One-way-linkable Blind Signature Security Architecture for VANET

Baber Aslam and Cliff C. Zou
Dept. of Electrical Engineering and Computer Science
University of Central Florida, Orlando, FL, USA

***Abstract*— Security attributes of a Vehicular ad hoc network (VANET) include confidentiality, integrity, authentication, non-repudiation (liability), revocation and privacy. Privacy, having characteristics opposing to the rest of the attributes, makes design of a security architecture quite difficult. A commonly used solution is to have a large number of temporary certificates (i.e., pseudonyms) to achieve these security attributes. To guard against their malicious use, these pseudonyms are stored in expensive tamper-proof-devices (TPDs). Further, a large number of valid pseudonyms, at any given time, make non-repudiation and revocation quite complex and difficult to achieve. Another solution is to get pseudonyms blindly signed from a certificate server, thus eliminating the need of TPDs (given the pseudonyms are not generated in bulk). However, blind signatures provide unconditional privacy and thus require complex/multi-transaction procedures to ensure non-repudiation/revocation.**

We present a security architecture by revising the original Blind signature scheme. Our proposed architecture provides "one-way-link-ability" that helps to achieve all the security attributes without introducing complex/multi-transaction procedures. It does not require expensive TPDs or complex pseudonym issuance/revocation procedures and is especially suited to VANET during initial deployment phase which is characterized with intermittent connectivity. Further, non-repudiation/revocation requires cooperation between multiple entities thus ensuring privacy without a single point of failure.

Keywords- security; privacy; VANET; Blind certificate

I. Introduction

Vehicular ad hoc network (VANET) is characterized by dynamic topology and membership. Vehicles may cross city/county/state/country boundaries thus leaving one network and joining another. Further, the initial deployment stages of VANET will have sparsely/widely spaced roadside units (RSUs) thus resulting in intermittent vehicle to infrastructure communication with long blackouts.

The desired security attributes for VANET include authentication, confidentiality, integrity, non-repudiation, revocation and privacy. It is important to note that privacy is the most important attribute, but at the same time it is in conflict with other attributes thus complicating the design of VANET security architecture.

The simplest security architecture is to assign a single permanent certificate to each vehicle, this ensures authentication, confidentiality, integrity, non-repudiation, revocation but not the privacy. To address privacy, basic architecture can be extended to use multiple temporary certificates (normally referred as pseudonyms) instead of one permanent certificate; this ensures privacy since pseudonyms cannot be linked with each other and to the user [1-6]. Different schemes for pseudonym-management have been proposed to ensure unlink-ability. One such scheme is to issue pseudonyms in bulk to vehicles [1]; the vehicle can then use these to ensure privacy. The bulk pseudonyms based scheme requires a tamper-proof-device (TPD) to store the pseudonyms and perform cryptographic operations [1], since these pseudonyms may be used for malicious purposes such as Sybil attacks. The TPDs are expensive and need reloading with new pseudonyms when old ones expire or are used up.

Possible solutions can be to let vehicles generate pseudonyms themselves [2, 3] or periodically get new pseudonyms from some certificate servers [4, 5]; thus eliminating the need of TPD (given the pseudonyms/other-authenticating-credentials with overlapping validity are not generated in bulk). First option makes revocation very complex and difficult while second option makes privacy difficult to achieve (since certificate server can link various pseudonyms). Blind signature scheme [7], with some kind of link-ability, is usually employed to address privacy issues of second option [4, 5]. The process requires multiple-certificate-servers/multiple-transactions for one signature (i.e., for getting one pseudonym) and is thus difficult to realize, especially with an intermittent communication link with the infrastructure. Blind signature scheme is also used in [6], but the solution requires generation of authenticating-tokens in bulk thus needing TPD.

Other architectures include those based on principles of group signatures and ID cryptography [8]. In case of group signatures, vehicles form part of a group with a trusted group manager. The architecture requires members to trust the group manager (who can find the true identity of signer), which will be difficult to achieve in a dynamic VANET. Further, size, membership revocation and dynamic membership (new nodes entering a group and old nodes leaving the group) increase the complexity and overheads of this method.

