I-PRE: Improved Proxy Re-Encryption Scheme for VANET

Arun Malik[1], Dr. Sanjeev Rana[2] and Dr. Rajneesh Gujral[3]

[1] Asian Institute of Management and Technology, Yamunagar, Haryana, India

[2] M. M. University, Computer Science and Engineering Department, Ambala, Haryana , India

[3] M. M. University Department, Computer Science and Engineering Department, Ambala, Haryana , India

Abstract

Vehicular Ad hoc Network (VANET) is a network of self organized vehicles and road-side infrastructure communicating with each other over wireless, with a view to improve traffic safety and efficiency. Data interchanged over VANETs often play a vital role in traffic safety. Such information must be accurate and truthful, as lives could depend on this application. Thus, Authentication is one of the important services for traffic safety in VANET. In this paper, we discuss various approaches of authentication in VANET such as Digital Certificates (DCA), Pairing and Proxy Re-encryption (PRE). PRE is a better solution among all existing authentication approaches but still it is vulnerable to many attacks i.e. Denial of Service. We proposed an improvement in the PRE (I-PRE) that overcome the exiting attacks on PRE. The effectiveness of the proposed scheme is analysed using OMNet++ simulator.

Keywords: *Authentication, Proxy re-encryption, VANET.*

1. Introduction

VANETs are a subset of Mobile Ad hoc Network in which communication nodes are mainly vehicles. There are many entities involved in a VANET settlement and deployment. Although the vast majority of VANET nodes are vehicles, there are other entities that perform basic operations in these networks i.e. Access point (AP), Service Provider (SP). VANET entities communicate with each other in many different ways, i.e. Vehicle-to-Vehicle communications (V2V), Vehicle-to-Infrastructure (V2I), in order to get some services [6]. The SPs and the APs can communicate with each other by some application-layer proprietary protocols via Internet. [2]. This infrastructure is assumed to be located along the roads [1] The APs are deployed along the roadside with reasonable wireless coverage to facilitate communication. A car typically belongs to one wireless network service provider, and communicates with the APs for accessing the internet along the road it travels through. When it travels, it also roams into wireless coverage that provide by other authorities.

Data interchanged over VANETs often play a vital role in traffic safety. Such information must be accurate and truthful, as lives could depend on this application. Therefore, Authentication is one of the important services for traffic safety in VANET. The attacker pretends to be another entity by stealing other entity's credential to get some benefit [7][8]. To make the authentication process time-efficient, traditional solutions using centralized authentication server (AS) is not preferable because of the large amount of messages exchanged among the car, the APs and the ASes. If the overlay network interconnecting the APs and the ASes is based on Internet, the delay for exchanging authentication messages could be minimized by reducing the communication duration between the fast moving car and an individual AP [9]. In order to meet this requirement, the authentication protocols must involves as less parties as possible besides the car, AP and ASes over Internet in order to control number of authentication messages.

In this paper, we discuss the major existing authentication approaches of VANET. PRE is a better solution among all existing authentication approaches but still it is vulnerable to many attacks i.e. Denial of Service. We proposed an improvement in the PRE (I-PRE) that overcome the exiting attacks on PRE.

2. Existing Authentication Approaches of VANET

In VANET, There are three major authentication approaches that are digital certificates based authentication

(DCA), authentication using Pairing, and Proxy Re-encryption (PRE) [3]. The detail working of these approaches are described next:

2.1 Authentication using Digital Certificate (DCA)

Earlier digital certificate are used to conduct the car-to-AP authentication. The SP partitions the service duration into time slots. When the car signs up at the SP, SP assigns a series of the car's public keys PK_{CAR} (ti) and their digital certificates $Cert_{CAR}$ (ti) to the car. Only one specific public key and digital certificate pair can be used in the corresponding time slot during subscription. For each time slot during the SP's service, the SP has a corresponding public key. The SP also sends its own time-related public keys $PKSP$ (ti) to the car. The SP administrates a large number of distributed APs and monitors the behavior of them. The SP distributes its time-related public keys to the APs periodically for the upcoming time slots. The DCA Method is explained as follows:

Step 1: As shown in Figure 1, the authentication request is initiated by the car. According to its clock, it gets the time $t1$ and the corresponding public key $PK_{CAR}(t1)$ and certificate $Cert_{CAR}(t1)$ issued by SP. The car sends a message consisting of the three data fields $<t1, PK_{CAR}(t1), Cert_{CAR}(t1)>$ to the AP.

