Lightweight VANET Authentication Protocols

Otto B. Piramuthu
Computer Science, UIUC
Urbana, Illinois 61801, USA
obp2@illinois.edu

Matthew Caesar
Computer Science, UIUC
Urbana, Illinois 61801, USA
caesar@illinois.edu

Abstract

Security and privacy of vehicles, occupants of such vehicles, roadside infrastructure, and other entities that are part of Vehicular Ad hoc NETwork (VANET) cannot be overstated. Cryptography is commonly used to authenticate and to secure communication among VANET entities. As vehicles are mobile, it is essential for authentication protocols to be lightweight, quick, and with minimal number of passed messages. It is also necessary to ensure that these protocols are secure against attacks. However, extant authentication protocols are not necessarily lightweight and almost all of them are vulnerable to relay attacks. We propose secure and truly *lightweight* authentication protocols for the VANET environment.

CCS Concepts

• **Security and privacy** → **Authentication**;

Keywords

VANET, authentication, lightweight)

ACM Reference Format:
Otto B. Piramuthu and Matthew Caesar. 2022. Lightweight VANET Authentication Protocols. In *The 37th ACM/SIGAPP Symposium on Applied Computing (SAC '22), April 25–29, 2022, Virtual Event,* . ACM, New York, NY, USA, Article 4, 4 pages. https://doi.org/10.1145/3477314.3507178

1 Introduction

VANETs facilitate seamless integration of communication among nodes in such networks that include vehicles, roadside units (RSU), and trusted authorities (TA). Communication among mobile VANET nodes (e.g., vehicles) may not be secure and occurs through wireless network technology while those between stationary nodes (e.g., RSU, TA) generally occur via secure wired channels. VANETs assist with vehicle navigation (e.g., information on traffic jams), road safety (e.g., information on compromised road conditions), among others through messages that are passed between vehicles (V2V) as well as between vehicles and infrastructure/everything (V2I/ V2X). RSUs mediate messages among vehicles as well as broadcast messages of interest to vehicles nearby. Given their significance, the messages that are passed among entities in VANETs need to be secured [4] to prevent their misuse or attacks from adversaries [1]. Security is operationalized through authentication of the communicating entities to prevent various forms of attacks (e.g., man-in-the-middle, replay). Over the years, researchers have developed several authentication protocols with the goal to ensure security and privacy of the entities as well as the passed messages in VANETs.

While these protocols consider various vehicle-RSU-back-end configurations and attack scenarios, to our knowledge, none of the VANET authentication protocols consider the possibility of relay attacks. Relay attacks involve attacker(s) who relay messages between two entities that then falsely believe they are indeed in direct communication with each other [2]. Since the attacker(s) do not modify any message, the sender and receiver of the message may not even be aware of the presence of attacker(s). Relay attacks are dangerous due to their surreptitious nature and the resulting damage.

We address this void in extant VANET authentication protocol literature and develop protocols that are secure against attacks in general and relay attacks in particular. Relay attacks have the potential to wreak havoc in VANETs. For example, relay attacks can be used to misrepresent location information, which is an important aspect in VANETs. Relay attacks operate primarily through extension of the distance between communicating entities. The result is that these entities are made to believe that they are in close physical proximity to each other than in reality. An example of relay attack in a non-VANET (https://www.locksmiths.co.uk/faq/keyless-car-theft/) context is vehicle theft that compromises a remote keyless system (RKS) to extend the radio signal range between the key fob which is inside the house and the car parked just outside the house [5]. The car then believes that the key fob is physically close and in its signal range. Note that we are not interested in such RKS-based relay attacks between stationary vehicles and their key fobs but rather in preventing messages that are relayed between VANET entities without the knowledge of sender and/or receiver of such messages.

In the absence of relay attacks, direct communication is possible only between entities that are within their communication range. When entities (i.e., vehicles, RSU) are physically close together, their ambient conditions are bound to be similar. Examples of ambient conditions include temperature, atmospheric pressure, and relative humidity. Their physical separation can also be determined through their GPS coordinates. Therefore, when ambient conditions are abnormally different between the source and destination nodes of a *direct* message, there is a high likelihood of a relay attack.