In this paper we present a security architecture based on revised Blind scheme. The architecture satisfies required security attributes by using carefully-designed pseudonyms. The pseudonyms are refreshed by vehicles via Roadside Units (RSUs) using revised Blind signature scheme. To refresh pseudonyms, a vehicle uses its previous valid certificate to authenticate its blinded pseudonym-signature-request message to a passing-by RSU. The RSU generates/stores a tag/link based on its received blinded pseudonym-signature-request message and the certificate that was used to authenticate the message. The tag/link helps to ensure non-repudiation and certificate revocation. We do not require multiple sessions or multiple RSUs to generate this tag/link. We have modified the original Blind signature scheme by enforcing a condition on the blinding factor; this also helps to guard against other attacks towards the original Blind signature scheme (discussed later).

978-1-4244-8788-2/11/$26.00 ©2011 IEEE

We do not generate pseudonyms, with overlapping validity, in bulk which must be guarded against malicious use by user/attacker (e.g., by storing these in a TPD). The non-overlapping pseudonyms or other long term certificates (that may exist at any time) can be securely stored without need of TPD by employing methods that are currently being used in securing certificates in personal computers/servers. The architecture satisfies all security attributes without requiring expensive TPDs or complex multi-step transactions with multiple certificate servers. The architecture does not require users to trust a party with their private/secret keys and thus will have more user acceptance. Further, non-repudiation/revocation requires cooperation between multiple entities thus ensuring privacy without a single point of failure.

The paper is organized in seven sections. Section II discusses system model, Section III introduces Blind signature scheme, Section IV presents proposed architecture, Section V explains system setup, Section VI describes realization of security attributes, Section VII discusses related research, and Section VIII gives conclusions/future work.

II. System Model

A. *Security Objectives*

VANET's security requirements are more complex than other wired/wireless networks. In addition to basic security attributes of authentication, confidentiality and integrity, it also requires non-repudiation, revocation and privacy. These additional security attributes are briefly discussed below:-

1) Non-repudiation: A user should not be able to later deny that she originated a message. It adds liability to user for the messages which she generates. This is especially important in case of VANET safety applications. If this requirement is not fulfilled then a malicious node may generate fake public safety message without any liability.

2) Revocation: Revocation of user's credentials is also an important security attribute. It helps to minimize the damages if a user's credentials are lost or a user engages in malicious activity.

3) Privacy: Privacy is one of the most important security attributes in VANET applications. This is due to the fact that VANET communication can be used to track a vehicle (driver) which causes great concerns to many users. Privacy comes in direct conflict with the other security attributes. One has to strike a balance between privacy protection and the other security attributes, especially non-repudiation.

B. *Threat Model*

We do not make very stringent security requirements for vehicle's on-board device or restrict the capabilities of attacker node. We assume that an attacker is capable of:

- eavesdropping when within the routing path or in the transmission range of a message
- injecting, modifying, spoofing or dropping the messages
- trying to track the movement of another vehicle either alone or in collaboration with other mobile or fixed nodes (total number of such collaborating nodes will be a small fraction of all the nodes participating in the network since we assume that majority of nodes are honest)
- taking complete control of her on-board device and also crafting any protocol related messages

C. *Desired Requirements*

Keeping in mind VANET characteristics, attacker capabilities and security attributes, our desired requirements for the proposed security architecture are:-

- Ensure authentication, confidentiality, integrity, non-repudiation, revocation and privacy.
- Guard against traceability by one or more collaborating entities. An attacker alone or with collaboration of limited other mobile or fixed nodes should not able to track a user. In other words, two messages from the same user should not be linkable (if desired).
- Ensure privacy revocation involves multiple authorities. A single authority, by itself alone, should not be able to revoke the privacy of a user. Privacy revocation could only be achieved by cooperation of multiple identities.
- Provide security without need of expensive TPDs, or large storage requirements at central authority/ RSU.
- Guard against a user using legitimate pseudonyms for malicious purposes such as Sybil attack, etc.
- Do not require multiple transactions for various routine operations, such as certificate issuance, certificate revocation, etc. This is especially necessary due to the intermittent nature of connectivity of VANET.

III. Basic Blind Signature - Introduction

Blind signature scheme was first introduced by Chaum [7]. It makes use of multiplicative property of RSA (discussed below). Blind signature scheme based on elliptic curve cryptography can be used interchangeably; we, in this paper, will restrict ourselves to RSA based scheme only.