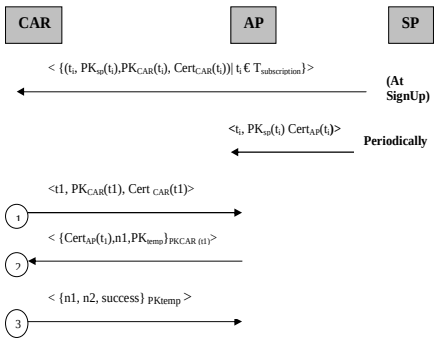

Fig 1: Authentication using DCA

Step 2: After the AP receives these messages, it checks $t1$. If it considers $t1$ unacceptable with regard to a deviation threshold, it can either simply disregard the request, or send a time-correction message to the car in order for it to have its clock adjusted.

Step 3: After the time adjusting, the car can initiate the authentication request again. If the time is validated, the AP tries to verify the certificate of the car's public key carried in the authentication request message by the SP's public key corresponding to $t1$.

Step 4: If the verification is successful, it randomly chooses a nonce $n1$ and generates a temporary public key $PKtemp$. After encrypting them by the PK_{CAR} ($t1$) provided in the request, the AP sends the message back to the car. The car can decrypt the message and get $n1$.

Step 5: After generating another nonce $n2$, it can send verification to the AP consisting $n1$, $n2$ and a success tag encrypted altogether using PK_{temp}. The AP can decrypt the message and get $n2$. Both parties can use some method E to generate session secret key from $n1$ and $n2$.

Step 6: The session key $E(n1, n2)$ is used for the data communication. The last verification message can be also piggybacked to the first data packet sent by the car. Hence authentication is successfully maintained.

2.2 Authentication Using Pairing (PA)

Pairing mechanism can also be used for authentication between the car and the AP. The basic idea of pairing mechanism is that a security authority (SA) can issue pseudonym/secret point pairs based on a master secret. Without the knowledge of the master secret, any two parties who possess a pseudonym/secret point pair can present pseudonyms to each other and a common secret key can be established. The pairing method is explained as follows: -

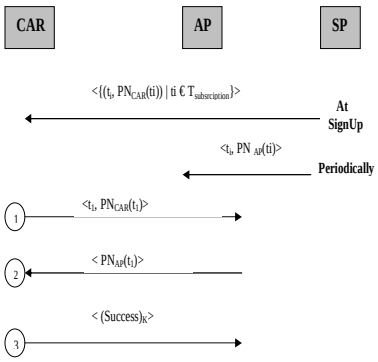

Fig 2: Authentication Using Pairing

Step 1: During sign-up stage, when the car subscribes service from the SP, a series of pseudonym/secret point pairs are assigned to the car, with each pair being used in a time slot of subscription. The number of pairs is determined by the subscription length. The APs also get these pseudonym and

secret point pairs, but in a periodic way similar to that of DCA. The SP stops assigning these pairs to an AP if the AP's found misbehaving.

Step 2: The authentication message exchange still involves a three-way handshake. As shown in Figure 2, the car initiates an authentication by sending a request message to the AP: < $t1$, PN_{CAR} ($t1$) >. The message contains a timestamp $t1$ and the car's pseudonym PN_{CAR} ($t1$) bounded to that timestamp.

Step 3: If the time provided by the car is within normal deviation, the service provider picks one of its secret points corresponding to the time provided by the car and computes a shared secret key K; otherwise it can initiate time synchronization with the car as mentioned before.

Step 4: It then replies the car with a message containing the pseudonym just used to generate the secret key K: < PN_{AP} ($t1$) >. After the car receives the message, it can calculate the same secret key K based on the pseudonym provided by the AP.

