2FLIP: A Two-Factor Lightweight Privacy-Preserving Authentication Scheme for VANET

Fei Wang, *Member, IEEE*, Yongjun Xu, *Member, IEEE*, Hanwen Zhang, *Member, IEEE*, Yujun Zhang, *Member, IEEE*, and Liehuang Zhu, *Member, IEEE*

***Abstract*—Authentication in a vehicular ad-hoc network (VANET) requires not only secure and efficient authentication with privacy preservation but applicable flexibility to handle complicated transportation circumstances as well. In this paper, we proposed a Two-Factor LIghtweight Privacy-preserving authentication scheme (2FLIP) to enhance the security of VANET communication. 2FLIP employs the decentralized certificate authority (CA) and the biological-password-based two-factor authentication (2FA) to achieve the goals. Based on the decentralized CA, 2FLIP only requires several extremely lightweight hashing processes and a fast message-authentication-code operation for message signing and verification between vehicles. Compared with previous schemes, 2FLIP significantly reduces computation cost by 100–1000 times and decreases communication overhead by 55.24%–77.52%. Furthermore, any certificate revocation list (CRL)-related overhead on vehicles is avoided. 2FLIP makes the scheme resilient to denial-of-service attack in both computation and memory, which is caused by either deliberate invading behaviors or jammed traffic scenes. The proposed scheme provides strong privacy preservation that the adversaries can never succeed in tracing any vehicles, even with all RSUs compromised. Moreover, it achieves strong nonrepudiation that any biological anonym driver could be conditionally traced, even if he is not the only driver of the vehicle. Extensive simulations reveal that 2FLIP is feasible and has an outstanding performance of nearly 0-ms network delay and 0% packet-loss ratio, which are particularly appropriate for real-time emergency reporting applications.**

***Index Terms*—Conditional traceability, privacy, strong nonrepudiation, two-factor authentication, vehicular ad-hoc network (VANET).**

I. Introduction

THE VEHICULAR ad-hoc network (VANET) has been subject to extensive research efforts from government, academia, and industry in recent decades. In a VANET, every vehicle is equipped with an onboard unit (OBU), through which it could communicate wirelessly with other vehicles and roadside units (RSUs) over one or more hops. Thus, a large-scale wireless network could be constructed, which utilizes dedicated short-range communications (DSRC) [2] to realize high-speed reliable data exchange of vehicle-to-vehicle (V2V) and vehicle-to-RSU (V2R), simultaneously achieving features of mobile ad hoc and communicatively opportunistic. The fabulous characteristics of the VANET are significant to traffic management and roadside safety. In addition, V2V aims at transmitting basic safety information between vehicles to facilitate warnings to drivers concerning impending crashes [3].

Two V2V applications are already available in production vehicles using vehicle-resident sensors: forward collision warning (FCW) and blind spot warning (BSW), and we focus on FCW in this paper. Compared with other safety technologies, safety applications for V2V are cooperative, which is a new paradigm in contrast to standalone sensor-based vehicle systems. Vehicles must send, receive, and analyze data in real time. This cooperative exchange of data about potential threats and hazards forms the basis of alerts and warnings to drivers to support their decisions and actions to avert impending incidents. However, a cooperative system can only work when participants in the system are able to trust the alerts and warnings issued by V2V devices working with messages from other V2V devices. In pervasive computing applications similar to this, security and privacy are two important and contradictory objectives, and users like to enjoy the services when the privacy is preserved in computing pervasive environments. Thus, we must solve the relevant security and privacy challenges [4].

Security requirements of a VANET could be divided into two types: a basic type due to the inheritance from a mobile ad-hoc network (MANET) and a special type concerning vehicular communications. Traditional security threats in wireless communication, such as eavesdropping, forgery, and modification, could be easily taken advantage of in VANETs. This incurs the basic security goals, such as resilience to forgery or modification of messages and nonrepudiation. Particularly for vehicular communication, the VANET system must collect and transmit only "anonymous" data from mobile users for mandatory

Manuscript received March 24, 2014; revised September 1, 2014; accepted February 4, 2015. Date of publication February 10, 2015; date of current version February 9, 2016. This work was supported in part by the National Basic Research Program of China (973 Program) under Grant 2012CB315804; by the National Natural Science Foundation of China (NSFC) under Grant 61173132, Grant 61402446, and Grant 61173133; and by the Special IOT Program of China's National Development and Reform Commission. The review of this paper was coordinated by Prof. D. H. C. Du.

F. Wang, Y. Xu, H. Zhang, and Y. Zhang are with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: wangfei@ict.ac.cn; xyj@ict.ac.cn; hwzhang@ict.ac.cn; zhmj@ict.ac.cn).

L. Zhu is with the School of Computer Science, Beijing Institute of Technology, Beijing 100081, China (e-mail: liehuangz@bit.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2015.2402166

applications and keep such data "anonymous" until securely destroyed. [3] This requires the VANET system's ability of privacy preserving, which means preserving private information related to individual vehicle (e.g., driver's name, license plate, speed, position, marker, model and vehicle identification number, trajectory). In addition, to prevent an adversary from linking multiple anonymous messages and tracking a vehicle, unlinkability is also needed (implied by level 3 privacy in [11]). If such properties are not guaranteed, malicious vehicles and adversaries can easily track some designated vehicle and acquire the driver's daily schedule to commit a theft, a robbery, or other crime. However, the privacy preservation here should be conditional to allow authorized parties to acquire evidence from the VANET. Thus, real identities of message senders (vehicle and driver) could be revealed under some disputable circumstances such as a crime/car accident scene investigation. We call such a requirement as nonrepudiation and strong nonrepudiation (trace vehicle and driver simultaneously). In summary, it is indispensable to develop an elaborate and carefully designed scheme to achieve security and conditional privacy in wide utilization of VANETs.

Apart from the security requirements, the performance requirement of security scheme is critical in VANETs. The two main reasons are high-speed mobility of vehicles and real-time data analysis. For example, a vehicle's speed could be measured in miles per hour, which makes the communication time between two vehicles extremely short. Both vehicles must send, receive, and analyze data in real time. This is hard when we also need to keep security and privacy, particularly in high traffic density. For example, DSRC requires that every vehicle sends a message every 100–300 ms, with a communication range of 300 m; when the traffic load is high as 100–1000 vehicles in the communication range, it catches a vehicle in the predicament of buffering and verifying 1000–10 000 messages every second. This would dramatically challenge the storing and computing power, causing potential disruption of service. To sum up, the security scheme must be lightweight enough.

Among previous studies [1], [5], [7], [15], [20], [21], one of the most public recognized idea to ensure the security of VANETs and privacy of vehicles is privacy-preserving authentication (PPA). Until now, there already suggest a large quantity of PPA schemes for VANETs, some of which are based on public key infrastructure (PKI) and are employing traditional digital signature techniques to authenticate messages. Such schemes have some downsides: 1) vulnerable availability due to effortless denial-of-service (DoS) attack and 2) collapse of scheme caused by high packet-loss ratio. The utilization of digital signature and corresponding certificate significantly adds the size of an actual on-the-fly message, leading to a heavy burden on wireless communicating bandwidth. This might cause some important messages in a life-or-death VANET application being dropped. Consequently, cautiously designed schemes crashed, making vehicles out of VANET services 3) have considerable cost for a certificate authority (CA) performing certificate updating and revoking. The VANET is characteristic of across-country-level widespread, which causes a significant consumption for CA to update all vehicles in the annual inspection. Moreover, according to conditional privacy concern, revocation of victim vehicles also costs a lot after privacy leaking accidents. To overcome the aforementioned drawbacks, some researchers proposed schemes trying to hybrid digital signature, hash function, and authentication code, which, however, could not solve the problems in essence.

The main reason of vulnerability in previous PPA schemes is the centralized architecture of CA. Honestly, transferring all authentication functions to distributed VANET nodes is hard. However, we are convinced that decentralization of some PPA functions is practical and beneficial to the construction of a strong PPA scheme. How can it be implemented? 2FA utilizes any two of the three factors, which are "something you know" (namely password), "something you have" (e.g., credit card, SMS phone, access token), and "something you are" (e.g., constant or at least stubborn feature of user, such as digital certificate or biometric identifier) to enhance the authentication process of users. Currently, 2FA has been widely used in world banking industry, a variety of web-based systems, and different kinds of wireless networks. The other core element to implement the decentralization is a portable telematics device (e.g., OBD-II support device with GPRS/3G/4G and GPS) because it could act like one factor (something you have) in 2FA and a distributed trusted security agent. Moreover, the portable telematics device should be equipped with a tamper-proof module (e.g., the IBM 4758 cryptographic coprocessor, which supports tamper-resistant packaging and is mature in industry and on the market), because implementing security services in vehicular networks requires vehicles to store sensitive data such as cryptographic keys (secret keys, private keys), event logs, etc. [1].

In this paper, we proposed a Two-Factor LIghtweight Privacy-preserving (2FLIP) authentication scheme for VANETs, which introduces the idea of a two-factor authentication technique to VANETs mainly by utilizing message-authentication-code (MAC) and hash operations for improving the security and privacy of VANETs. In the proposed scheme, each vehicle would be bonded to a telematics device, which would be utilized along with biometric technology [6] (e.g., face, fingerprint, iris, etc.) equipped on this vehicle, to verify the identities of multiple drivers and to provide evidences to trace each driver. Resilience to biometrics is not considered in this paper. Moreover, a tamper-proof device (TPD) is embedded in an OBU to store the system key and to sign/verify messages. To secure communications of V2V and V2R, 2FLIP only requires several extremely lightweight one-way hash operations and a MAC generation operation, for message signing, and a hash function along with one fast MAC regeneration, for verification. A digital signature verification process is only launched when a vehicle needs system key updating, which would not affect the performance. As far as we are concerned, 2FLIP is the first authentication scheme that achieves both strong privacy preservation and DoS resilience for secure VANET communication without employing a symmetric or asymmetric key mechanism; additionally, it is also the first authentication scheme trying to authenticate multiple users of one single vehicle, which conditionally traces each one of them in postevent investigation.

The advantages of our proposed 2FLIP scheme are as follows.

1) *Strong privacy preservation*
   2FLIP is able to guarantee level 3 privacy: authentication, anonymity, and unlinkability. Moreover, responsibilities of RSUs are purposefully weakened, which leads to strong privacy, such that, even if all RSUs are compromised, malevolent parties still could not pry into the real identities of vehicles.
2) *Strong nonrepudiation*
   2FLIP provides the basic nonrepudiation that the vehicle could not deny the message from itself. Moreover, considering multiple drivers of one vehicle could also not deny himself from sending the message. A driver has to first hold the telematics device and then offers his password (transformed from some biometrics, e.g., a fingerprint, or an iris scan) to start the vehicle. The evidences generated from the password are transmitted to a CA after some proper time interval, which are used to trace each driver conditionally, hence providing strong nonrepudiation.
3) *Secure system key update*
   Once the system key is leaked, 2FLIP provides a mechanism to restore the whole system by updating the system key at a low cost, which is essential for a practical system.
4) *Secure offline password update*
   Biological password embedded in a telematics device could be updated without connection to RSUs or CA, therefore providing support to flexible use right transfer.
5) *Extremely lightweight and efficient*
   2FLIP employs only hash operations coupled with MAC generation to accomplish the signing of messages and a fast MAC regeneration for verification, subsequently achieving a significant reduction of nearly 102–103 times in computational consumption compared with subsisting schemes. This makes 2FLIP DoS-resilient compared with concurrent schemes, even in large-scale VANET with large vehicle density.
6) *Low certificate management overhead, communication cost, and network delay*
   In 2FLIP, a dynamic pseudoidentity and a short MAC are carried within a message packet, rather than digital signature and certificate. On one hand, all certificate revocation list (CRL)-related overhead is eliminated whether it is responsible for by CA or vehicles. On the other hand, in the comparison with other current schemes, our proposed 2FLIP achieves a decrease of 55.24%–77.52% in communication costs and a considerably lower network delay.