Entity ***A*** wants to get message *m* blindly signed by entity ***B***; *m* could be hash of some message *M*. Note that the entity ***A*** may need to prove to entity ***B*** that it is entitled to receive blind signatures. The authentication could be done using some token or signatures on message *m*. The details of authentication are omitted, since it is not essential to the basic concept of Blind signatures. The Blind signature scheme is shown in Fig. 1.

Given *m*, *s* and public parameters, the signatures are valid if $y = m;$ where $y = (s)^{e} = (m^{d})^{e} = m \bmod n.$ The Blind signature scheme can be used to certify pseudonyms; but it raises many security issues, such as:

- There is no way to ensure non-repudiation and certificate revocation, since newly signed pseudonyms cannot be linked to authenticator/node (i.e., given <*m, s*> it is not possible to construct *m'* or a link to authenticator of *m'*).
- The signed pseudonyms may be used to launch Sybil attacks, and we cannot deal with it since there is no way to link pseudonyms with each other or with the true identity of the node.

- A node with valid authenticator may share its pseudonyms with another node that does not have a valid authenticator and who is unable to get pseudonyms.

A:	1.	Generate random number *r:* gcd $(r, n)=1$
	2.	Compute blinding factor b_f: $b_f = r^e$
	3.	Blind message m to m' : $m' = b_f m = (r^e m) \bmod n$
$A \rightarrow B$:	4.	m'
B:	5.	Sign message m' using private key d: $x = (m')^d \bmod n$
$B \rightarrow A$:	6.	x
A:	7.	Recover message signature: $s = m^d = r^{-1}(x) \bmod n$
	8.	$r^{-1}(x) \bmod n = r^{-1}(m')^d \bmod n = r^{-1}(r^e m)^d \bmod n$ $= r^{-1} r\, m^d \bmod n = m^d \bmod n$

Figure 1. Basic Blind signature scheme (public key parameters: *n, e* = public key of *B* and *d*= secret key of *B*).

IV. Proposed Architecture

The architecture uses a certificate chain consisting of long-term and short-term certificates. Initially, a long-term certificate is used to get the initial short-term certificate and later a new short-term certificate can be obtained based on the previous short-term certificate, thus making a certificate chain (details discussed later in this section). We have revised the Blind signature scheme to meet our requirements of non-repudiation and revocation.

A. Notations and Function Definitions

We define several notations and functions which we will use in formal description of our architecture. A certificate or pseudonym $Cert_x$ is essentially defined by its associated identification ID_x and key pair (P_x , S_x); public key P_x forms part of certificate and secret key S_x is known only to the holder of $Cert_x$. $Sig\text{-}Cert_x (M)=N$, is a signature function on message M using key S_x or certificate $Cert_x$. The signature N is computed by first creating a message digest ($M_h = Hash(M)$) using some well known hashing function (such as SHA1) and then encrypting the digest using key S_x. $VerSig\text{-}Cert_x (M, N)$, is a signature verification function with two inputs: the message M and the signature N. It verifies the signature by computing the message digest of message M and comparing it with received signature N (after decrypting it with the corresponding public key P_x). Note that knowledge of $Cert_x$ is needed for function $VerSig\text{-}Cert_x (M, N)$. $Cert_x$ should either be publicly available or attached along with the $Sig\text{-}Cert_x (M)$. In the rest of the paper it is assumed that $Cert_x$ is either publicly available or attached along with the $Sig\text{-}Cert_x (M)$, and will not be explicitly mentioned.

B. Proposed Revised Blind Signature Scheme

In order to address the security issues of the original Blind signature scheme and to satisfy our security objectives, we revised Blind signature scheme (Fig. 2). Our proposed scheme achieves *one-way-link-ability,* i.e., given a blinded pseudonym (m') the signer cannot find the un-blinded pseudonym (m), but given a certificate (<un-blinded pseudonym -*m,* signatures -*s*>) the signer can construct the associated blinded pseudonym (m') and find a link to its authenticator (authenticator of m'). *One-way-link-ability* ensures privacy since the signer, at the time of signing signatures, cannot determine the pseudo-credentials. Whereas for revocation/non-repudiation (given the pseudonym), it is possible to construct the chain/link leading to the node's true identity.

