop a simulation program to investigate the network performance of our scheme using network simulator ns-2.34 with IEEE 802.11p parameters for MAC and PHY provided by IEEE 802.11Ext package from Chen et al. [16].

A small roadside parking facility of 200 m length and 100 m width has been considered where vehicles unicast periodic parking updates/requests (PRQs) to the PSC, as well as broadcast the regular periodic safety messages. PSC at the parking lot responds to each vehicle's update request with individual response (PTN/PRJ). While an OBU disseminates periodic safety messages every 100 ms (10 messages per sec.), the interval for parking message updates has been set to 300ms.

The simulator is configured for two major types of data traffic: i. periodic broadcasts by OBUs and the PSC for safety messages and parking updates respectively, and ii. parking request/responds messages from OBUs and RSU respectively. Application messages- PRQ and PTN/PRJ are associated to a higher priority access class (AC2, or AC3), whereas periodic broadcasts of OBUs and PSC are of less importance, and associated to one of the lower priority traffic classes (either AC0, or AC1) over DSRC control channel (CCH) at IEEE 802.11p MAC. Other MAC and PHY parameters used in our simulation are listed in Table III.

IEEE 802.11p MAC allows four distinguished priority classes for best effort, background, video, and voice data traffic. The latter two categories are given higher priorities over the first two traffic classes as the related EDCA parameters for IEEE 802.11p control channel (CCH) are listed in Table IV.

We choose the signed WSM protocol format (see C.6 of [6]) for the used payload size and signature overhead of broadcast communications, while an encrypted message format (see C.7 of [6]) has been adapted for a unicast operation. We compared the security overheads and payloads for different communications from the WAVE security services and our proposed authentication scheme. The details of the used message payloads for different broadcast and unicast communications are given in Table V.

Figure 3(a)-3(d) describe the network performance in terms of the normalized throughput for different combination of access categories used in our simulation.

TABLE III
Simulation Parameters for MAC and PHY

Parameters	Values
Data Rate	6Mbps
Slot Time	16μs
SIFS	32μs
Short Retry Limit	7
Long Retry Limit	4
Bandwidth	10MHz
Frequency	5.89GHz
Propagation Model	TwoRayGround

TABLE IV
EDCA Parameters for IEEE 802.11p CCH

Priorities	Type	CWMin	CWMax	AIFSN
AC1	Background	aCWMin=15	aCWMax=511	9
AC0	Best effort	7	15	6
AC2	Video	3	7	3
AC3	Voice	3	7	2

TABLE V
Used payloads for different communication types.

Communication Type	WAVE Services	Our scheme
PSC Status Updates	254 Bytes	132 Bytes
OBU Safety Messages	254 Bytes	128 Bytes
Parking Request (PRQ)	394 Bytes	260 Bytes
PSC Response (PTN/PRJ)	394 Bytes	260 Bytes

As shown in Figure 3, parking status messages have greater success ratio than that of VANET's periodic safety message broadcasts. The lower offered load, as well as the association with higher priority traffic classes (AC2 or AC3) for the authenticated parking status updates ensure the higher performance in periodic safety message delivery.

Figure 3(a) illustrates the scenario where periodic safety messages and parking updates are associated with AC0 and AC2 respectively. Parking assistance messages using WAVE

security have been delivered with over 97% success rate for 60 vehicles. Since ID-based signature scheme allows secure parking assistance with smaller messages (refer to the Table V), between PSC and OBU, similar success rate is obtained for up to 80 OBUs in the parking facility using our approach.

Comparison of the periodic safety message broadcast over access class AC1, and parking status message updates over AC2 is shown in Figure 3(b). The significant difference of the corresponding contention window (CW) sizes, as well as the Arbitration Inter-Frame Space Number (AIFSN [1], refer to the Table IV) values between AC1 and AC2 access classes enable the parking updates with higher ratio of parking message delivery. While the WAVE security protocol allows over 99% successful delivery of parking status messages for up to 80 vehicles, our scheme allows the similar success rate for as many as 100 vehicles in the VANET.