Organization of this paper is presented as follows: Section II presents the related work about privacy authentication in VANETs. In Section III, the system model is defined. Then, Section IV gives an overview of 2FLIP and then full details of it. In Section V, we analyze the security of the scheme using the symbolic approach. Section VI evaluates the performance of 2FLIP through standalone experiments and network simulations. Section VII concludes the paper and look into the future work.

II. Related Work

Numerous schemes have been proposed to improve the security and conditional privacy preservation in VANETs. They could be classified into three categories: 1) schemes based on pseudonymous certificate; 2) schemes based on group signature; 3) hybrid schemes that combine the pseudonymous authentication and group signature.

1) *Pseudonymous-certificate-based schemes*
   Pseudonymous-authentication-based schemes first link many pairs of private key and pseudonymous certificate to a pseudoidentity. Afterward, a source vehicle could utilize its private key to sign messages, and all receivers could authenticate the messages by the corresponding pseudonymous certificate. Therefore, the real identity of the source vehicle is preserved in V2V communications. Raya and Hubaux [1] proposed a baseline pseudonymous authentication scheme (BP), which predistributes large quantities of pairs of anonymous private keys and matching pseudonymous certificates (e.g., 438 000 certificates in [1]) to every vehicle. Messages that are broadcasted in a short time interval (e.g., 1 min in [1]) are signed with a randomly chosen private key and then authenticated by the receiver vehicles with the corresponding pseudonymous certificate attached in broadcasted packets. The real identity of the sender can never be revealed by any vehicle or infrastructure because pseudonymous certificates are generated based on pseudoidentity. However, the conditional privacy could be achieved by CAs, which keep the matching between a vehicle's list of pseudonymous certificates and its real identity.
2) *Group-signature-based schemes*
   The cord idea of group-based schemes is that group members are hidden in a group, with real identity covered and privacy protected. In [15], Lin *et al.* suggested a privacy-preserving authentication scheme based on group signature [16], [17] and identity (ID)-based signature [18] (GSIS). Group signature is used to anonymously sign messages with private key by senders and verified with the group public key by receivers, while identities of senders can only be recovered by authorities. ID-based signature is applied by RSUs to digitally sign each message launched by RSUs to ensure its authority, where the signature overhead could be greatly reduced. CRL size of group signature is linear with the number of revoked vehicles, but the checking operation involves two paring calculations, which would take about 104 times computation cost than a string comparison [19]. In [20], Zhang *et al.* employ each RSU to maintain and manage an on-the-fly group within its communication range. Vehicles entering the group can anonymously broadcast V2V messages, which can be instantly verified by the vehicles in the same group (and neighbor groups). Due to numerous RSUs sharing the load to maintain the system, performance does not significantly degrade when more vehicles join the VANET. However, this scheme needs RSUs to be pervasive; otherwise, the scheme is ineffective. In [21], Sampigethaya *et al.* proposed a scheme that dynamically

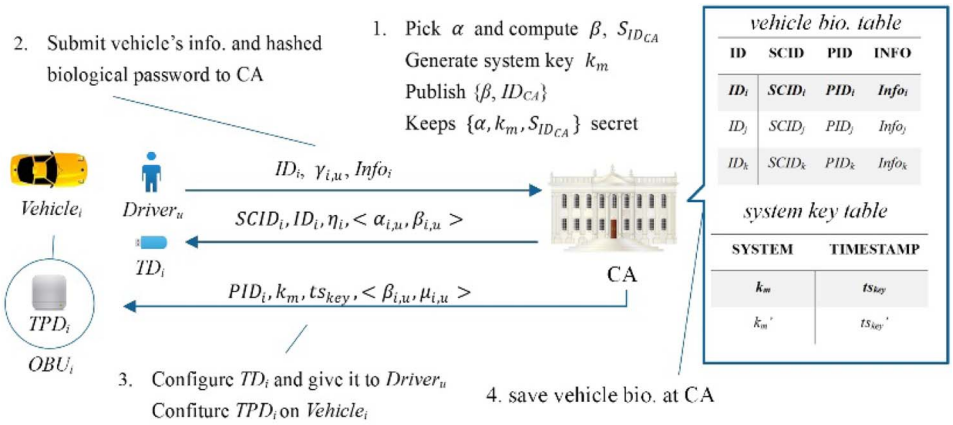


Fig. 1. Initialization phase of 2FLIP.

forms a group, and each group has a group key and a group leader. For group members in the same group communication, the group key is used for signing and authenticating messages; for group-to-group or group-to-infrastructure communication, the group leader acts as a proxy to send or request data instead of group members. The idea of group navigation of vehicles provides nature anonymity, but it may cost a lot of group leader's communication energy and computation resource and makes the group leader become the bottleneck of the system.

3) *Hybrid schemes*

Hybrid schemes combine pseudonymous authentication protocol, digital signature, MAC, and other authentication technologies to make a tradeoff between computation efficiency, CRL size, bandwidth consumption, verification delay, and so on. Calandriello *et al.* [19] proposed a hybrid scheme by combining a pseudonym scheme with group signature. Each vehicle V is equipped with a group signing key gskv and a group public key pgkCA. A vehicle can issue a "self-certify" certificate for itself by gskv and then signing its message using private key corresponding to the "self-certify" certificate. In such a way, the average overhead of message authentication can be reduced, but the expensive group signature CRL checking still remains a problem. Studer *et al.* [23] introduced a scheme based on the Elliptic Curve Digital Signature Algorithm (ECDSA) and a modified version of the Timed Efficient Stream Loss-Tolerant Authentication (TESLA++) [24]. ECDSA signatures provide fast authentication and nonrepudiation, and TESLA++ provides data integrity. The scheme is flexible, extensible, and efficient, but it does not provide privacy preservation and conditional traceability. In [25], Lin *et al.* suggested a similar scheme with [23]. Lin's scheme utilizes the pseudonymous authentication instead of direct using the ECDSA for nonrepudiation. Hence, Lin's scheme offers privacy preservation and conditional traceability. However, the TESLA is directly used in Lin's scheme, which makes the scheme vulnerable to memory DoS attack and increased verification delay.

As discussed earlier, the aforementioned schemes have different flaws. As for schemes based on pseudonymous certificate, the high overhead of certificate management on CA and vehicles could easily disrupt the service of VANETs. As for schemes based on group signature, computational cost for signing or verification is also unfit for VANETs. Hybrid schemes try to achieve the tradeoff between some schemes but are still not efficient enough. However, to achieve the goal of privacy preservation, high overhead of signing, verification, or certificate management by PKI-centered architecture is inevitable. Moreover, all these schemes rely heavily on RSUs, which might not be pervasive in real environments.

III. System Model

A. Network Model

We consider a typical VANET scenario. It consists of a top CA, some stationary RSUs, and a large number of vehicles equipped with OBUs running on the road, as shown in Fig. 1.

CA is a centered trusted authority, which is fully trusted by others. It has nearly unlimited computation and storage resources and is able to accomplish tasks such as 1) RSU and vehicle registration, 2) vehicle information and system key management, 3) message nonrepudiation verification and conditional traceability for both vehicles and biological drivers. The RSU is infrastructure deployed on the roadside, which is able to communicate with a CA directly usually through wired channel. It has large storage and powerful communication capability of 1–3 km. RSU is responsible for message forwarding and distributed RSU-aided key updating.

Every vehicle is equipped with an OBU, in which there is a TPD. The OBU is used to communicate with each other by sending (usually broadcasting) messages containing local traffic information, traffic light information, and emergency warning. The TPD is used to store cryptographic materials and process cryptographic operations. A TPD is similar to a trusted device and is extremely hard to hack into; thus, it is secure against any compromising attempt in relevant circumstance. Moreover, server failure of hacking try would trigger the self-destruct of the TPD. As for every vehicle, there could be more than one biological driver. However, only one portable telematics device is allowed for a single vehicle, which could be used to assist in driver identity verification and use right transfer. We also assume time synchronization for all OBUs on vehicles.

B. Adversary Model

An adversary has terrific communication abilities. Through powerful receivers, it is able to control the whole communication channel, monitor all the on-the-fly data through these channel, and tamper the message. Moreover, they could deliberately drop some packets, delay the transmission of them, and replace the original messages. Plus, infrastructure and devices are easily becoming the targets of adversaries. To exemplify it, small part of RSUs or vehicles could be compromised, and a telematics device might be stolen. DoS is another possible threat, which is usually performed by channel jamming and aggressive injection of dummy messages. Moreover, high-density traffic could also be the reason for DoS. Notwithstanding adversary is powerful as previously discussed, one hypothesis that we should be aware of here is that cryptographic materials are kept safe in TPDs and would never be achieved.

An adversary intends to induce the legitimate vehicles to accept false or harmful messages without being detected, thus abusing the VANET to maximize its gains (e.g., cheating neighboring vehicles to make a clear path to greedy driver's destination regardless of the cost to the system, snooping legitimate users' privacy to commit a crime). Furthermore, in order to take no responsibility for his injurious behaviors, an adversary would try to impersonate as other drivers through a stolen telematics device. Moreover, a malicious adversary would have deliberately generated large amounts of legitimate or invalid messages in a relatively short time interval in an attempt to disrupt the service of the VANET and to create disorder.

C. Design Goals

In order to make 2FLIP strong and robust in practical use, we set up series of security design goals and performance design goals.

First, basic security goals for wireless communication include resilience to forgery or modification of message and nonrepudiation.

1) *Resilience to forgery or modification of message*
   Every on-the-fly message should be authenticated to ensure that its source is legitimate and its payload is unaltered. Any forged or modified messages shall be detected and rejected by vehicle as soon as possible.
2) *Nonrepudiation*
   This goal includes three meanings, which are 1) a vehicle could not claim to be another vehicle, 2) a vehicle could not cheat about their position and related parameters, 3) a vehicle could not deny the actions and the time of generating and sending messages.