Suppose that vehicle V has a current certificate $Cert_{i-1}$ which is valid for time period T_{i-1} (time period defines a start and an end time) and now needs to get a new certificate $Cert_i$ valid for time period T_i from a nearby RSU R (Fig. 2). V generates $Cert_i$ (step 1), blinds the certificate using public credentials of RSU R (steps 2, 3), authenticates the blind-signature-request-message with $Cert_{i-1}$ and sends the request to the RSU (step 4). RSU R verifies validity of $Cert_{i-1}$ (step 5), verifies signatures on request (step 6), generates/stores the tag/link (steps 7, 8) and sends signed message to V (step 9). V un-blinds the message to get the signatures on pseudonyms (step 10) and then uses the pseudonym as required (step 11), but makes sure to not use $Cert_i$ with R.

V:	1.	Generate $Cert_i = < ID_i , P_i >$ and S_i ; $ID_i = b_f = r^e$; where ID_i is pseudo ID, P_i and S_i are public and secret keys of V.
	2.	Compute $m_i = Hash\ (Cert_i)$
	3.	Compute $m_i' = b_f m_i = ID_i m_i = (r^e m_i) \bmod n$
$V \rightarrow R$:	*4.*	$m_i', T_i , Sig\text{-}Cert_{i-1} (m_i'), Cert_{i-1}$
R:	5.	Verify $Cert_{i-1}$ for time-period validity and revocation. (details in certificate revocation)
	6.	$VerSig\text{-}Cert_{i-1} (m_i', Sig\text{-}Cert_{i-1} (m_i'))$
	7.	Compute $x =(m_i')^d\ mod\ n$
	8.	Store link $<m_i', T_i , Cert_{i-1}>$ *or* alternatively $< m_i', T_i , m_{i-1}' >$
$R \rightarrow V$:	*9.*	$x, T_i , Sig\text{-}Cert_R (T_i)$
V:	10.	Recover certificate signature $Sig\text{-}Cert_R (Cert_i)$: $Sig\text{-}Cert_R (Cert_i) = s = (m_i)^d = r^{-1}(x)\ mod\ n$
	11.	Use $<Cert_i , T_i , Sig\text{-}Cert_R(Cert_i), Sig\text{-}Cert_R(t_n), Cert_R>$ as new credentials

Figure 2. Proposed revised Blind signature scheme – initial version (public key parameters: *n, e* = public key of *R, d* = secret key of *R*).

The solution ensures *one-way-link-ability* to achieve non-repudiation/revocation: $Cert_i$ cannot be generated from m_i', but m_i' can be generated from $Cert_i$ and m_i' can be linked to $Cert_{i-1}$. Note that revocation/non-repudiation cannot be accomplished by a single signing RSU (or a few RSUs); it requires cooperation between all involved RSUs to reconstruct the chain/link iteratively. The utility of *one-way-link-ability* rests on the assumption that a node should not declare (use) the un-blinded pseudonym to (with) the RSU who issued signatures on its blinded version.

The solution shares a limitation with Blind signature scheme, i.e., the signer cannot ensure that the blinded message (certificate) is well-formed (constructed as per agreed protocol/scheme). Specifically the RSU cannot ensure $b_f =ID_i$. One commonly used solution is to use cut-and-choose method [4, 6]. Here the user sends multiple certificates to the signer (e.g., user sends two blinded messages, if she wants to get signatures on one); the signer can then choose which half to sign and the user un-blinds the other half for the signer to check if these were well-formed or not. This reduces the success probability of attacks by malicious users but at the same time adds considerable overhead, which is not affordable in the face of intermittent connectivity in VANET environment. In order to address this vulnerability, we have further refined the Blind signature scheme. The modifications are given in Fig. 3 (only shows the several revised steps in the initial proposed approach given in Fig. 2).