Step 5: The car then encrypts a tag indicating successful authentication with the common secret key K and sends the message to the AP. After the AP confirms the message, the trust relationship between the car and the AP is established.

2.3 Authentication using Proxy Re-encryption (PRE)

Proxy re-encryption is a concept introduced by Blaze et al [10] in that allows a semi trusted entity called the "proxy" to convert cipher texts addressed to an entity B called the "delegators" to another entity C called the "delegate", while maintaining that the proxy cannot learn anything about the underlying plaintext, and C cannot learn anything about the underlying plaintext without co-operation from the proxy. B does this delegation by providing a special piece of information, called the "rekey", to the proxy. Proxy re-encryption has found various applications like secure email forwarding, etc.

The basic concept of proxy re-encryption says that, a cipher text for Alice that is encrypted by Alice's public key can be transformed by a proxy to a cipher text for Bob that can be decrypted by Bob's private key. The proxy however cannot read the cipher text. In this procedure, Alice delegates her decryption right to Bob. The key that the proxy uses to do the transformation is called re-encryption key $rk_{a \to b}$.

In VANET, a car first needs to subscribe from a service provider SP. The car is assigned a pair of public and private keys at signup. For each time slot the SP has a public key PK_{SP} (ti). According to the subscription contract, the SP assign a series of re-encryption keys $ReKey_{CAR}$ (ti) corresponding to the time slots in subscription duration, by which the car can re-encrypt a message originally encrypted by the SP's public key to generate a cipher text encrypted by its own public key. The authentication process is depicted in Figure 3 and explained as follows:-

Step 1: The car sends an authentication request to the AP detected in its range. The request message just contains the time of request t and a random number n1: <t1, n1>.

Step 2: After the AP receives this message, it compares the time t1 provided by the car to its own clock. If the time is considered to be within normal deviation, the access point sends a message back to the car. The message constitutes a new random number n2 encrypted by the public key of the service provider of the time slot related to t1.
t1: < (n2) PK_{SP} (t1) >.

Step 3: After the car receives the reply, it uses the re-encryption key corresponding to t1 to re-encrypt the message. The outcome is thus available for it to decrypt using its own private key, and the n2 is revealed.

Step 4: It then takes n1 and n2, combines them by some cryptographic algorithm E known to both parties to generate E(n1, n2), and uses it as a symmetric key to encrypt a success tag as the authentication proof.

Step 5: The encrypted message is sent back to the AP separately, or the car can also choose to immediately start sending data packets, with the authentication proof piggy-backed to the first data packet.

Step 6: After the AP verifies the message by decrypting it using E(n1, n2), a secure and trusted connection is established. For the AP to show itself as authorized, it needs to answer a challenge just as it posts to the car. For this purpose the AP needs to get time-related re-encryption keys along with the SP's public keys from the SP in a periodic fashion.

When the car initiates authentication request, besides the timestamp, the nonce n1 is encrypted by the current public key of the SP as a challenge. After the AP receives the request, it can use re-encryption to resolve the challenge. In the response message, besides the challenge message to the car, it includes the proof of re-encryption capability by a success tag encrypted using n1.

Among three authentication approaches, DCA and PA required session keys that is are used during authentication process whereas PRE has the higher level of anonymity it achieves. In PRE, the cryptographic material (re-

encryption key) is not included in any of the authentication messages exchanged. Instead, the car only uses the re-encryption key to respond to the challenge from the AP. So, that is why PRE method is preferred over DCA and PA approaches of authentication. But still, the PRE approach has various possible common attacks and hence is not suitable for secure VANET communication. The comparison between these approaches is shown in Table 1.