   Second, special goals concerning V2V communications include identity privacy preserving, strong privacy preservation, unlinkability, and conditional traceability.
3) *Identity privacy preserving*
   Privacy leaking is the leaking of binding between real identities and valuable information generated in the usage of vehicles. However, a traditional digital signature could not prevent identity information from leaking [26] because of the broadcasting nature in vehicular communication. Thus, to preserve privacy, identity privacy has to be guaranteed during V2V and V2R communication.
4) *Strong privacy preservation*
   Even if all RSUs are compromised, the adversary cannot obtain vehicles' real identities and privacy information.
5) *Unlinkability*
   Pseudoidentity is used as a mask to cover the real identity and provide anonymity in some schemes, but unlinkability is not provided. Unlinkability means that adversaries could never find common properties in multiple messages and then link them to one particular vehicle and trace its location. Namely, the location privacy violation [27] problem can never be incurred under unlinkability. In [11], the author defines three levels of user privacy. In this paper, we aim to achieve the level 3 privacy: authentication, anonymity, and unlinkability.
6) *Conditional traceability*
   With V2V and V2R communications being anonymous and unlinkable, a CA could still verify nonrepudiation and strong nonrepudiation of a message, to ensure that no vehicles or drivers can deny the message generated by itself, and retrieve a vehicle's real identity when the message is in dispute.

Third, goals to make the scheme complete for a secure system and to increase flexibility include *secure system key update, secure password update, and strong nonrepudiation*.

The goal of *secure system key update* aims to provide a mechanism to restore the system quickly once the system key is leaked; maybe it is not important as the aforementioned goals, but it is necessary for a complete secure system. As for *strong nonrepudiation* and *secure offline password update*, they are set up, due to utilization of biometric factor, and are also necessary to keep the scheme complete and flexible. The former is constructed on the accomplishment of nonrepudiation, which guarantees that a driver could never deny the action and the time interval of driving a vehicle. The latter one, i.e., secure *offline password update*, comes from the scenarios that many drivers share a car (e.g., many taxi drivers share a taxi or a driver just lends his car to his friend). However, one vehicle is equipped with at most one telematics device because of cost, thus allowing a telematics device to update the embedded biological information, namely, password in the proposed scheme, is essential, i.e., stable connection to RSUs or CA is hardly guaranteed in real environments. Therefore, flexible offline password updating is necessary.

Performance goal is *resilience to DoS*, which implies that authentication of V2V or V2R messages should be lightweight enough (in both storage and computation) to handle DoS attack caused by high transportation density or malicious parties.

IV. Proposed Two-Factor Lightweight Privacy Scheme

2FLIP employs mainly two core methods to achieve the design goals presented in the previous chapter: CA decentralization and the biological-password-based 2FA. The notations in the paper are shown in Table I.

TABLE I
SECURITY NOTATIONS AND DESCRIPTIONS

Notations	Descriptions
CA	certificate authority
$Vehiclei$	The ith vehicle
TPD_i	temper proof device of $Vehicle_i$
TD_i	portable telematics device of $Vehicle_i$
$\boldsymbol{G}$	a cyclic additive group
V	a cyclic multiplicative group
k_m	system key
ts	current timestamp
ts_{key}	timestamp of current system key being updated
m	payload of a message
ID_i	real identity of $Vehicle_i$
u	biological driver of $Vehicle_i$
$pw_{i,u}$	biological password of driver u of $Vehicle_i$
$< \alpha_{i,u}, \beta_{i,u} >$	energy density
$< \beta_{i,u}, \mu_{i,u} >$	biological password keeper for driver u of $Vehicle_i$
ID_{CA}	identity of CA
$S_{ID_{CA}}$	identity secret key of CA
$SCID_i$	identity of TD_i
PID_i	initial pesudo identity of $Vehicle_i$
$PID_{i,ts}$	dynamic pseudo identity of $Vehicle_i$ at ts
$Info_i$	vehicle information of $Vehicle_i$
$UT_{i,ts}$	update token for $Vehicle_i$ at time ts
$h(.)$	hash function h: $\{0,1\}^*\times V\rightarrow \mathbf{Z}_q^*$, $\mathbf{Z}_q^* = \{x \in \{1, \ldots, q\text{-}1\} \mid \gcd(x, q) = 1\}$
$h_k^1(.)$	hash function h_k^1: $\{0,1\}^*\rightarrow\{0,1\}^n$
$H(.)$	hash function H: $\{0,1\}^*\rightarrow\mathbf{G}^*$, $\mathbf{G}^* = \mathbf{G}\backslash\{0\}$ [31]
$Enc_k(.)$	encryption function using k as key, such as AES [28]
$Dec_k(.)$	decryption function using k as key, such as AES [28]
$mac_k(.)$	MAC computation function using k as a key, such as HMAC [29]
$mac_{m,ts}$	MAC of message m at ts
$Sign_k(.)$	identity-based message signing function [30]
sg	output of $Sign_k(.)$
$Verify_{id}(.)$	identity-based message verification function [30]
$\|\|$	message concatenation operation
!=	if equal, returns false, otherwise returns true

To reduce the CA's workload and communication burden, CA's functions are decentralized to the local security center, which consists of TD_i and TPD_i. Following is how the local security center works. In initialization phase, all vehicles need to register themselves to CA. In this phase, TPD_i and TD_i are cryptographically configured by CA. In the login/authentication stage. Before a driver needs to start his vehicle, he needs to pass the driver verification process first. After that, TD_i is free to generate the instant access token of TPD_i and use it to log on TPD_i. If the logon succeeds, TPD_i could be used to generate MAC with the system key and access token of TPD_i. When the vehicle has new status information, TD_i would redo the TPD logon. Then, TPD_i forms a packet with three parts: message payload containing new status information, MAC, and dynamic pseudoidentity. Then, the packet is broadcasted to its neighbors. When a nearby vehicle receives the message, all it needs to do is to perform one extremely lightweight hash operation and a MAC regeneration operation to do the message authentication. Obviously TD_i and TPD_i work together as CA agents to accomplish the authentication process, while CA has no work load in V2V communication.

In the initialization stage, the driver needs to submit not only his vehicle information but also his hashed biological password. Then, in the configuration process, CA writes a biological password verifier $\langle \alpha_{i,u}, \beta_{i,u} \rangle$ to TD_i and a biological password keeper $\langle \beta_{i,u}, \mu_{i,u} \rangle$ to TPD_i. $\langle \alpha_{i,u}, \beta_{i,u} \rangle$ is utilized to verify pw_i, by TD_i in the login/authentication phase, after driver u of Vehicle_i plugs TD_i into the OBD interface and inputs $\text{pw}_{i,u}$. The latter is used by TPD_i to keep evidences for identifying the instant biological anonym driver in postevent conditional tracing. The design of $\langle \beta_{i,u}, \mu_{i,u} \rangle$ provides a essential mechanism to trace the instant biological anonym driver, when there exists many drivers for a vehicle, because the broadcasted message in VANET application contains dynamic pseudoidentity of the vehicle, which could be used to conditionally trace the corresponding vehicle only. Such a mechanism includes a $\beta-$table. It keeps a $\langle \beta_{i,u}, \text{ts}_u \rangle$ for every single biological password update, which could be then used to trace one single biological anonym driver when CA locates Vehicle_i. Moreover, through biological password, the efficiency and stability of the authentication process could also be improved dramatically.

A. System Initialization and Entity Registration

Let $\mathbf{G}$ be a cyclic additive group of order q, $P \in \mathbf{G}$ a generator of $\mathbf{G}$, and let e: $\mathbf{G} \times \mathbf{G} \rightarrow V$ be a bilinear map, which satisfies the following conditions [30]: *bilinear*, i.e., $e(x_1 + x_2, y) = e(x_1, y)e(x_2, y)$ and $e(x, y_1 + y_2) = e(x, y_1 + y_2)$; *nondegenerate*, i.e., there exists $x \in \mathbf{G}$ and $y \in \mathbf{G}$ such that $e(x, y) \neq 1$.

Then, the CA initiates system parameters as follows.

1) CA randomly picks integer $\alpha \in \text{Z}_q^*$ as system private key, and computes $\beta = \alpha P$ as system public key.
2) CA computes $S_{ID_{CA}} = \alpha H(\text{ID}_{CA})$ as its identity secret key and generates system key $k_m = \{k_m^1, k_m^2\}$, where $k_m^1 \in \{0,1\}^a$, a is the key length of $\text{Enc}_k(.)$; $k_m^2 \in \{0,1\}^b$, b is the key length of $h_k^1(.)$.
3) CA publishes $\{\beta, \text{ID}_{CA}\}$ and keeps α, k_m, $S_{ID_{CA}}$ secret.

As for vehicle and driver registration, we have the following.

1) For Vehicle_i, along with its biological driver, first it submits its real identity ID_i, $\gamma_{i,u} = h(\text{pw}_{i,u})$, and Info_i (e.g., engine serial number, date of manufacture, and vehicle owner) to the CA through secure channels (e.g., drive to CA to submit information personally).
2) CA checks the correctness of these information (usually with assistance of the national vehicle management department). If the information is valid, CA randomly picks $\text{PID}_i \in \mathbf{Z}_q^*$ as initial pseudoidentity of Vehicle_i, and SCID_i for TD_i.
3) CA computes the following to acquire the biological password verifier and the biological password keeper:

$$\eta_i = h(\text{ID}_i||\text{SCID}_i||\text{PID}_i) \oplus h(\text{SCID}_i||k_m)$$
$$\mu_i = h(\text{ID}_i||\gamma_{i,u}||\text{PID}_i) \oplus h(\text{SCID}_i||k_m)$$
$$\alpha_{i,u} = h(\gamma_{i,u} \oplus \text{PID}_i),\ \beta_{i,u} = \text{PID}_i \oplus h(\text{SCID}_i \oplus \gamma_{i,u}).$$

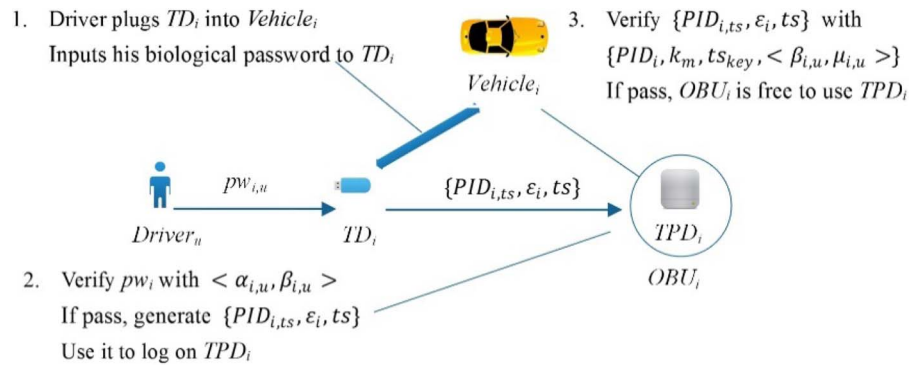


Fig. 2. Driver identity verification and TPD login of 2FLIP.

Here, $\langle \alpha_{i,u}, \beta_{i,u} \rangle$ is employed as a biological verifier to authenticate driver's identity and $\beta_{i,u}$ is used to protect the $\langle \beta_{i,u}, \mu_{i,u} \rangle$ and update the biological password locally.

4) Finally, CA saves $\langle \mathrm{ID}_i, \mathrm{SCID}_i, \mathrm{PID}_i, \mathrm{Info}_i \rangle$ of $\mathrm{Vehicle}_i$ to a user & bio table and writes $\{\mathrm{SCID}_i, \mathrm{ID}_i, \eta_i, \langle \alpha_{i,u}, \beta_{i,u} \rangle\}$ to a telematics device TD_i, which shall be distributed to the corresponding biological driver, and preloads $\{\mathrm{PID}_i, k_m, \mathrm{ts}_{key}, \langle \beta_{i,u}, \mu_{i,u} \rangle\}$ on TPD_i.