$Sig\text{-}Cert_R\ (m_i'_H \,||\, T_i)$ is the modified Blind signature, which serves three purposes: attaching a certificate-valid-time-period condition to the signature, adding link-ability to the certificate for later non-repudiation/revocation purpose, and guarding against malicious use of the signature (malforming the blind message, changing ID to make certificate un-linkable, sharing the signed certificate, etc).[1]

R:	7.	Compute $m_i'_H = Hash\ (m_i')$
	8.	Store link $<m_i'_H, T_i, Cert_{i-1}>$ *or* alternatively $< m_i'_H, T_i, m_{i-1}'_H >$
$R \rightarrow V$:	9.	$m_i'_H, T_i, Sig\text{-}Cert_R\ (m_i'_H \,||\, T_i)$; $x||y$ is concatenation of x and y.
V:	10.	Use $<Cert_i, T_i, Sig\text{-}Cert_R(m_i'_H \,||\, T_i),\ Cert_R>$ as new credentials

Figure 3. Proposed revised Blind signature scheme – final version.

V. System setup

Three types of certificates have been defined: permanent certificates, long-term/daily certificates, and short-term/temporary certificates (i.e., pseudonyms) (Fig. 4). Each vehicle will have a permanent certificate that is registered with a Central Certification authority (CCA) similar to vehicle registration authority. The CCA can be state or country based and its operational area is divided into regions, with each region having a Regional Certification Authority (RCA). A vehicle on entering a region registers itself with the RCA; RCA in turn updates the vehicle's current region information on CCA (the update only takes place when a vehicle moves from one region to another). Either RCA or CCA can confirm that the permanent certificate of vehicle has not been previously revoked. This helps to target the revocation to concerned regions only, and hence, simplifies revocation and reduces certificate revocation list (CRL) size. The size of a region depends on the desired privacy granularity.

A vehicle gets one long-term certificate per day from RCA using proposed Blind signature scheme. RCA stores the relevant link. One long-term certificate per day reduces the chain size which makes revocation simple. A vehicle uses this long-term certificate to get its first short-term certificate (of the day) from an RSU, using the modified Blind signature scheme proposed in this paper. The RSU stores the relevant link/tag in its database and informs RCA (via a confirmation message) that a short-term certificate has been issued based on a particular long-term certificate. The RCA modifies the freshness/used bit associated with the record. If later the RCA receives another certificate-issue-confirmation-message for the same long-term certificate, it marks the vehicle as malicious and takes appropriate measures such as certificate revocation. RSUs can use $m_i'_H$ instead of un-blinded long-term certificate in confirmation message to further ensure privacy.

For each subsequent certificate, the vehicle uses its previous/last short-term certificate to authenticate its current request. The issuing/signing RSU in this case sends a confirmation message to the RSU who issued/signed the previous short-term certificate. The RSU who issued/signed the previous short-term certificate then modifies the freshness/used flag associated with the record. This ensures that more than one certificate are not issued based on one particular short-term certificate. The time period of the new certificate will be non-overlapping and later than the validation period of the previous certificate.

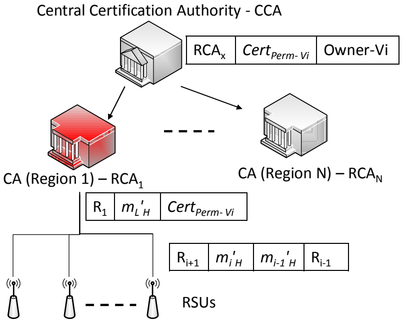

Figure 4. Certificate Architecture: CCA maintains current RCA and permanent certificate to owner link, RCA maintains permanent certificate to blinded-long-term certificate link and RSU that reported usage of long-term certificate, each RSU maintains authenticating-certificate (and its issuer) to blinded-short-term certificate link and the RSU that reported usage of issued short-term certificate.

The issuer/signer (RCA/RSU) of authenticating certificate ($Cert_{i-1}$) while modifying the freshness/used bit may also record the RSU which has sent the confirmation message. This will simplify revocation process (discussed later in Section VI) but will also raise limited privacy issues since the RSU knows the link to the next RSU as well as the previous RSU. Note that even with this knowledge a single RSU cannot compromise the privacy of a vehicle; it still needs cooperation from other RSUs, though in this case it knows the RSUs with which it should cooperate.

We require RSUs to send certificate-issue-confirmation-message to issuer/signer (RCA/RSU) of authenticating certificate ($Cert_{i-1}$). This ensures that no more than one pseudonym with same/overlapping validity will be issued. For this goal, the RSU sends certificate-issue-confirmation-message to the issuer/signer (RCA/RSU) of authenticating certificate ($\text{Cert}_{\text{i-1}}$) and waits for a timeout before signing new pseudonym. Issuer/signer (RCA/RSU) of authenticating certificate ($\text{Cert}_{i\text{-1}}$) responds within the timeout period only if malicious activity is detected. This ensures desired security with minimum overhead.