While not all ambient conditions are appropriate under all circumstances, an appropriate set can be chosen based on context. For example, in a mountainous area, the atmospheric pressure at two vehicle locations that are not farther apart might be quite different based on the elevation differences between these locations. However, their GPS coordinates might provide a good approximation of their physical separation. To accommodate such ambient condition differences even between two physically close locations, we allow

SAC '22, April 25–29, 2022, Virtual Event,

ACM ISBN 978-1-4503-8713-2/22/04…$15.00
https://doi.org/10.1145/3477314.3507178

the simultaneous use of multiple ambient condition information in our authentication protocols as defense against relay attacks. We therefore incorporate ambient condition information at the sender's location which the message recipient can use to verify. Specifically, we use GPS information and at least one other ambient condition information at the sender's location to corroborate location identification since the message recipient can receive direct messages from only so far. While GPS coordinates can be easily spoofed, encrypted GPS coordinates cannot be easily modified. When exact GPS coordinates are unavailable (e.g., in tunnels), the other ambient conditions bridge the gap along with the last known GPS location information.

2 The Proposed Protocols

When a vehicle enters the range covered by an RSU, it requires the local group key for communication with that RSU. Therefore, among the first protocol of interest to a vehicle entering an RSU's field is that to register with the TA and local RSU as well as receive the group key. The group key can be used by RSU or TA to share messages of general interest to the vehicles in that area. To this end, we propose a group key sharing protocol. Another aspect of VANET includes community-generated alerts where vehicles inform the TA or RSU (e.g., oil-spilled slick part of a road), which verifies and shares this information with the rest of the VANET entities in that area. We develop an authentication protocol for this purpose. While a rogue vehicle can generate a fake alert, the identity associated with that message helps ensure identification of the culprit. We consider the required characteristics before presenting the truly lightweight authentication protocols that use only concatenation, XOR, and rotation operations, while avoiding the use of expensive operations.

2.1 Adversary Model

Possible threats in a VANET environment could primarily come from manipulated messages. Since we use location information to prevent relay attacks, location information manipulation by unauthorized entities is a possibility. As one of the primary elements of VANET communication is reporting of roadside issues by vehicles, manipulation of the reported messages is also a possibility.

2.2 Security Model

We assume that the adversary follows the Dolev-Yao intruder model [3] with the ability to freely monitor, block, eavesdrop, inject, and modify messages. The trusted authority (TA) cannot be compromised and is the only entity that knows the real identity of each vehicle. The TA is the only entity that is authorized to authenticate vehicles and RSUs. The RSUs are assumed to be vulnerable to attacks from resourceful adversaries. The following are assumed.

- The trusted authority cannot be compromised, but the vehicles and RSUs can be compromised and so cannot be trusted.
- Adversaries can monitor, block, eavesdrop, and modify any message(s) passed between vehicles and RSUs as well as inject new message(s) in the vehicle-RSU wireless channel.
- The wired communication channel between RSUs and TA is secure and adversaries cannot monitor, block, eavesdrop, or modify any message nor inject message(s) in this channel.
- Adversaries cannot retrieve any shared secret (private) keys from transmitted messages. Adversaries also cannot decrypt any of the message(s) passed in the vehicle-RSU channel

2.3 Authentication Protocols

The notation used in the rest of this paper follows.