B. Driver Identity Verification and TPD Login

Before a driver joins the VANET, he needs to first pass the driver identity verification. After that, whenever the vehicle generates a new message and broadcasts it, the TPD login process should be launched instantly.

1) Driver identity verification: In this phase, the driver's identification would be verified through the cooperation of TD_i and TPD_i. Driver first plugs the TD_i into the $\mathrm{Vehicle}_i$ and inputs his biological identification information $\mathrm{pw}_{i,u}$ as password into it (maybe just a quick scan of his finger on TD_i). Then, the biological verifier $\langle \alpha_{i,u}, \beta_{i,u} \rangle$ could be used to verify the driver's identity: $\gamma^*_{i,u} = h(\mathrm{pw}_{i,u})$, $\mathrm{PID}_i = \beta_{i,u} \oplus h(\mathrm{SCID}_i || \gamma^*_{i,u})$, $\alpha^*_{i,u} = h(\gamma^*_{i,u} \oplus \mathrm{PID}_i)$. If the $\alpha^*_{i,u} ! = \alpha_{i,u}$ returns false, which means that the driver is legitimate, TD_i would keep PID_i until TD_i is unplugged.
2) TPD login: Plugged TD_i would first calculate the instant pseudoidentity like $\mathrm{PID}_{i,ts} = h(\mathrm{ID}_i || \mathrm{SCID}_i || \mathrm{PID}_i) \oplus h(\mathrm{PID}_i || \mathrm{ts})$, in which timestamp is embedded in to prevent possible replay attack, and the signature of pseudoidentity $\varepsilon_i = h(\eta_i || \mathrm{PID}_i || \mathrm{ts})$. Then, TD_i sends $\{\mathrm{PID}_{i,ts}, \varepsilon_i, \mathrm{ts}\}$ to TPD_i. TPD_i would verify the legitimacy of TD_i by calculating $\chi^* = \mathrm{PID}_{i,ts} \oplus h(\mathrm{PID}_i || \mathrm{ts})$ and $\varepsilon^*_i = h((\chi^* \oplus h(\mathrm{SCID}_i || k_m)) || \mathrm{PID}_i || ts)$. If $\varepsilon^*_i ! = \varepsilon_i$ returns false, which means that the TD_i plugged in is legitimate, then OBU is free to use TPD_i to do further action.

Fig. 2 shows the driver identity verification and the TPD login phase of 2FLIP.

C. Message Signing

When the vehicle generates a new message payload m, TD_i redoes the TPD login phase to facilitate the TPD with up-to-date dynamic pseudoidentity $\mathrm{PID}_{i,ts}$. If the TPD login is finished, TPD_i would calculate the message authentication value of the m like $\sigma_i = \mathrm{mac}_{k_m}(\mathrm{PID}_{i,ts} || h(m || k_m) || \mathrm{ts})$ and broadcasts $\{\mathrm{PID}_{i,ts}, \sigma_i \mathrm{ts}, m\}$ to nearby vehicles.

D. Message Verification

TPD_j calculates $\sigma^*_i = \mathrm{mac}_{k_m}(\mathrm{PID}_{i,ts} || h(m || k_m) || \mathrm{ts})$ to verify the legitimacy of the message after $\mathrm{Vehicle}_j$ receives a packet $\{\mathrm{PID}_{i,ts}, \sigma_i, \mathrm{ts}, m\}$ from $\mathrm{Vehicle}_i$. If $\sigma^*_i ! = \sigma_i$ returns false, $\mathrm{Vehicle}_j$ then accepts the message and employs the message for application use; otherwise, it rejects the message.

E. System Key Update

System key k_m is the cornerstone of the whole system and is protected by the TPD; thus, an adversary cannot take advantage of the TPD even if the vehicle is stolen. In order to further enhance the system security, we introduce a system key updating strategy to update k_m periodically, as shown in Fig. 3. Update of the system key ought to be carried out by the national vehicle management department on vehicle annual inspection and implemented by CA and distributed RSUs.

1) CA first generates the new system key like $k'_m = \mathrm{genkey}()$ and encrypts it like $c = \mathrm{Enc}_{k_m}(\mathrm{ts}'_{key} || \mathrm{ID}_{CA} || k'_m)$. Then, it signs the encrypted message to get the signature $\mathrm{sg} = \mathrm{Sign}_{S_{ID_{CA}}}(c)$. Afterward, CA broadcasts the message $\{c, \mathrm{sg}\}$ to the whole network with the help of RSUs.
2) After a vehicle receives a key update message $\{c, \mathrm{sg}\}$, the corresponding TPD first decrypts c and gets payload $\langle \mathrm{ts}'_{key} || \mathrm{ID}^*_{CA} || k'_m \rangle$ and then checks the ts'_{key} to prevent the replay attack, compares ID^*_{CA} with the published ID_{CA} to authenticate the origin of this message, and verifies the signature sg as $1 ! = \mathrm{Verify}_{S_{ID_{CA}}}(c, \mathrm{sg})$, if it returns false, which means that the message is valid.
3) The system key update process needs the permission of a legitimate driver to modify the security materials in TPD_i, namely, needs the telematics device to redo the TPD login process. Thus, after the aforementioned two steps, TPD_i would notify the TD_i and the driver of the system key update. The driver would check it with the announcement from the national vehicle management department. After TPD login is passed, on one hand, TPD_i updates k_m to k'_m and ts_{key} to ts'_{key}; on the other hand, it would update the information in TD_i related to k'_m.

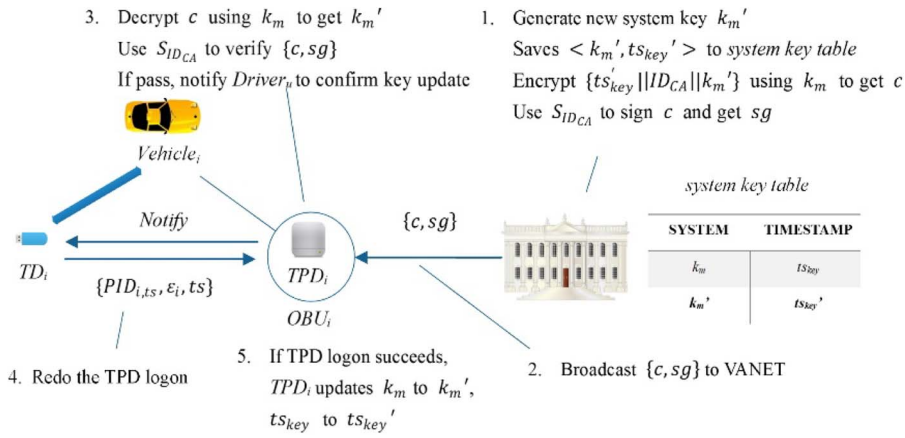


Fig. 3. System key update of 2FLIP.

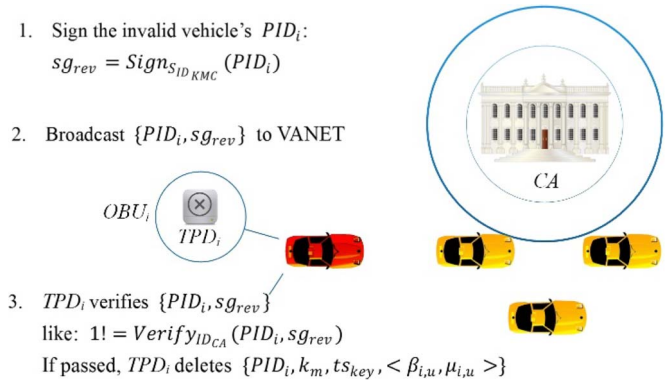


Fig. 4. Vehicle revocation of 2FLIP.

F. Vehicle Revocation

Once Vehicle_i is judged as invalid, CA would perform the revocation process of it to notify all other vehicles. It is direct and quick. CA broadcasts $\{\text{PID}_i, \text{sg}_{rev}\}$ to all vehicles, in which sg_{rev} is the signature of PID_i calculated by $\text{sg}_{rev} = \text{Sign}_{S_{ID_{CA}}}(\text{PID}_i)$. When Vehicle_i receives the revocation message, it would verify the source legitimacy of it. If legitimate, TPD_i deletes all the secret materials preloaded in the registration phase, including $\{\text{PID}_i, k_m, \text{ts}_{key}, \langle \beta_{i,u}, \mu_{i,u} \rangle\}$; thus, TPD_i is made illegal and is no longer able to generate legitimate messages. The phase is shown in Fig. 4.

G. Message Tracing

Although the anonymity and unlinkability are preserved, CA is able to trace the source vehicle and biological driver of each disputable message in after-event investigation, as shown in Fig. 5. Take one message $\{\text{PID}_{i,ts}, \sigma_i, \text{ts}, m\}$ for example, CA first selects a corresponding k_m by ts from the system key table and then verifies the validity of this message just like the TPD does, as shown earlier in this paper. If the message is legitimate, it could be used to trace the source of the message by searching all records of *vehicle bio. table* until one of them $\langle \text{ID}_i, \text{SCID}_i, \text{PID}_i, \text{Info}_i \rangle$ qualifies the equation $\text{PID}_{i,ts} == h(\text{ID}_i||\text{SCID}_i||\text{PID}_i) \oplus h(\text{PID}_i||\text{ts})$. After the record is found, Info_i could be used by the national transportation department to trace the source vehicle, while the disputable message is broadcasted. Once the Vehicle_i is found, further investigation is carried out by searching the $\beta -$ table and doing the comparison of $\beta_{i,u} == \text{PID}_i \oplus h(\text{SCID}_i||h(\text{pw}_{i,u^*}))$, in which pw_{i,u^*} is the biological information from all drivers of Vehicle_i. Thus, the biological driver could be traced.

H. Biological Password Update

In the proposed scheme, benign flexibility is provided that such biological password update could be implemented offline without any contact with CA or RSUs but only relying on the telematics device and TPD, as shown in Fig. 6.

The password update starts with a driver identity verification process, as shown in Section IV-B. After current driver u passes the identity verification, he is able to let the new driver u' input $\text{pw}_{i,u'}$. Then, TPD_i hashes $\text{pw}_{i,u'}$ to get $\gamma_{i,u'}$ and computes the update token $\text{UT}_{i,ts} = h(\text{ID}_i||\gamma_{i,u}||\text{PID}_i) \oplus h(\text{ID}_i||\gamma_{i,u'}||\text{PID}_i) \oplus h(\text{PID}_i||\text{ts})$, which will be used later to update secret parameters stored in TPD_i (μ_i) and in telematics device $(\alpha_{i,u}, \beta_{i,u})$. Then, TD_i performs the TPD login process and sends $\text{UT}_{i,ts}$ to TPD_i. If the TPD login process is passed, TPD_i would update secret parameters $\psi^* = \text{UT}_{i,ts} \oplus h(\text{PID}_i||\text{ts})$, $\mu_{i,u'} = \mu_{i,u} \oplus \psi^*$, set $\mu_{i,u} = \mu_{i,u'}$.