VI. Security Attributes

Confidentiality, integrity, and authentication can be achieved by using short-term certificates for signatures and/or encryption. Rest of the security attributes are discussed below:

A. Privacy

The solution ensures privacy since the RCA/RSUs do not know the ID and public keys of a vehicle at the time of signing the blinded certificate. Further, since a vehicle gets a short-term certificate from one RSU and uses it later with another RSU, a single RSU cannot link different short-term certificates of a particular vehicle. Tracing is possible but quite difficult for attackers, which can be achieved only when all the RSUs that

[1] To guard against blind decryption or blind signatures on some other message besides certificates, certification servers will have different certificates for signing and encrypting other messages.

issued certificates to a particular vehicle cooperate with each other. Even if attackers can trace a vehicle in this way, the true ID of a vehicle cannot be determined without the help from RCA. Also, RCA by itself cannot compromise the privacy of the vehicle. This property improves users' confidence since even the government authority itself cannot compromise user privacy---government authority must get cooperation from commercial operators who operate the RSUs in order to trace a vehicle and its user.

B. *Non-repudiation (Liability)*

If a malicious message, signed by a particular certificate, has been identified then the privacy of signer can be revoked with the cooperation between RCA and RSUs. The signed message will contain the information < $Cert_i$, T_i , $Sig\text{-}Cert_{Rn}$ ($m_i{}'_H \parallel T_i$), $Cert_{Rn}$>. It is assumed that the certificate and signatures on the malicious message are valid, since if the certificate is not valid then the message will be discarded and there will be no need of revocation. The revocation is performed backwards iteratively as following (refer to Fig. 5):

1) RSU – R_n will generate $m_n{}'_H$, locate the record and find corresponding $Cert_{n-1}$.

2) It will then forward the revocation request to RSU-R_{n-1} which issued $Cert_{n-1}$.

3) The chain will be followed till first RSU and then RCA which will reveal the true ID of malicious vehicle (based on long-term certificate).

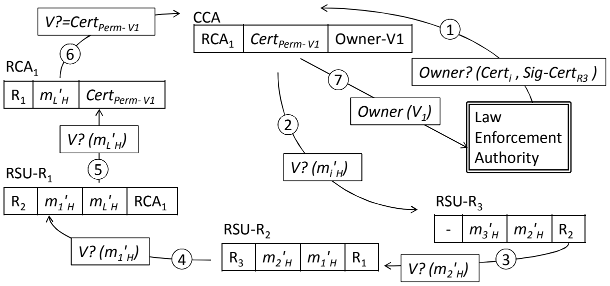

Figure 5. Non-repudiation procedure: (1) Law enforcement forwards the short-term certificate under investigation to CCA, (2) CCA forwards the blinded-short-term certificate to the concerned RSU, (3,4) RSUs iteratively forward the blinded-authenticating-short-term certificate to its issuing RSU, (5) RSU forwards the blinded-long-term certificate to RCA, (6) RCA forwards corresponding permanent certificate to CCA, (7) CCA provides the ownership information to requesting Law enforcement authority.

C. *Certificate Revocation*

There may be a situation when the certificates of a particular vehicle are to be revoked. It is important to note that the majority of vehicles will be honest and certificate revocation will not be routine so the proposed protocol has been designed to minimize overheads in normal situations. The certificate revocation decision may be made at CCA based on either request of user (for stolen credentials) or law enforcement (for malicious use). The detail of decision methodology is out of the scope of this paper and will not be discussed. The CCA will inform the RCA of the region where the vehicle last registered. Revocation is processed iteratively along the certificate chain in forward direction as following (Fig. 6):

1) RCA will check to see if the vehicle has already used its long-term certificate (to get short-term certificate from some RSU) or not. If the vehicle did not get any short-term certificate then RCA will revoke long-term certificate by broadcasting revocation command containing the hash of the blinded long-term certificate ($m_i{}'_H$). RSUs will not issue first short-term certificate based on this long-term certificate. The certificate revocation command will expire after the validity of long-term certificate.