V_i	vehicle i
R_j	road-side unit j
TA	trusted authority
r_i, r_j	nonce generated by vehicle$_i$ and RSU$_j$
i, j	subscripts denoting vehicle$_i$ and RSU$_j$
ID_i, ID_j	identifiers for vehicle$_i$ and RSU$_j$
AID_i, AID_{old_i}	current, previous anonymous identity of V_i
$k_1^i, k_2^i, k_3^i, k_4^i$	set of shared keys between V_i and TA
k_j^g	group key for vehicles in R_j's field
k_{j-TA}	shared secret key between R_j and TA
A_i	ambient condition information at entity i
M_{ij}, M_{ji}	message from/to vehicle$_i$ to/from RSU$_j$
T_i	time stamp from entity i
δ_j	farthest signal travel distance for RSU$_j$
$H(r)$	Hamming weight of r
$Rot^R_{H(r)}(X)$	right rotate X by the Hamming weight of r
$Rot^L_{H(r)}(X)$	left rotate X by the Hamming weight of r
$\|\|$	concatenation operator
$\oplus$	exclusive-OR operator

Since message passing is an important task in VANETs, the following protocols ensure that the communicating parties are authenticated and the security and privacy of messages are maintained. These protocols are for communication between a vehicle and an RSU and the generation and sharing of group key for group communication.

2.3.1 Sharing Group Key

Each vehicle privately communicates with RSU to receive or pass message (e.g., on local road conditions). On the other hand, RSU communicates with all vehicles in its range (e.g., important information on road closure due to an accident). As shared key between two entities is necessary to accomplish secure (symmetric key cryptography) one-to-one communication between the entities, a similar setup is seen in group communication. The core requirement in these scenarios is the shared secret key between entities that communicate with each other. To this end, when an RSU wishes to share a message with all vehicles in its range, it uses a group key to encrypt its message, which is then decrypted by the vehicles through use of that group key. Since all vehicles in the range know the group key, communication can be secured with such a group key. We now show how this can be accomplished as a part of an authentication protocol. We use only lightweight operations such as exclusive-OR, concatenation, and rotation to encrypt messages. A first step in this process is the generation and transmission of the group key.

When a vehicle enters the field of an RSU, it generates a nonce r_i, its ambient condition A_i, and timestamp T_i (Fig. 1). The vehicle then generates Z_i, which is one of $(A_i||r_i||T_i)$, $(T_i||A_i||r_i)$, or $(r_i||T_i||A_i)$ with equal probability. We do this to ensure that the order of the three terms (A_i, r_i, and T_i) is not predictable. Vehicle i then takes exclusive-OR of Z_i and concatenation of keys it shares with the TA ($a \leftarrow Z_i \oplus (k_1^i||k_2^i||k_3^i \oplus k_4^i)$). To reinforce security of the encrypted terms, vehicle i also generates b ($b \leftarrow Rot^L_{H(r_i)}(A_i \oplus T_i \oplus r_i)$). Next, i transmits (AID_i, a, b) to the RSU. Note that the RSU is not privy to the shared keys k_1^i, k_2^i, k_3^i. While the communication channel between vehicle and RSU is not assumed secure, the channel between RSU