TPD_i also needs to compute $\beta_{i,u'} = \text{PID}_i \oplus h(\text{SCID}_i||\gamma_{i,u})$, for two reasons: One is stored in the $\beta -$ table for tracing the biological driver in postevent investigation; the other is for TD_i updating parameters. After TD_i receives $\beta_{i,u'}$, it would compute $\alpha_{i,u'} = h(\gamma_{i,u'} \oplus \text{PID}_i)$ and then set $\alpha_{i,u} = \alpha_{i,u'}$, $\beta_{i,u} = \beta_{i,u'}$. Up to now, the password update process is completed, and the new driver u' is free to use the vehicle, even after the car is launched again.

There may be doubts on how to trace the new driver after the password is changed; as stated in Section IV-G, the CA could trace the biological driver through the broadcasted message and get the registered information Info_i. Here, we believed the assumption that, once a driver lends the vehicle to the other, he literally knows and believes this driver and, thus, could provide sufficient information to trace this new driver.

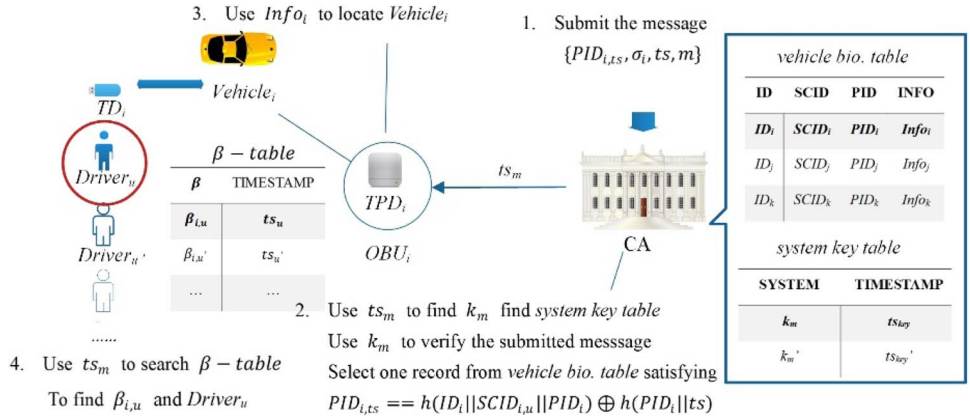


Fig. 5. Message tracing of 2FLIP.

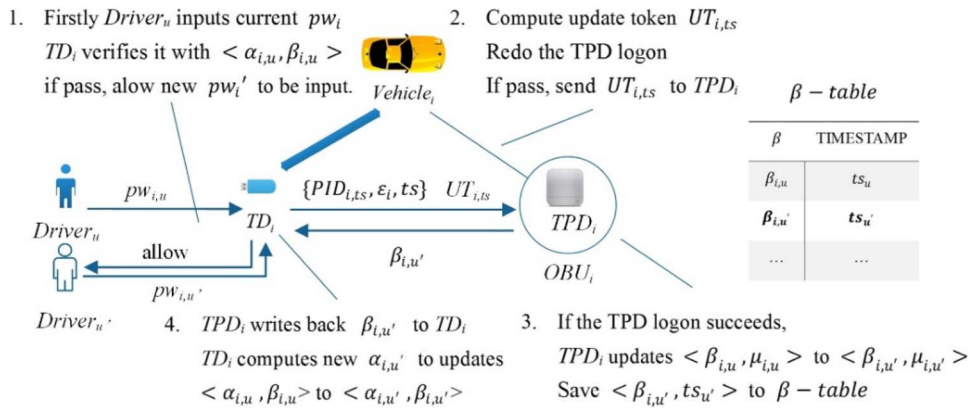


Fig. 6. Biological password update of 2FLIP revocation of 2FLIP.

V. Security Analysis

Here, we first give some preliminaries about the symbolic approach. Then, we implement core phases of 2FLIP using ProVerif and give an analysis of essential security properties. In the end, we compare the security properties of 2FLIP with BP, GSIS, VANET Authentication using Signatures and TESLA++ (VAST), and VAST$*$.

A. Preliminaries

In the proposed scheme, benign flexibility is provided that such a biological password update could be implemented offline with no contact between CA or RSUs but only relying on the telematics device and TPD.

The computational approach and the symbolic approach are two major directions to analyze the cryptographic protocols in the last two decades. Each has its advantage and disadvantage. As for the computational approach, it is computationally sound because it applies computational complexity and probability theory to reduce the security of the protocols to some cryptographic hardness assumptions. However, the proof is hard to be realized through programming. Therefore, it is tedious and highly error prone for even moderately complex protocols [42]. In comparison, the symbolic approach is amenable to be automated because of its algebraic structure [38]. There are many automated tools for the symbolic approach. For example, ProVerif is a tool for applied spi calculus [39]. However, three concerns exist in the symbolic approach: 1) The computational soundness is unclear, 2) the number of participants must be fixed, 3) the time complexity increases exponentially along with the number of participants. Recently, Canetti *et al.* [40], [41] have proposed the universally composable symbolic analysis (UCSA) approach, which proved that the security is unrelated with the number of sessions in their approach. However, it could only be used to deal with two-party cryptographic protocols. In [42], the UCSA approach is extended to deal with arbitrary number of participants. In addition, according to [42, Th. 2], the symbolic approach implies the computational approach. Some important keywords of the pi calculus are as follows.

query ⟨query⟩: The declaration tells the system which properties we want to prove. We use two kinds of facts for this keyword such as:

query attacker: M means that the attacker may have M in some phase (M is not secret).

query ev : $f(x1, \ldots, \text{xn})$ ==> ev : $f'(x1, \ldots, \text{xn})$ is a non-injective agreement: It is true when, if the event $f(x1, \ldots, \text{xn})$ has been executed, then the event $f'(x1, \ldots, \text{xn})$ must have been executed (before the event $f(x1, \ldots, \text{xn})$).

choice [⟨term⟩, ⟨term⟩]: It reconstructs a trace until a program point, at which the process using the first argument of choice behaves differently from the process using the second argument of choice. If a trace is reconstructed, it means the attacker could

```
Starting query not attacker:(ID_i_50[!1 = v_8135],ID_j_41[!1 = v_8136],pw_iu_88!1 = v_8137],pw_ju_87[!1 = v_8138])
RESULT not attacker:(ID_i_50[!1 = v_8135],ID_j_41[!1 = v_8136],pw_iu_88[!1 = v_137],pw_ju_87[!1 = v_8138]) is true.

Starting query ev:endAuthV2V(x4_13303,x5_13304,x6_13305) ==> ev:beginAuthV2V(x4_13303,x5_13304,x6_13305)
RESULT ev:endAuthV2V(x4_13303,x5_13304,x6_13305) ==> ev:beginAuthV2V(x4_13303,x_13304,x6_13305) is true.

Starting query ev:endAuthTD(x1_26532,x2_26533,x3_26534) ==> ev:beginAuthTD(x1_2532,x2_26533,x3_26534)
RESULT ev:endAuthTD(x1_26532,x2_26533,x3_26534) ==> ev:beginAuthTD(x1_26532,x2_6533,x3_26534) is true.
```

Fig. 7. Results for the ProVerif program A.

```
Starting query not attacker: (ID_i_36[], pw_iu_71[], km_57[], km'_67[reg_info_i = v_566])
RESULT not attacker: (ID_i_36[], pw_iu_71[], km_57[], km'_67[reg_info_i = v_566]) is true.
```

Fig. 8. Results for the ProVerif program B.

```
Starting query not attacker: (ID_i_31[], pw_iu_65[], km_55[], pw_iu'_44[pw_iu' = v_1045, reg_td_i = v_1046,pw_iu = v_1047])
RESULT not attacker: (ID_i_31[], pw_iu_65[], km_55[], pw_iu'_44[pw_iu' = v_1045,reg_td_i = v_1046,pw_iu = v_1047]) is true.
```

Fig. 9. Results for the ProVerif Program C.

distinguish the first argument from the second one of choice.

$!\langle$process$\rangle$: It means the replication executes an unbounded number of copies of $\langle$process$\rangle$ in parallel : $\langle$process$\rangle|\langle$process$\rangle|\langle$process$\rangle|\ldots$.

B. Experiment and Analysis

Here, we applied other researchers' work [39]–[42] to analyze the security of the 2FLIP scheme. We realized core phases of system initialization, entity registration, driver identity verification, TPD login, message signing, and message verification as ProVerif Program A. Moreover, the system key update and biological password update phases were realized in ProVerif Program B and C respectively. The results for ProVerif Program A, B and C [43] are shown in Figs. 7–9. In addition, the corresponding analysis for the security properties are as follows.

1) *Resilience to Forgery or Modification of Message*: Traffic message is protected by MAC, and the system key updating message is signed by CA. Any forgery or modification will be detected. Results shown in Fig. 7 mean that, if "event endAuthV2V(PID_i_ts, sigma_i, ts)" has been executed, then "event beginAuthV2V(PID_i_ts, sigma_i, ts)" must have been executed. In other words, the adversary cannot forge or modify the message $\{\text{PID}_{i,ts}, \sigma_i, \text{ts}, m\}$ to let it be accepted by other vehicles.
2) *Nonrepudiation*: Every broadcasted message is integrated with dynamic pseudoidentity, which is generated by the corresponding TPD integrated with identity, pseudoidentity, smart card identity, and timestamp. Due to the former three elements, a vehicle can never deny the generating action of a message through its TPD. Due to the timestamp, it can never deny the generating time of a message. Thus, nonrepudiation is guaranteed.
3) *Identity Privacy Preserving*: Dynamic pseudoidentity $\text{PID}_{i,s}$ is utilized for V2V and V2R communication to preserve the real identity of a vehicle. A biological driver needs to input his biological information on the telematics device before launching the vehicle. Thus, even if the vehicle or the telematics device is stolen, the identity privacy is still preserved. As shown in Fig. 7, we queried the adversary ID_i and ID_j in the program, and the result is true, which means that the adversary could not get any information about ID_i and ID_j.
4) *Unlinkability*: Both MAC generation and message authentication could be accomplished without knowing the real identity of the sender; moreover, owing to the fact that dynamic pseudoidentity differs as time changes, an adversary can never launch replay attack nor link numerous messages to one vehicle. Thus, the proposed scheme achieves *level 3 privacy*: authentication, anonymity, and unlinkability. To test the anonymity, we modify the program in ProVerif Program A by adding the keyword "choice[PID_i_ts,r0]." The result is "RESULT Observational equivalence is true (bad not derivable)," which means that the adversary could not distinguish $\text{PID}_{i,ts}$ from a random number r0; thus, anonymity is preserved. To test the unlinkability, we add "!" before the processes, and the result is still true, which means no matter how many messages the adversary collects, it still could not get any information about vehicle's identity.
5) *Strong Privacy Preservation*: In 2FLIP, RSUs are only responsible for traffic message forwarding and distributed RSU-aided system key updating. Thus, even with all RSUs compromised, the adversary knows nothing more. Therefore, even if all RSUs are compromised, the adversary still cannot pry into vehicles' nor drivers' privacy.
6) *Conditional Traceability*: In postevent investigation, system key k_m and *vehicle bio. table* are the key to disclose the real identities of the vehicle and the biological driver of a disputable message. In ProVerif Program B, we queried the adversary value of k_m, and the results in Fig. 8 show that the adversary could not get any information about k_m; thus, the property of traceability is conditional. Compared with 2FLIP, the VAST scheme only guarantees unconditional traceability, which make it free to trace a particular vehicle and incurs location privacy violation.
7) *Strong Nonrepudiation*: Through the use of $\beta-$table, driving evidences are kept securely in the TPD; no one but the CA is able to look up and change the records of $\beta-$table. The core part of each record of driving evidences is parameter $\beta_{i,u}$, which is computed like $\beta_{i,u} = \text{PID}_i \oplus h(\text{SCID}_i||\gamma_{i,u})$, where $\gamma_{i,u}$ serves as abstract of driver's biological information and could be utilized to identify the biological driver in a driving interval of Vehicle_i.
8) *Secure System Key Update*: We queried the adversary k_m and k'_m in the ProVerif program in ProVerif Program B, and the result is true, as shown in Fig. 8, which means