2) If the vehicle has used its long-term certificate to get the short-term certificate then the RCA will broadcast revocation command to all RSUs containing hash of corresponding blinded short-term certificate ($m_i{}'_H$). Note if the RCA maintains the ID of RSU that issued the first short-term certificate (based on confirmation message sent by the RSU) then the revocation command is only needed to sent to the single RSU that issued the first short-term certificate.

3) The RSU that issued the first short-term certificate will find the link and broadcast the revocation command containing hash of corresponding blinded short-term certificate ($m_i{}'_H$). RSU may also acknowledge to the RCA. The broadcast may be limited to a few hops since it is likely that vehicle would have got the next short-term certificate from some RSU in geographical proximity of the first RSU. The broadcast range may be expanded if no RSU acknowledges. Similarly if RSUs maintain the ID of the next RSU in certificate chain then revocation message may be sent directly to the concerned RSU.

4) The revocation broadcast ends at the last RSU that issued the short-term certificate, the RSU then broadcasts revocation message containing hash of current blinded short-term certificate ($m_i{}'_H$). Other RSUs will not issue any new certificate based on this short-term certificate and will also not trust any message signed by this certificate. RSUs may also broadcast hash of current blinded short-term certificate ($m_i{}'_H$) to vehicles in the limited region (the limit can be defined) within the validity time period of certificate.

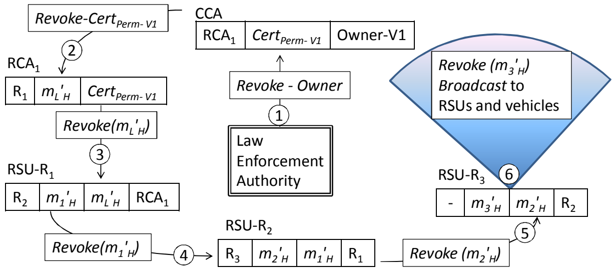

Figure 6. Revocation Procedure: (1) Law enforcement authority forwards ownership information, (2) CCA forwards the permanent certificate to concerned RCA for revocation, (3) RCA forwards the blinded-long-term certificate to concerned RSU, (4,5) RSUs iteratively forward the blinded-short-term certificate to next RSU that reported its usage as authenticating certificate (6) Last RSU broadcasts the blinded-short-term certificate to all RSUs and nearby vehicles.

VII. Related Work

Papadimitratos et al. [1] have presented a comprehensive solution based on central/regional certification authorities and pseudonyms. The solution defines multiple certificate revocation options with varying complexity and efficiency. It leaves misbehavior detection on vehicle between infrequent (once per day) CRL distributions. The solution requires expensive TPDs, further, a single authority can link the messages signed by pseudonyms.

In [2] Armknecht et al. propose a public key infrastructure where users derive public keys, certificates and pseudonyms. The certificate generated by user is verifiable by CA's public key. For revocation the CA publishes some data depending on which, all nodes have to update their keys. The excluded nodes cannot update the keys based on this data. This means for each revocation everybody has to update their certificates.

In [3] Fan et al. present detailed operation of public key infrastructure mechanism based on bilinear mapping. They achieve privacy through pseudonyms which are generated by users themselves. Revocation is accomplished through distribution of CRL that is stored by each user. Every time a user receives a beacon it performs certain computations on complete CRL to ensure that the received beacon is from an unrevoked user.

In [4] Rahman et al. present an automated crash reporting application. For privacy, they use Blind signature scheme to get anonymous credentials signed by local certification authority (government transportation authority -GTA) through a multiple transaction protocol. They achieve non-repudiation by adding an invisible identity field in pseudonym. A vehicle's unique identity (within a GTA's domain) is doubly encrypted (first by GTA's public key then by local law enforcement authority's public key) to get an invisible identity. They suggest using cut-and-choose method to ensure that blind messages are well formed, which has high overheads especially to confirm the invisible-identity. Further, the cut-and-choose method will reveal the identity of vehicle thus compromising privacy.