Periodic broadcasts are sent over AC0 whereas parking status updates are delivered over AC3 as indicated in Figure 3(c). WAVE-secured parking messages are delivered at about 95% success rate while our scheme enables around 97% successful parking status message delivery for up to 90 vehicles in the vicinity of the controller.

Again, due to the significant relative difference of CW and AIFSN values between AC1 and AC3 (refer to the Table IV), parking status messages assigned with AC3 perform with greater success rate compared to the corresponding low priority (AC1) periodic safety messages' delivery ratio. As shown in Figure 3(d), for about 85 vehicles in the parking lot, the success ratio of WAVE-secured parking assistance messages is 95% while our scheme provides successful message delivery of over 99% parking update messages for as many as 100 vehicles.

Clearly, the parking status updates have to be assigned with higher priority of traffic classes than that of the VANET's periodic safety messages in order to achieve high degree of successful message delivery in a parking assistance application. In our experiments, we have shown that our scheme can provide over 99% successful message delivery for 100 vehicles in a parking site. However, the successful parking message delivery ratio can be maximized for even higher number of vehicles by choosing smaller window size, and/or by selecting smaller AIFSN value for the associated higher traffic class assigned to the parking status messages.

VII. Conclusion And Future Direction

A WAVE-enabled system for secure and privacy-preserving car parking assistance has been presented in this paper. This work relies on two modified ECDSA authentication mechanisms for infrastructure-to-vehicles and vehicles-to-infrastructure message authentications during the communications between OBU and infrastructure for parking assistance application. The development of signature schemes for providing message authentication, integrity and anonymity is in harmony with the current WAVE security services which makes our approach compatible with the existing VANET standards. We investigated our scheme in the light of all known malicious attacks and scenarios, while it is proved that a successful attack is reasonably hard to launch on our proposed system. Simulation results compare the network performance of the scheme with different combinations of traffic classes for the infrastructure-to-vehicles and vehicles-to-infrastructure communications using WAVE security protocol and our scheme.

Our approach can be further extended for new VANET applications like automated toll collection system, highway access control mechanism etc. The existing network- and security standards for VANETs have essentially addressed the highly dynamic nature of the communicating nodes on roads and highways. Because of the added features with unique requirements, scenario-specific applications like the proposed one may require a common platform with some degree of amendments to the original security and network primitives in the VANET standards.

References

[1] "Draft amendment for wireless access in vehicular environments (WAVE)," IEEE, New York, NY, IEEE Draft 802.11p, Jul. 2007.

[2] R. Lu, X. Lin, H. Zhu, and X. Shen, "An intelligent secure and privacy-preserving parking scheme through vehicular communications," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 6, pp. 2772 –2785, july 2010.

[3] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Advances in Cryptology CRYPTO 2001*, ser. Lecture Notes in Computer Science, J. Kilian, Ed. Springer Berlin / Heidelberg, 2001, vol. 2139, pp. 213–229.

[4] S. D. Galbraith, K. G. Paterson, and N. P. Smart, "Pairings for cryptographers," *Discrete Appl. Math.*, vol. 156, pp. 3113–3121, September 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1450345.1450543

[5] D. Johnson and A. Menezes, "The elliptic curve digital signature algorithm (ECDSA)," Certicom Research, Canada; and Dept. of Combinatorics and Optimization, University of Waterloo, Canada, Tech. Rep., 1999.

[6] "IEEE trial-use standard for wireless access in vehicular environments (WAVE)- security services for applications and management messages," IEEE, New York, NY, IEEE Std 1609.2, Jul. 2006.

[7] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing vehicular communications," *Wireless Communications, IEEE*, vol. 13, no. 5, pp. 8–15, October 2006.

[8] Y. Sun, R. Lu, X. Lin, X. S. Shen, and J. Su, "An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 7, pp. 3589 –3603, 2010.