TABLE II
COMPARISONS OF PROPERTIES BETWEEN SCHEMES

Schemes / Properties		BP	GSIS	VAST	VAST*	2FLIP
Integrity		√	√	√	√	√
Non-repudiation		√	√	√	×	√
Level 3 Privacy	Authentication	√	√	√	√	√
	Anonymity	√	√	×	×	√
	Unlinkability	×	√	×	×	√
Strong Privacy Preservation		×	×	×	×	√
Conditional Traceability		√	√	×	×	√
Strong Non-repudiation		×	×	×	×	√
Secure System Key Update		×	×	×	×	√
Secure Offline Password Update		×	×	×	×	√
Resist to DoS	Computation	×	×	√	√	√
	Memory	√	√	√	√	√

that the adversary could not get any information about old system key k_m or the updated system key k'_m.

9) *Secure Offline Password Update*: With CA's agent, the biological password update could be done offline, which in fact is the driving right transferring. We queried the adversary $\text{pw}_{i,u}$ and $\text{pe}'_{i,u}$ in the ProVerif program in ProVerif Program C, and the result is true, as shown in Fig. 9, which means that the adversary could not get any information about old password $\text{pw}_{i,u}$ or the new password $\text{pw}'_{i,u}$.

As shown in Table II, 2FLIP achieves all the issued security properties and is more practical than their schemes.

VI. PERFORMANCE EVALUATION

Here, we evaluate the performance of the proposed 2FLIP with BP, GSIS, VAST, and VAST$*$ schemes (VAST$*$ is the performance result when nonrepudiation is not necessary in VAST). Tate pairing [33] is adopted in our evaluation, where $\mathbf{G}$ is represented by 161 bits, and the order q is represented by 160 bits. Moreover, we utilize AES-128 as $\text{Enc}_k(.)$, hash-based MAC (HMAC) as $\text{mac}_k(.)$, and SHA-1 as $h_k^1(.)$. Let N_{crl} denote the number of CRL items, T_{mul} denote the time to compute one point multiplication, T_{par} denote the time to perform one pairing operation, T_{h} denote the time of one hash-function operation, T_{mac} denote the time of one MAC operation, and T_{enc} denote the time of one encryption operation. $T_{\text{mul}}, T_{\text{par}}, T_{\text{h}}, T_{\text{mac}}$, and T_{enc} dominate the computation performance of schemes; for simplicity, we only consider these operations for authentication overhead evaluation, certificate updating overhead evaluation, and vehicle revocation overhead. We run 100 times point multiplication, Tate pairing, SHA-1 hash function, AES-128 encryption on a machine equipped with an Intel Core 2 Duo CPU at 2.4 GHz, respectively, and the average operation times are 5.4 ms, 40.7 ms, 6 μs, 16.7 μs, and 40.7 μs, respectively. The following simulation adopts the measured processing time based on these data. The certificate validity period $\Delta T = 60$ s, and vehicles broadcast a message every 300 ms according to DSRC.

TABLE III
COMMUNICATION OVERHEAD FOR ONE MESSAGE

Schemes	BP	GSIS	VAST	VAST*	2FLIP
Overhead(Byte)	105	192	145	145	47

TABLE IV
MESSAGE SIGNING COST

Schemes	BP	GSIS	VAST	VAST*	2FLIP
Computation Cost	T_{mul}	$3T_{\text{par}}$ $+T_h$	$T_{\text{mul}}+$ T_{mac}	$T_{\text{mul}}+$ T_{mac}	$7T_{\text{h}}+$ T_{mac}

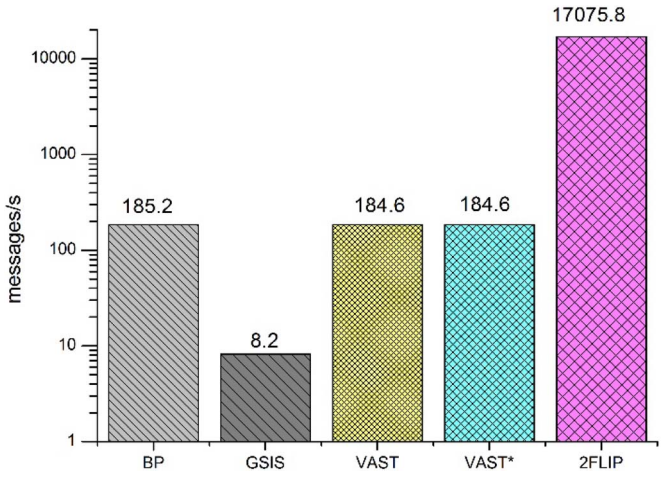


Fig. 10. Message signing speed.

A. Authentication Overhead

1) Communication: Table III shows the comparisons of communication overhead for one message. Communication overhead of one message contains the attached certificate and signature. In BP, the certificate is 63 bytes and signature is 42 bytes, which makes the total overhead of 105 bytes. While in GSIS, due to the fact that group signature utilizes group public key to verify messages, no certificate is needed, only 192 bytes of signature. Communication overhead of VAST and VAST$*$ is 145 bytes in total (63-byte certificate, 20-byte MAC, 42-byte signature, 16-byte symmetric key, and 4-byte index ID). No certificate is needed in 2FLIP; thus, the 47-byte communication overhead includes 20-byte MAC, 23-byte pseudoidentity, and a 4-byte timestamp.

Obviously, compared with BP, GSIS, VAST, and VAST$*$, 2FLIP causes the lowest bandwidth consumption. It significantly decreases the communication overhead by 55.24%–77.52%.

2) Message Signing: The message signing cost comparisons are shown in Table IV. In BP, message signing requires one point multiplication; hence, the cost is T_{mul}. According to our former experiment, BP can sign $1/0.0054 \approx 185.2$ messages every second. As for GSIS, every message signing requires three bilinear-pairing operations whose cost is T_{par} and one hash operation whose cost is T_{h}, thus incurring the largest signing cost. In VAST and VAST$*$, the signing needs one point multiplication and one message authentication, and the speed of message signing is nearly the same as BP. However, for 2FLIP, only lightweight hash operation and message authentication operation are needed. Fig. 10 illustrates the number of messages

TABLE V
COMMUNICATION OVERHEAD FOR ONE MESSAGE

Schemes	BP	GSIS	VAST	VAST*	2FLIP
CRL Checking	0	$2N_{\text{crl}}T_{\text{par}}$	0	0	--
Certificate Verification	$2T_{\text{mul}}$	0	$2T_{\text{mul}}^{*}$	0	--
Signature Verification	$2T_{\text{mul}}$	$5T_{\text{par}}+T_{\text{h}}$	$2T_{\text{mul}}^{*}+2T_{\text{mac}}$	$2T_{\text{mac}}$	$T_{\text{h}}+T_{\text{mac}}$
Total	$4T_{\text{mul}}$	$2N_{\text{crl}}T_{\text{par}}+5T_{\text{par}}+T_{\text{h}}$	$4T_{\text{mul}}^{*}+2T_{\text{mac}}$	$2T_{\text{mac}}$	$T_{\text{h}}+T_{\text{mac}}$

*Note: In VAST, certificate and digital signature verification is only performed when non-repudiation is necessary.

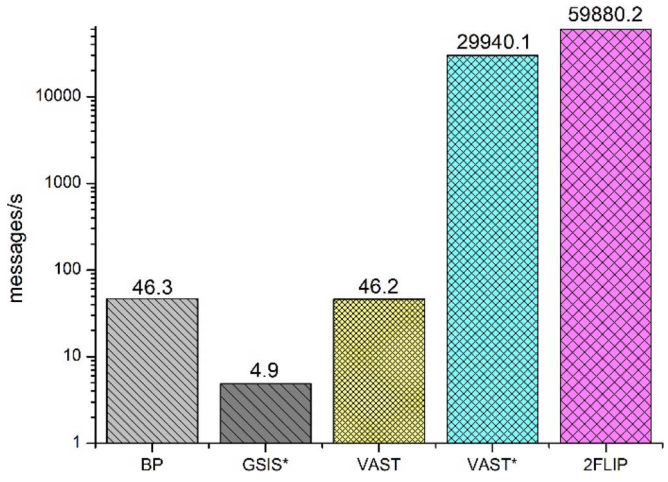


Fig. 11. Message signing speed.

that each scheme can sign per second. Obviously, 2FLIP is the most efficient scheme for message signing and can sign 17 075.8 messages per second.

3) Message Verification: For BP, GSIS, VAST, and VAST∗, message verification includes CRL checking, certificate verification, and signature verification. BP, VAST, and VAST∗ perform CRL checking through string comparison; computation cost of which could be ignored. GSIS needs two paring operations for each CRL item, which makes the total CRL checking cost $2N_{\text{crl}}T_{\text{par}}$. However, 2FLIP only needs one fast MAC operation to accomplish the message verification. Table V shows the comparisons of message verification cost. As concluded in Tables IV and V, we can observe that 2FLIP significantly reduces the computation cost by $10^2 \sim 10^3$ times compared with the other three typical schemes.

Fig. 11 illustrates the number of messages that each scheme can verify per second. Obviously, 2FLIP is the most efficient scheme for message verification. In a high-vehicle-density scenario, such as 1000 cars in a 300-m communication range, only 2FLIP and VAST∗ are able to work. However, VAST's precondition of global time synchronization increases message delay and unfeasibility, which is not suitable for the actual scenarios.

B. Certificate/Key Updating Overhead

According to [7], BP needs to update its 48 830 pseudocertificates once the preloaded ones are consumed. The communication overhead is 48 830 ∗ 63 bytes, and the computation cost for a vehicle is $2T_{\text{mul}}$. In GSIS, CA sends all the revocation list to every vehicle to update group key, the communication overhead is $41N_{\text{crl}}$ bytes; the computation cost for a vehicle is T_{par}. Certificate updating in VAST and VAST∗ is similar with BP, but it only needs to update one certificate at a time. Table VI shows the certificate/key updating overhead of different schemes.