In [8] Lin et al. present a security mechanism using group signature and identity based signature techniques. The solution minimizes the storage at CA for later liability establishment, however the revocation is road side unit aided. CA sends RL to roadside unit which then monitors certificates in messages broadcasted by passing-by vehicles and if a message with revoked certificate is observed then roadside unit broadcasts warning messages. In another option it is suggested that each passing-by vehicle get its certificate signed from roadside unit. These signatures are then used to show that the certificate has not been revoked. First option is open to attacks (malicious node does not transmit within range of a roadside unit) and second increases complexity and overhead.

Our solution comes closer to the method presented by [5, 6]. In [5], Fisher et al. used a large number of pseudonyms (defined as Inter-Vehicle-Communication-IVC certificates) to achieve un-link-ability. These pseudonyms are blindly signed by IVC certification servers' (ICS) private key. The private signing key is shared amongst multiple ICS by means of Secret Sharing. An IVC certificate is distributedly calculated through a quorum of ICS. For non-repudiation a tag, that can be linked to the vehicle, is generated/stored by ICS and is protected by a secret key shared amongst ICS. The solution requires transactions with multiple servers to get a pseudonym which may be difficult due to intermittent connectivity in VANET. Further, a pseudonym cannot be revoked during its validity period, and no definite solution to malformed pseudonyms (having validity larger than defined maximum period) has been defined. In [6], Schaub et al. also use pseudonyms to achieve un-link-ability. The pseudonyms are issued by pseudonym providers (PP_k) based on V-tokens (that also later form part of pseudonyms), V-tokens cannot be linked to the each other or to the owner by PP_k thus ensuring privacy. V-tokens, containing identifying information of the vehicle and Certification Authority (CA), are blindly signed by CA after being encrypted by vehicle with public key of resolution authority (RA). The decryption ability of V-tokens (i.e, resolution/non-repudiation) is distributed using threshold encryption scheme. The solution relies on cut-and-choose method to ensure well-form-ness of V-tokens, thus adding overheads in addition to the need of TPD (to store the V-tokens or corresponding pseudonyms). Further, the revocation method only revokes long-term identity and does not address already issued pseudonyms/V-tokens which may continue to be used for malicious purpose.

VIII. Conclusion and Future Work

Security architecture for VANET must cater for competing security attributes. Privacy, one of the important security attributes, competes with other attributes such as non-repudiation, revocation etc. We have proposed a security architecture that is based on revised Blind signature scheme. We have revised the Blind signature scheme to ensure provision of all the security attributes. The solution does not require tamper-proof-devices or multiple interactive transactions. Non-repudiation/revocation requires cooperation between multiple entities thus ensuring privacy without a single point of failure. In future work, we intend to further enhance the privacy by using mechanism which makes identification of issuing authority difficult. One such option is to incorporate group signatures. We also intend to introduce hierarchical certificate architecture with multiple long-term certificates to shorten the chain and thus simplifying revocation/non-repudiation procedures. Further, we also intend to extend the idea to protocol specification and test the protocol for performance and security.

References

[1] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J. P. Hubaux, "Secure vehicular communication systems: design and architecture", In IEEE Wireless Communication Magazine, November 2008.

[2] F. Armknecht, A. Festag, D. Westhoff, and K. Zang, "Cross-layer privacy enhancement and non-repudiation in vehicular communication", In WMAN'07.

[3] C. I. Fan, R. H. Hsu, and C. H. Tseng, "Pairing-based message authentication scheme with privacy protection in vehicular ad hoc network", In ACM Mobility'08.

[4] S.U. Rahman and U. Hengartner, "Secure crash reporting in vehicular ad hoc networks", In SecureComm'07.

[5] L. Fischer, A. Aijaz, C. Eckert, and D. Vogt. "Secure Revocable Anonymous Authenticated Inter-Vehicle Communication", ESCAR'06.

[6] F. Schaub, F. Kargl, Z. Ma, and M. Weber, "V-tokens for Conditional Pseudonymity in VANETs", ", In IEEE WCNC'10.

[7] D.Chaum, "Blind signatures for untraceable payments", In Advances in Cryptography, CRYPTO 82, PlenumPress. 1983: pp.199-20.

[8] X. Lin, X. Sun, P. H. Ho, and X. Shen, "GSIS: a secure and privacy preserving protocol for vehicular communications", IEEE Transaction on Vehicular Technology, Vol.56, No.6, pp.3442-3456, 2007.