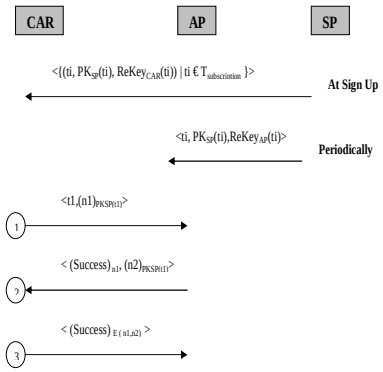

Fig 3: Authentication using Proxy Re-encryption

Table 1 Comparison between existing authentication methods

	Encryption Technique	Messages required for authentication	Point of compromise
Digital Certificate	Asymmetric key	Large no of messages required for authentication	If a node knows the public key of the signing node.
Pairing	Symmetric key	Extra messages not required	If an attacker gets the secret key of communication
Proxy Re-encryption	Re-Encryption	AP to show itself as authorized	If re-encryption key is compromised

3. Improved Proxy Re-encryption (I-PRE)

The Improved Proxy Re-encryption (I-PRE) comprises of all the features of earlier Proxy Re-encryption (PRE). We added a one more private key between a car and access point. The working of I-PRE is shown in figure 4.

ALGORITHM: Improved Proxy Re-Encryption (I-PRE)

Step 1: A pair of public and private key is assigned at sign up.

Step 2: The Car sends time slot t1 and nonce n1 and an encrypted private key <$\mathbf{PrK_{ca}}$> to the AP. Since this private key is also known to the AP, it will decrypt it and check with its own private key.

Step 3: After the two keys matches and the time t1 provided by the car comparable to its own clock, the AP sends a message back to the car. The message constitutes a new random number n2 encrypted by the public key of the service provider of the time slot corresponding to < t1, E($\mathbf{PrK_{ca}}$) > : < (n2) PK_{SP} (t1), $\mathbf{PrK_{ca}}$ >.

Step 4: After the car receives the reply, it uses the re-encryption key corresponding to t1 to re-encrypt the message. The outcome is thus available for it to decrypt using its own private key, and the n2 is revealed.

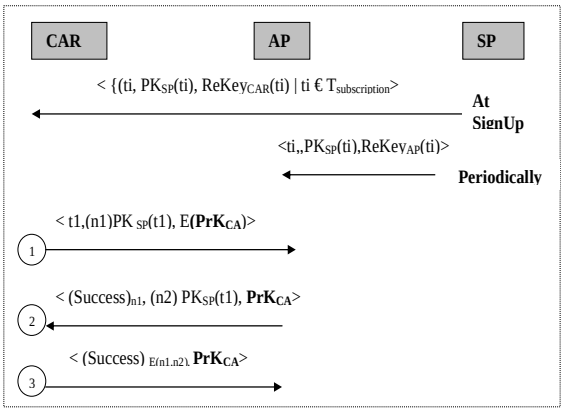

Fig 4: Working of I-PRE

Step 5: It then takes n1 and n2, combines them by some cryptographic algorithm E known to both parties to generate E(n1, n2), and uses it as a symmetric key to encrypt a success tag as the authentication proof.

4. Implementation of PRE and I-PRE

The proposed solution is analysed using OMNet++ simulator. In simulation, Random mobility of nodes is considered for PRE and proposed I-PRE. Following parameters are evaluated.

A. Throughput (b/s) Vs Speed (m/s): Throughput measures the data rate at which information is exchanged in the network. The formula for throughput is:

Throughput = No. of bytes delivered/sec

Table 2: shows values for Throughput Vs Speed for I-PRE and PRE

Speed (m/s)	Throughput (b/s)	
	I-PRE	PRE
60	4450	4200
70	4427	4159
80	4400	4120
90	4400	4112
100	4400	4100
110	4400	4100
120	4400	4100

Figure 5 shows that, with the regular increase in the speed of the vehicle, the throughput decreases gradually and after some point becomes constant approximately for the remaining speeds. But the result in case of I-PRE is better than PRE method.

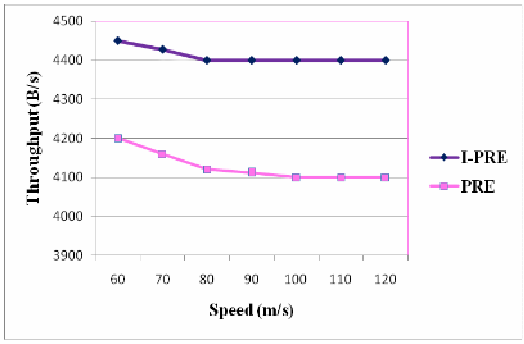

Fig 5: Comparison of Throughput Vs Speed

B. Average Delay(s) Vs Speed (m/s): Latency specifies the delay in the packet transmission. Table 3 show values for Average Delay Vs Speed for I-PRE and PRE.