TABLE VI
CERTIFICATE/KEY UPDATING OVERHEAD

Schemes	BP	GSIS	VAST	VAST*	2FLIP
Communication (Byte)	3076290	$41N_{\text{crl}}$	63	63	88
Computation cost(s)	$2T_{\text{mul}}$	T_{par}	T_{mul}	T_{mul}	$T_{\text{enc}}+2T_{\text{par}}+T_{\text{h}}$

TABLE VII
VEHICLE REVOCATION OVERHEAD

Schemes	BP	GSIS	VAST	VAST*	2FLIP
Communication (Byte)	512715 + 105	41 + 192	21 + 105	21+105	21 + 42
Computation cost (s)	$2T_{\text{mul}}$	$5T_{\text{par}}+T_{\text{h}}$	$2T_{\text{mul}}$	$2T_{\text{mul}}$	$2T_{\text{par}}$
CRL size (Byte)	512715 N_{crl}	$41N_{\text{crl}}$	$21N_{\text{crl}}$	$21N_{\text{crl}}$	0

GSIS, VAST, VAST∗, and 2FLIP ought to update the certificate/key as needed similar to an annual inspection, with assistance of the national vehicle management department. BP has to update pseudocertificates once the certificates are used up. Certificate/key updating overhead of GSIS grows linearly with N_{crl}; therefore, when N_{crl} is large, the overhead may be very high. Both VAST and 2FLIP have an extremely low certificate/key updating overhead, but VAST and VAST∗ do not provide identity privacy preservation and unlinkability.

C. Vehicle Revocation Overhead

Once a vehicle is determined invalid and needs revoking in BP, all the public keys of valid and invalid pseudocertificates from the vehicle shall be inserted into the CRL.

The average number of valid pseudocertificates a vehicle possesses is 48830/2. Hence, the communication overhead is $24415 * 21 + 105$ bytes. The 105-byte overhead is the attacked certificate and signature. While in GSIS, VAST, VAST∗, and 2FLIP, only one item needs to be inserted into the CRL. The corresponding overhead is shown in Table VII.

It can be seen that 2FLIP has the lowest communication overhead for revoking a vehicle. Furthermore, 2FLIP employs TPDs to revoke vehicles; then, vehicles in 2FLIP do not need to maintain a CRL to record the revoked vehicles. This benign property makes 2FLIP particularly suitable for large-scale VANETs.

D. Simulation

Here, we run simulations for BP, GSIS, VAST, VAST∗, and 2FLIP with an opportunistic networking environment (ONE [35]). Aiming at estimating real-world road systems properly,

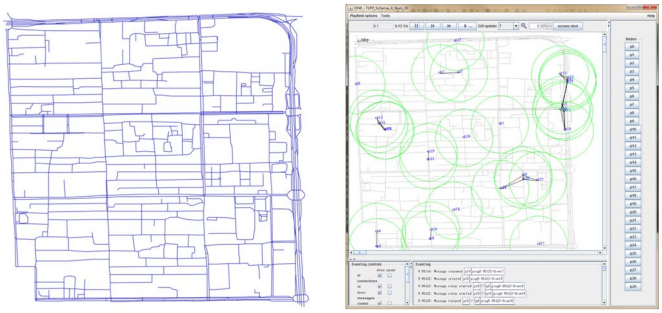


Fig. 12. City street scenario corresponding to a roughly square area of size $2250 \times 2250\ \mathrm{m}^2$.

TABLE VIII
SIMULATION CONFIGURATION

Parameter	Values
Communication Range	300m
Simulation Time	100s
Channel bandwidth	6Mbps
Wait time	0~5s
Buffer Size	1M bytes
Broadcast Interval	0.3s
Speed	[20km/h,100km/h]

we select a part from the real map of Beijing (northeast corner of area surrounded by the No.2 Ring Road of Beijing) and import it into ONE as a city street scenario. The adopted map and the user interface of ONE in this paper are presented as in Fig. 12.

All vehicles are distributed deliberately on the roads of the map at the beginning of each simulation. Each of them would choose one casual point separately on roads and move toward it following some kind of movement model, at a random speed generated from a range of ± 10 km/h centered at a velocity value configured in advance. ONE provides several advanced practical movement models to imitate different actual scenarios in life. Hereby, we cautiously equip every vehicle with Shortest-PathMapBasedMovement, in which Dijkstra's algorithm is used to find the shortest path along the connected road between two random map nodes. Having arrived at a destination, a carriage waits for a short time, and then, it would pick the next random target on some road of the grids and repeat the aforementioned moving process until the end of this round of simulation. Other essential parameters are listed in Table VIII.

Metrics for performance evaluation in this paper are the average message delay, average message loss ratio, and percentage of signature verified, which are represented as avgD_{msg}, avgLR, and avgPerSV, correspondingly, and are stated as follows:

$$\mathrm{avgD}_{msg} = \frac{1}{N_D \cdot M_{sent_n} \cdot K_n} \cdot \sum_{n \in D} \sum_{m=1}^{M_{sent_n}} \sum_{k=1}^{K_n} \cdot \left(T_{sign}^{n_m} + T_{transmission}^{n_{m_k}} + T_{verify}^{n_{m_k}} \right) \cdot \left(L_{n_{m_k}} + 1 \right)$$

where D is the simulation district, N_D is the total number of vehicles in D, M_{sent_n} is the number of messages sent by vehicle n, K_n is the number of vehicles within the one-hop communication range of vehicle n, $T_{sign}^{n_m}$ represents the time consumed for signing message m by vehicle n, n_{m_k} is one message sent by vehicle n and received by vehicle k, and $L_{n_{m_k}}$ is the length of buffer queue equipped in vehicle k when n_{m_k} is received by vehicle k. Then

$$\mathrm{avgLR} = \frac{1}{N_D} \sum_{n=1}^{N_D} \frac{M_{dropped}^n}{\sum_{k=1}^{K_n} M_{arrived}^{M_n}}$$

where $M_{dropped}^n$ means the total of dropped messages by vehicle n in application layer, and $M_{arrived}^n$ is the number of received messages in network layer by vehicle n. Here, consideration of message loss caused by wireless transmission is excluded, as leaving only message loss by security protocol due to full buffer space. In addition

$$\mathrm{avgPerSV} = \frac{1}{N_D} \sum_{n=1}^{N_D} \frac{M_{consumed}^n}{\sum_{k=1}^{K_n} M_{arrived}^n}$$

where $M_{consumed}^n$ means the total of consumed messages by vehicle n in application layer. In the following, we conduct a set of experiments to analyze the impacts of different traffic loads.

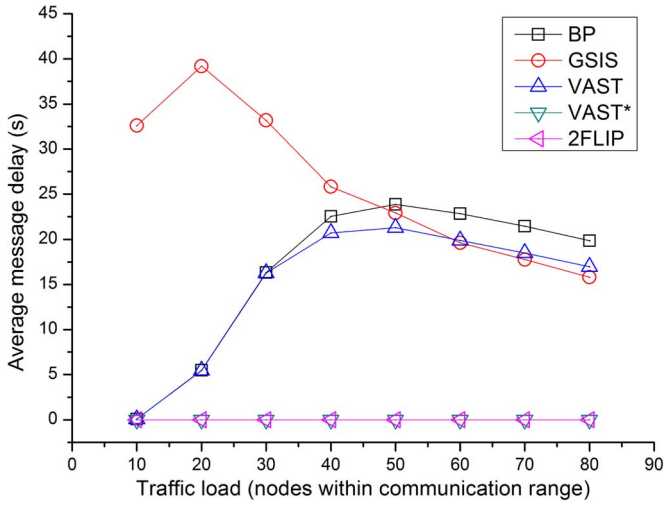


Fig. 13. Traffic load's impact on average message delay.

Following the simulation definitions earlier, we conduct a set of experiments to analyze the impacts of different traffic loads. The corresponding results are shown in Figs. 13–15.

As shown in Fig. 13, with the growth of traffic load, the average message delay avgD_{msg} of GSIS decreases dramatically when the traffic load is above 20. As for BP, VAST, and VAST$*$, it increases when traffic load is lower than 40 but decreases after that. The reason is that the buffering mechanism produces a time point, at which the buffer space is full; as a result, older unverified messages are dropped, but newer ones are verified and counted into statistics. While for 2FLIP, avgD_{msg} keeps nearly 0, which is fit for real-time emergency reporting applications.

As vehicle number in the communication range gets larger, the average message loss ratio of VAST$*$ and 2FLIP is stable at

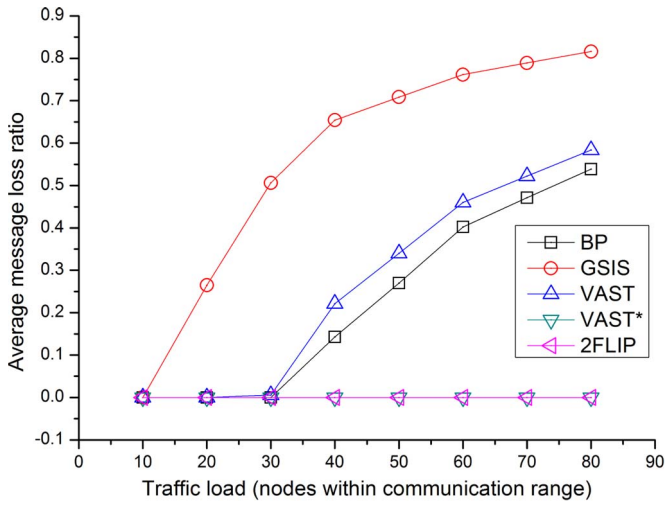


Fig. 14. Traffic load's impact on average message loss ratio.

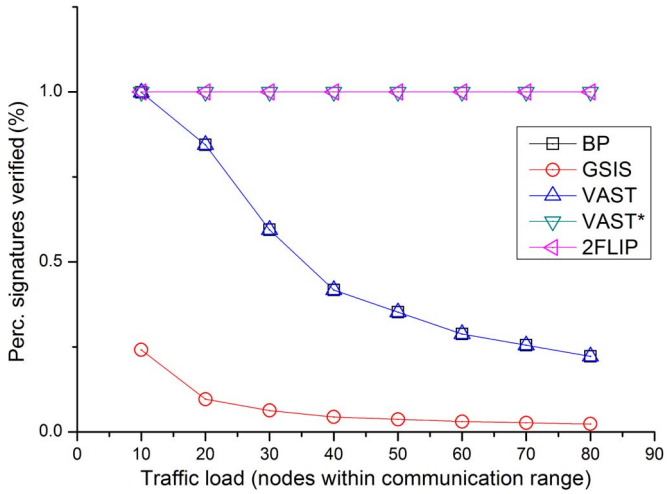


Fig. 15. Traffic load's impact on percentage of signatures verified.

0, even when the traffic load reaches 80, while for the other three schemes, it turns out a rapid growth when the traffic load is above 30. Moreover, when the traffic load reaches 70, average message loss ratio is higher than 50% for BP, GSIS, and VAST. Such situations exist in severe traffic jams, in which vehicles' buffer spaces are filled rapidly, and would make the VANET application unavailable.

Percentage of signature verified for both VAST∗ and 2FLIP keeps near 100% at all traffic loads, while for each of the other three, it is decreasing as traffic load gets larger. For GSIS, it is lower than 25%, even with only ten vehicles in the communications range, and for BP and VAST, it cannot keep 50% messages verified normally once the traffic load is more than 35.