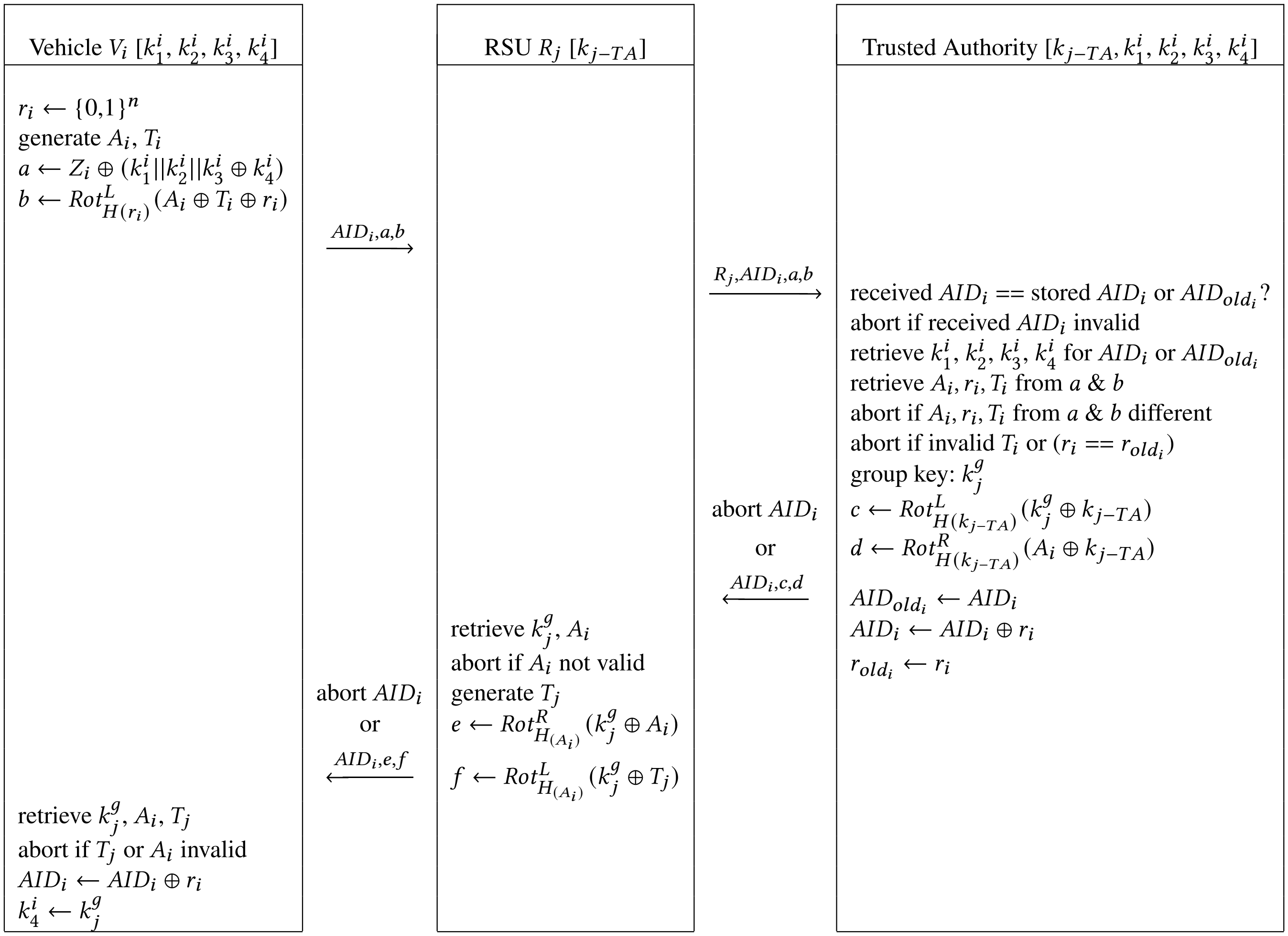

Figure 1: Protocol to register vehicle V_i at RSU$_j$ and receive group key [shared keys in square brackets]

and TA is wired and assumed secure. The RSU relays the message from i to the TA. Upon reception of this message, the TA locates and retrieves the shared keys ($k_1^i, k_2^i, k_3^i, k_3^i$) based on AID_i. When the trusted authority receives a, it tries all three of these variations and chooses the one that matches the T_i closest to current time. If none of these sets prove to be a match that is closest to current time, the trusted authority aborts the protocol.

The TA aborts the protocol if the received AID_i or T_i prove to be invalid, (A_i, r_i, or T_i) from a and b are different, or the previous r_i is reused. The TA then encrypts the group key (k_j^g) and A_i with the shared key (k_{j-TA}) to generate c and d which are forwarded to RSU j. The TA stores AID_i and r_i in case these are needed for the next authentication round with vehicle i and then updates AID_i. Upon reception of message from TA, RSU j decrypts the message and aborts the protocol if A_i is invalid. Otherwise, it generates timestamp T_j and encrypts this as well as A_i and k_j^g with A_i to generate e and f which are then forwarded to vehicle i. Vehicle i aborts the protocol if either T_j or A_i is invalid. It then updates its AID_i and k_4^i. The message between V_i and RSU_j is resent (with updated A_i, r_i, and T_i) after a pre-specified amount of time if no response is received from the other side (i.e., R_j) during this time.