Figure 6 shows that, the delay in case of I-PRE authentication method is less as compared to the PRE method if considered in Denial of Service (Dos) and masquerading attacks. The average delay is measured in seconds.

C. Average Jitter (s) Vs Speed (m/s): Jitter characterizes the variation of the latency (End-to-End Delay). Table 4 show values for Average Jitter Vs Speed for I-PRE and PRE.

Table 3: Average Delay (s) Vs Speed (m/s)

Speed (m/s)	Average Delay (s)	
	I-PRE	PRE
60	0.0074	0.0083
70	0.0078	0.0086
80	0.0082	0.0089
90	0.0079	0.0082
100	0.0073	0.008
110	0.0076	0.0084
120	0.008	0.0086

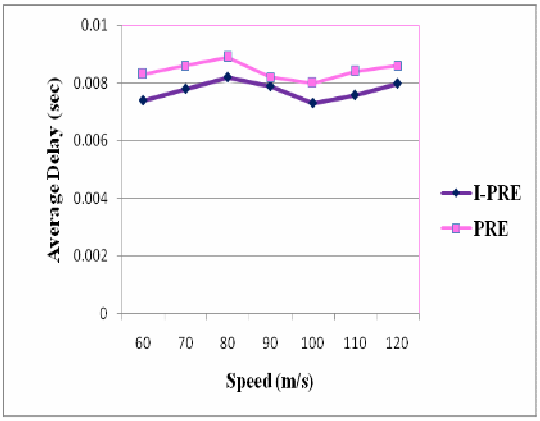

Fig 6: Comparison of Average Delay Vs Speed

It can be seen from the Figure 7 that, the average jitter in case of I-PRE authentication method is less as compared to the PRE method if considered in Denial of Service (DoS) and masquerading attacks. Less the average jitter, better is the transmission of information between vehicles and better will be the communication also. So, I-PRE is efficient than PRE.

Table 4: Average Jitter (s) Vs Speed (m/s)

Speed (m/s)	Average Jitter (s)	
	I-PRE	PRE
60	0.00035	0.00041
70	0.00039	0.00044
80	0.00041	0.00046
90	0.00037	0.00042
100	0.00033	0.0004
110	0.00036	0.00043
120	0.00037	0.00045

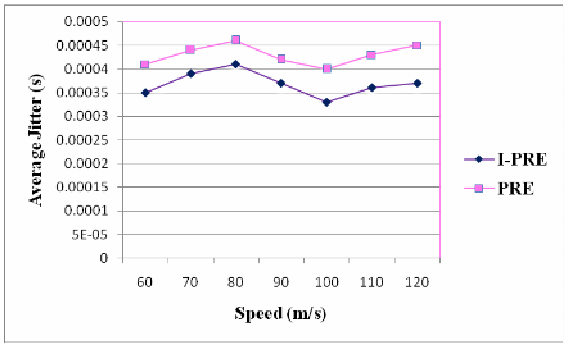

Fig 7: Comparison of Average Jitter Vs Speed

D. Packet Delivery Ratio Vs Speed (m/s): Packet Delivery Ratio: - Packet delivery ratio specifies the ratio of how many packets are transmitted from sender and how many are received to the receiver.

It can be seen from the Figure 8, that the packet delivery ratio in case of I-PRE authentication method is better as compared to the PRE method if considered in random mobility scenario. With the gradual increase in the speed, it is seen that the packet transmission and reception becomes better in case of I-PRE and hence the packet delivery ratio is increased in I-PRE.

Tables 5: Packet Delivery Ratio Vs Speed (m/s) for I-PRE and PRE method.