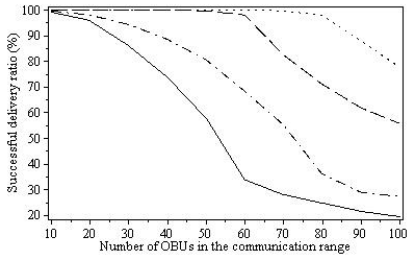
[9] B. Libert and D. Vergnaud, "Multi-use unidirectional proxy re-signatures," in *Proceedings of the 15th ACM conference on Computer and communications security*, ser. CCS '08. New York, NY, USA: ACM, 2008, pp. 511–520. [Online]. Available: http://doi.acm.org/10.1145/1455770.1455835

[10] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen, "ECPP: efficient conditional privacy preservation protocol for secure vehicular communications," in *Proceedings of 27th IEEE International Conference on Computer Communications INFOCOM 2008. Joint Conference of the IEEE Computer and Communications Societies*. Phoenix, AZ, USA: IEEE, 2008, pp. 1229–1237.

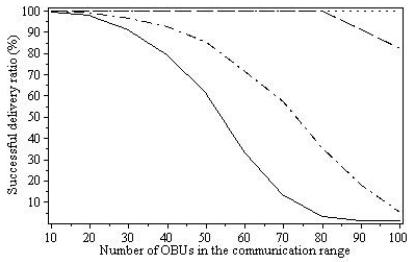
[11] Q. Wu, J. Domingo-Ferrer, and U. Gonzalez-Nicolas, "Balanced trustworthiness, safety, and privacy in vehicle-to-vehicle communications," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 2, pp. 559 –573, 2010.

[12] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in *CCS '96: Proceedings of the 3rd ACM conference on Computer and communications security*. New York, NY, USA: ACM, 1996, pp. 48–57.

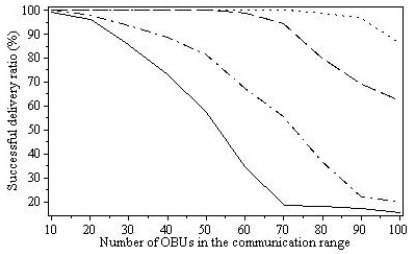
[13] S. Kim, S. Park, and D. Won, "Proxy signatures, revisited," in *ICICS '97: Proceedings of the First International Conference on Information and Communication Security*. London, UK: Springer-Verlag, 1997, pp. 223–232.
[14] "IEEE trial-use standard for wireless access in vehicular environments (WAVE)- networking services," IEEE, New York, NY, IEEE Std 1609.3, Apr. 2007.
[15] NIST, "NIST: national institute of standards and technology," http://www.nist.gov/index.html, 2011.
[16] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, "Overhaul of IEEE 802.11 modeling and simulation in ns-2," in *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, ser. MSWiM '07. Chania, Crete Island, Greece: ACM, 2007, pp. 159–168. [Online]. Available: http://doi.acm.org/10.1145/1298126.1298155



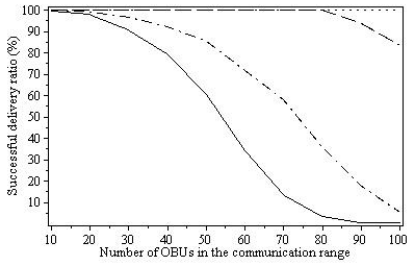
(a) Periodic broadcasts are sent over AC0, while parking status updates are sent over AC2



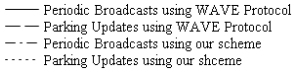
(b) Periodic broadcasts are sent over AC1, while parking status updates are sent over AC2



(c) Periodic broadcasts are sent over AC0, while parking status updates are sent over AC3



(d) Periodic broadcasts are sent over AC1, while parking status updates are sent over AC3



(e) Legend

Fig. 3. Percentage of successful message delivery in a prioritized WAVE-enabled parking facility.