The properties of 0 message loss ratio and about 100% signatures verified facilitate 2FLIP with resilience to DoS attack both in computation and in communication, which significantly increases the availability and stability of the VANET. Considering the aforementioned analysis of simulations, 2FLIP turns out to have the lowest average message delay, the lowest message loss ratio, and the highest of signature verified percentage. Although VAST∗ also performs well, it should be noted that it can never provide essential security features, such as unlinkability, conditional traceability, nonrepudiation, and others.

VII. Conclusion

In this paper, we have proposed a 2FLIP preserving authentication scheme, which employs two core methods: decentralization of CA and biological-password-based 2FA. Based on the decentralization of CA, the proposed scheme requires only several extremely lightweight hashing processes, and a fast MAC generation is needed for message signing and a hash function along with one fast MAC regeneration for verification, which increases efficiency of computation and communication. Extensive simulations reveal that the novel scheme is feasible and has an outstanding performance on message signing/verification, message loss ratio, and network delay. Moreover, decentralization of CA makes the certificate transmitting not necessary, which removes the overhead of certificate management. Through biological-password-based 2FA, 2FLIP achieves strong nonrepudiation that any biological anonym driver could be conditionally traced. To the best of our knowledge, 2FLIP is the first authentication scheme that achieves both strong privacy preservation and DoS resilience for secure VANET communication with the benefits of combining the two core methods. It also provides a feasible feature of offline biological password update to support the driving right transferring from one to the other of a vehicle.

In the current scheme, the security of the whole scheme relies heavily on the only system key from CA. Although now some of CA's responsibilities are decentralized to the portable telematics device and OBU, deadly threats still exist in applications of the VANET once the CA is compromised. In our future work, we will continue focusing on fully decentralized VANET authentication schemes, while maintaining the applicable efficiency.

References

[1] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proc. 3rd ACM Workshop Security Ad Hoc Sensor Netw.*, Alexandria, VA, USA, 2005, pp. 11–21.

[2] L. Armstrong, "Dedicated Short Range Communications (DSRC) Home," 2002.

[3] J. Harding *et al.*, "Vehicle-to-vehicle communications: Readiness of V2V technology for application," Nat. Highway Traffic Safety Admin., Washington, DC, USA, Tech. Rep. DOT-HS-812-014, Aug. 2014.

[4] K. Ren and W. Lou, "Privacy-enhanced, attack-resilient access control in pervasive computing environments with optional context authentication capability," *Mobile Netw. Appl.*, vol. 12, no. 1, pp. 79–92, Feb. 2007.

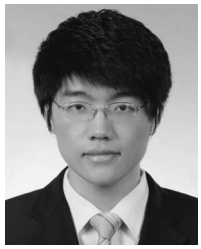
[5] M. Wang, D. Liu, L. Zhu, Y. Xu, and F. Wang, "LESPP: Lightweight and efficient strong privacy preserving authentication scheme for secure VANET communication," in *Computing*. Vienna, Austria: Springer-Verlag, 2014, pp. 1–24.

[6] L. Brown and W. Stallings, "User Authentication," in *Computer Security Principles and Practice*, 2nd ed. Upper Saddle River, NJ, USA: Pearson, 2012, pp. 71–105.

[7] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Security*, vol. 15, no. 1, pp. 39–68, Jan. 2007.

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

[9] "*IEEE Trial-Use Standard for Wireless Access in Vehicular Environments—Security Services for Applications and Management Messages*," IEEE Std. 1609.2-2006, 2006
[10] B. Libert and D. Vergnaud, "Multi-use unidirectional proxy re-signatures," in *Proc. 15th ACM Conf. Comput. Commun. Security*, Alexandria, VA, USA, 2008, pp. 511–520.
[11] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen, "ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications," in *Proc. INFOCOM 2008*, 2008, pp. 1903–1911.
[12] C. Zhang, R. Lu, X. Lin, P.-H. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in *Proc. INFOCOM 2008*, 2008, pp. 246–250.
[13] A. Fiat, "Batch RSA," in *Proc. Crypto*, 1989, pp. 175–185.
[14] J. Camenisch, S. Hohenberger, and M. Ø. Pedersen, "Batch verification of short signatures," in *Proc. Adv. Cryptology-EUROCRYPT*, 2007, pp. 246–263.
[15] X. Lin, X. Sun, P.-H. Ho, and X. Shen, "GSIS: A secure and privacy-preserving protocol for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3442–3456, Nov. 2007.
[16] D. Cham and E. van Heyst, "Group signatures," in *Proc. Adv. Cryptology-EUROCRYPT*, 1991, pp. 257–265.
[17] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Proc. CRYPTO*, 2004, pp. 41–55.
[18] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. Adv. Cryptology-Crypto*, ser. LNCS, vol. 196, 1984, pp. 47–53, New York, Springer-Verlag.
[19] G. Calandriello, P. Papadimitratos, J. Hubaux, and A. Lioy, "Efficient and robust pseudonymous authentication in VANET," in *Proc. 4th ACM Int. Workshop Veh. Ad Hoc Netw.*, 2007, pp. 19–28.
[20] L. Zhang, Q. Wu, A. Solanas, and D.-F. Josep, "A scalable robust authentication protocol for secure vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1606–1617, May 2010.
[21] K. Sampigethaya, M. Li, L. Huang, and R. Poovendran, "AMOEBA: Robust location privacy scheme for VANET," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1569–1589, Oct. 2007.
[22] R. Lu, X. Lin, T. H. Luan, X. Liang, and X. Shen, "Pseudonym changing at social spots: An effective strategy for location privacy in VANETs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 1, pp. 86–96, Jan. 2012.
[23] A. Studer, F. Bai, B. Bellur, and A. Perrig, "Flexible, extensible, and efficient VANET authentication," *J. Commun. Netw.*, vol. 11, no. 6, pp. 589–598, Dec. 2009.
[24] A. Perrig, R. Canetti, J. D. Tygar, and D. Song, "The TESLA broadcast authentication protocol," *CryptoBytes*, vol. 5, no. 2, pp. 2–13, 2005.
[25] X. Lin *et al.*, "TSVC: Timed efficient and secure vehicular communications with privacy preserving," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4987–4998, Dec. 2008.
[26] K. Ren, W. Lou, K. Kim, and R. Deng, "A novel privacy preserving authentication and access control scheme for pervasive computing environments," *IEEE Trans. Veh. Technol.*, vol. 55, no. 4, pp. 1373–1384, Jul. 2006.
[27] K. Sampigethaya *et al.*, "CARAVAN: Providing location privacy for VANET," Defense Techn. Inf. Center Document, Fort Belvoir, VA, USA, 2005.
[28] J. Daemen and V. Rijmen, "AES proposal: Rijndael," in *Proc. 1st Adv. Encryption Standard Candidate Conf., NIST*, 1998, pp. 1–45.
[29] M. Bellare, R. Canetti, and H. Krawczyk, "Message authentication using hash functions: The HMAC construction," *RSA Lab. CryptoBytes*, vol. 2, no. 1, pp. 12–15, 1996.
[30] F. Hess, "Efficient identity based signature schemes based on pairings," in *Proc. Sel. Areas Cryptography*, 2003, pp. 310–324.
[31] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," in *Proc. ASIACRYPT*, 2001, pp. 514–532.
[32] J. Katz and Y. Lindell, *Introduction to Modern Cryptography: Principles and Protocols*. Boca Raton, FL, USA: CRC, 2007.
[33] M. Scott, "Efficient implementation of cryptographic pairings," Dublin City University, Dublin, Ireland, 2007. [Online]. Available: http://ecrypt-ss07.rhul.ac.uk/Slides/Thursday/mscott-samos07.pdf
[34] C. Zhang, X. Lin, R. Lu, P.-H. Ho, and X. Shen, "An efficient message authentication scheme for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3357–3368, Nov. 2008.
[35] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proc. 2nd Int. Conf. Simul. Tools Techn.*, 2009, pp. 55.
[36] P. Papadimitratos *et al.*, "Secure vehicular communication systems: Design and architecture," *IEEE Commun. Mag.*, vol. 46, no. 11, pp. 100–109, Nov. 2008.
[37] H. Hsiao *et al.*, "Flooding-resilient broadcast authentication for VANETs," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Netw.*, 2011, pp. 193–204.
[38] D. Dolev and A. C. Yao, "On the security of public key protocols," in *Proc. 22nd Annu. Symp. FOCS*, 1981, pp. 350–357.
[39] B. Blanchet, "Automatic verification of correspondences for security protocols," *J. Comput. Security*, vol. 17, no. 4, pp. 363–434, 2009.
[40] R. Canetti and S. Gajek, "Universally composable symbolic analysis of Diffie-Hellman based key exchange," Int. Assoc. Cryptologic Res., Newcastle upon Tyne, U.K., 2010. [Online]. Available: http://eprint.iacr.org/2010/303.pdf
[41] R. Canetti and J. Herzog, "Universally composable symbolic analysis of mutual authentication and key exchange protocols," in *Proc. Theory Cryptography Conf.*, 2006, pp. 380–403.
[42] Z. Zhang, L. Zhu, L. Liao, and M. Wang, "Computationally sound symbolic security reduction analysis of the group key exchange protocols using bilinear pairings," *Inf. Sci.*, vol. 276, no. 20, pp. 93–112, Nov. 2012.
[43] F. Wang, "twoflip_proverif, ProVerif program for phases in 2FLIP scheme," 2015, [Online]. Available: https://github.com/finleywang/twoflip_proverif



Fei Wang (M'12) was born in Shandong, China, in 1988. He received the B.S. degree in computer science from the Beijing Institute of Technology, Beijing, China, in 2011. He has been working toward the Ph.D. degree with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, since 2011.

From 2011 to 2014, he was a Research Assistant with the Institute of Computing Technology, Chinese Academy of Sciences. His research interest includes privacy and security issues in vehicular ad-hoc networks and wireless sensor networks, fundamental study of vehicular telematics devices and cloud platforms, and new trends in automotive electronics.

Yongjun Xu (M'06) received the B.Eng. degree in computer communication from the Xi'an Institute of Posts and Telecommunications, Xi'an, China, in 2001 and the Ph.D. degree from the Institute of Computing Technology (ICT), Chinese Academy of Sciences, Beijing, China, in 2006.

In 2008, he joined the ICT, Chinese Academy of Sciences, where he is currently an Associate Professor. His current research interests include cyberphysical systems and multisensor data fusion.

Hanwen Zhang (M'07) received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2003 and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2009.

Since 2009, she has been with the Institute of Computing Technology, Chinese Academy of Sciences, where she is currently an Associate Professor. Her research interests include wireless and mobile networks and future internet architecture.

Yujun Zhang (M'04) received the B.S. degree from Nankai University, Tianjin, China, in 1999 and the Ph.D. degree from the Chinese Academy of Sciences, Beijing, China, in 2004.

He is a Professor with the Institute of Computing Technology, Chinese Academy of Sciences. From 2009 to 2010, he was a Visiting Scholar with the University of California, Los Angeles (UCLA), CA, USA. He is author or coauthor of more than 50 journal and conference papers. He also has 15 patents in China and has published one book on trusted network protocols. His current research interests include network security and future Internet architecture.

Dr. Zhang has served on the technical committees of several national and international conferences.

Liehuang Zhu (M'11) is currently a Professor with the School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China. He is an expert in network security. His research interests include security protocol analysis and design, group key-exchange protocol, wireless sensor networks, and cloud computing.