Speed (m/s)	Packet Delivery Ratio	
	I-PRE	PRE
60	0.91	0.833
70	0.9	0.81
80	0.87	0.79
90	0.85	0.78
100	0.88	0.77
110	0.84	0.74
120	0.85	0.75

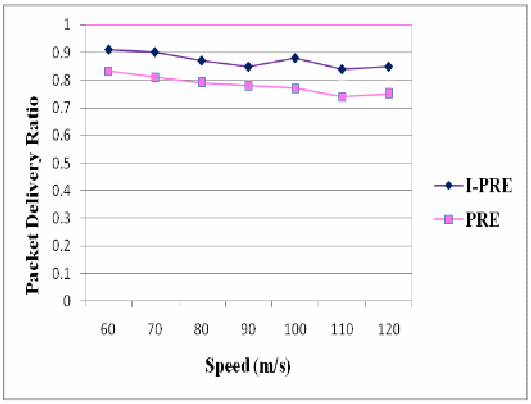

Fig 8: Comparison of Packet Delivery Ratio Vs Speed

5. Conclusion

Nowadays, vehicular networks are being developed and improved. As VANET based applications have impact in road traffic safety, strong security requirements must be achieved. In this paper, we discussed and analyzed different existing VANET authentication approaches. We proposed a security solution I-PRE that is an improvement in the existing PRE scheme which not only provide strong authentication among different entities of VANET but also prevent from various attacks i.e. DoS attack.
We also analyzed the effectiveness of I-PRE over PRE using OMNet++ simulator. Result also shows that I-PRE outperforms PRE by providing better throughput, reducing latency and jitter and increasing the packet delivery ratio by providing solutions for attacks. The proposed solution uses encryptions which have major impacts on its performance since it will use more processing power and time. In future, we will focus on some efficient signature scheme in order to reduce computation time required for security solutions.

6. References

[1] José María de Fuentes, Ana Isabel González-Tablas, Arturo Ribagorda, “Overview of security issues in Vehicular Ad-hoc Networks “, Handbook of Research on Mobility and Computing, pp 1-17, 2010

[2] Y Do, S Buchegger, T Alpcan, and J P Hubaux. “Centrality Analysis in Vehicular Networks”. Technical report, 2008. [30] Maxim Raya and Jean-Pierre Hubaux. “Securing Vehicular Ad Hoc Networks”. Journal of Computer Security, Special Issue on Security of Ad Hoc and Sensor Networks, 15(1):39 – 68, 2007.

[3] Imrich Chlamtac, Marco Conti, and Jennifer J Liu. “Mobile Ad Hoc Networking: “Imperatives and Challenges”. Ad Hoc Networks Ad Hoc Networking book contents, volume1 (1): pages:13–64, Jul 2003.

[4] Xiaonan Liu, Zhiyi Fang, Lijun Shi. “Securing Vehicular Ad Hoc Networks”, School of Computer Science and Technology, Jilin University Changchun, 130012, P.R China Lxn6O2@sina. com, zyfang@public. ccjl. Cn, Page(s): 424-429, Digital Object Identifier 10.1109/ICPCA.2007.4365481.

[5] Y. Qian and N. Moayeri. “Design of Secure and Application-Oriented VANETs”. In Vehicular Technology Conference, pages 2794–2799. VTC Spring 2008, IEEE, 2008

[6] Andreas Festag, Roberto Baldessari, Wenhui Zhang and Long Le, "CAR 2 CAR Communication Consortium Manifesto versions 1.1. Technical report" Workshop on IEEE Vehicular networking, CAR 2 CAR Communication Consortium (C2C-CC), Aug 2009.

[7] S. Schecheter, T. Parnell, and A. Hartemink. Anonymous authentication of membership in dyanmic group. January 1999.

[8] K. Sha, Y. Xi, W. Shi, L. Schwiebert, and T. Zhang. Adaptive privacypreserving authentication in vehicular networks. In Proceedings of IEEE International Workshop on Vehicle Communication and Applications, pp.1-8, 2006.

[9] Y. Xi, K. Sha, W. Shi, L. Schwiebert, and T. Zhang. Probabilistic adaptive anonymous authentication in vehicular networks. In Journal of Computer Science and Technology, Nov. 2008.

[10] M. Blaze, G. Bleumer, and M. Strauss. "Divertible protocols and atomic proxy cryptography". In Eurocrypt'98, LNCS 1403, 1998.