2.3.2 Message transmission from vehicle to RSU

The purpose of this protocol (Fig. 2) is to securely transmit message (M_{ij}) from vehicle V_i to RSU R_j with the help of the trusted authority. AID_i is an ephemeral vehicle V_i identifier which is updated after every authentication round and therefore cannot be used to track or trace V_i. The vehicle initiates the process by generating an n-bit nonce (r_i). The vehicle then generates its ambient condition information (A_i) and current time stamp (T_i). These terms (A_i, M_{ij}, r_i, and T_i) are then concatenated together as Z_i, which is exclusive-OR[d] with the shared keys ($k_1^i || k_2^i || k_3^i || k_4^i$). Each time a message is to be passed from vehicle V_i to RSU_j, V_i rotates the order of the four terms. I.e., one of ($A_i||M_{ij}||r_i||T_i$), ($M_{ij}||r_i||T_i||A_i$), ($r_i||T_i||A_i||M_{ij}$), or ($T_i||A_i||M_{ij}||r_i$) is chosen with equal probability to be Z_i. This is done to provide more variability in a. We use rotation again to resist modification of A_i, M_{ij}, r_i, or T_i in a by an adversary. To this end, e and f are respectively the right and left rotation of ($M_{ij} \oplus r_i$) and ($A_i \oplus T_i$) by the Hamming weight of r_i (i.e., $H(r_i)$). To randomize these rotated expressions further, we generate b as $e \oplus f$.

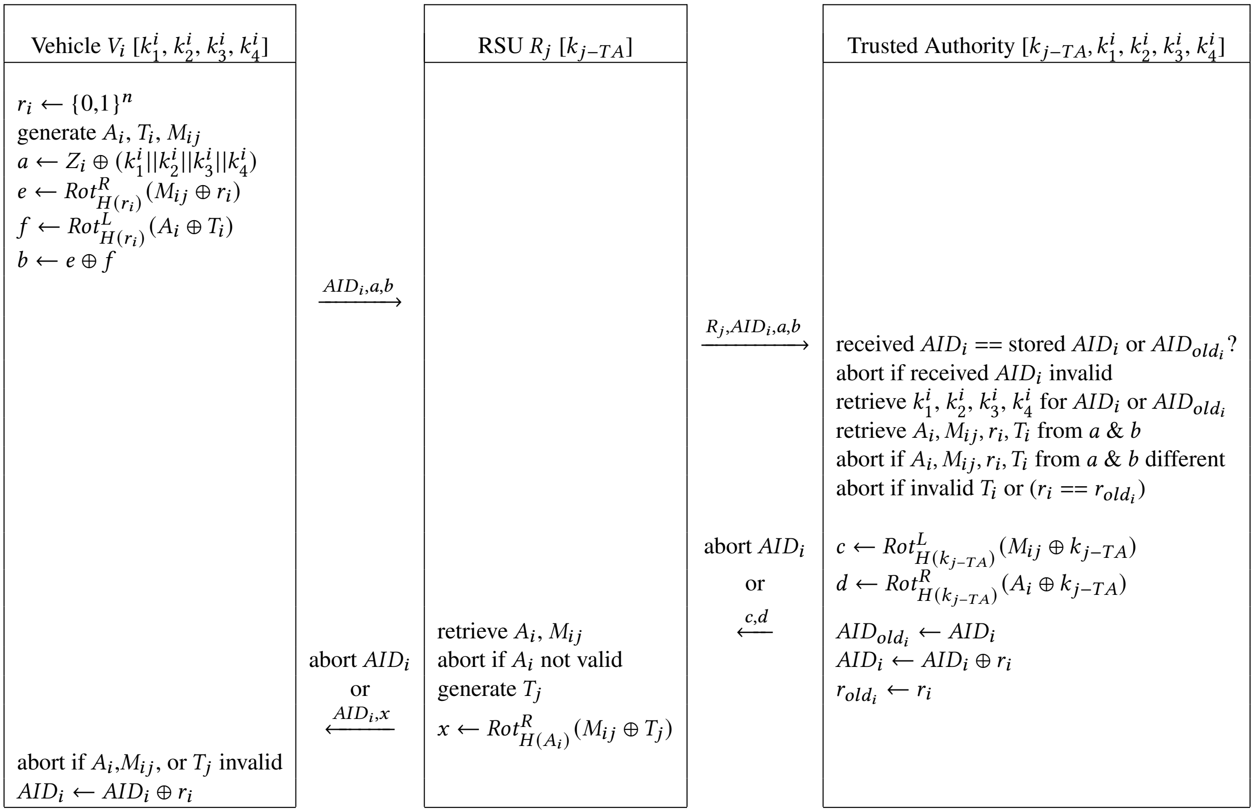

Figure 2: Protocol for message M_{ij} from vehicle V_i to RSU$_j$ [shared keys in square brackets]

The RSU R_j then sends its identifier (i.e., R_j) along with what it received from vehicle V_i (i.e., R_j, AID_i, a, b) to the trusted authority (TA). The TA validates the received AID_i from its two previous stored values (i.e., AID_i and AID_{old_i}). If this step fails, the TA aborts the protocol run. Based on AID, the TA retrieves the shared secret key set (k_1^i, k_2^i, k_3^i, k_4^i), which is then used to retrieve (A_i, M_{ij}, r_i, T_i) from a and b. The protocol run is aborted if any of the retrieved terms from a and b is different, T_i is invalid (i.e., it is not reasonably close to and less than the current time), or the previous r_i is used. The TA informs RSU R_j when an abort happens to a message from V_i. If not, the TA takes exclusive-OR of M_{ij} as well as A_i with its shared key with R_j (i.e., k_{j-TA}) and sends these as (c, d) to RSU R_j, which then retrieves M_{ij} and A_i. The TA updates its stored (AID_i, AID_{old_i}, and r_{old_i}) values. RSU R_j aborts the protocol run if A_i is found to be invalid (e.g., the GPS coordinates do not signify that V_i is within the read range of R_j). It then sends $M_{ij} \oplus A_i$ along with AID_i to vehicle V_i, which knows that this message is directed to it (from AID_i) and aborts the protocol run if M_{ij} or A_i is invalid. Else, it updates its AID_i. Note that we use GPS coordinate information for illustrative purpose only. Any relevant information based on context (e.g., temperature, atmospheric pressure) is appropriate. The message between V_i and RSU_j is resent (with updated A_i, r_i, and T_i) after a pre-specified amount of time if no response is received from the other side (i.e., R_j) during this time.

3 Conclusion

Our review of extant published literature in this general area revealed that although several types of attacks (e.g., replay, impersonation) are specifically considered to develop authentication protocols that are claimed to be resistant against such attacks, one type of attack is missing in such literature. Even though relay attacks have the potential to cause irreparable damage in VANETs, this type of attack seems to have been ignored in the VANET authentication protocol literature. Our goals are to bring this to the attention of VANET authentication protocol researchers and to develop lightweight authentication protocols that specifically target such attacks.

Acknowledgement

We thank Professor Ling Ren for his helpful input on earlier drafts of this work

References

[1] I. Ali, Y. Chen, N. Ullah, R. Kumar, W. He (2021) "An efficient and provably secure ECC-based conditional privacy-preserving authentication for vehicle-to-vehicle communication in VANETs." *IEEE Transactions on Vehicular Technology*.

[2] T. Beth, Y. Desmedt (1990) "Identification tokens - or: Solving the Chess grandmaster problem. *Advances in Cryptology-CRYPTO'90*, Springer LNCS 537, 169-176.

[3] D. Dolev, A.C. Yao 1983) "On the Security of Public Key Protocols." *IEEE Transactions on Information Theory*, 29(2), 198-207.

[4] M. Raya, J.-P. Hubaux (2007) "Securing vehicular ad hoc networks." *Journal of Computer Security*, 15(1), 39-68.

[5] B. Slater (2019) "How easy is it to steal your car?" https://www.which.co.uk/news/2019/01/how-easy-is-your-car-to-steal/