Improved CRL Distribution point (ICRLDP) for Cooperative Intelligent Transportation Systems (C-ITS)

Yves Christian, Elloh Adja, Ahmed Serhrouchni

Improved CRL Distribution point (ICRLDP) for Cooperative Intelligent Transportation Systems (C-ITS)

Yves Christian Elloh Adja
Telecom Paristech, France
elloh.adja@telecom-paristech.fr

Ahmed Serhrouchni
Telecom Paristech, France
ahmed.serhrouchni@telecom-paristech.fr

***Abstract*—The Cooperative Intelligent Transportation Systems (C-ITS) are already part of our daily life, and their adoption is exponentially increasing, especially with the rise of smart cities concept. However, the security of these infrastructures remains a critical and significant challenge to meet. The Public key infrastructure (PKI) using certificates is the most popular solution to address security issues. The vehicles are identified by a lot of pseudonyms certificates, which must be revoked when the vehicle becomes misbehaving or faulty. The use of multiple certificates introduces new critical problems, like the certificate revocation issue. The revocation management is critical for a PKI, even worse in vehicular communications, where there are many certificates to revoke. All nodes of a network must be aware of all pairs' revocation status as soon as possible to prevent the revoked node from unauthorized activities in the network. The revocation is still an open challenge that is starting to attract a lot of attention from researchers. In this paper, we propose a new scalable and reliable approach called improved certificate distribution point system (ICRLDP), which aims to disseminate vehicle revocation information in a distributed way. Our plan proposes a trade-off between vehicle privacy and security.**

I. INTRODUCTION

In the Vehicular communication (VC) architecture, vehicles can directly communicate with other vehicles in vehicle-to-vehicle(V2V) mode or exchange pieces of information with Roadside Units (RSUs) in a vehicle-to-infrastructure (V2I) mode. These communications are generalized with the name of vehicle-to-everything (V-to-X). A critical requirement of such technologies is to find an adequate balance between security and privacy. The messages exchanged between vehicles must be authenticated, dissuading misbehavior, and preventing data forgery. The user's privacy must be preserved at the same time, so the system is not used for mass surveillance purposes. This requirement calls for a privacy-preserving and scalable Vehicular Public Key infrastructure (VPKI) [1]. Among the existing VPKI solutions, the most prominent proposals are those from the Cooperative Intelligent Transport System (C-ITS) [2], led by the European Telecomunications Standards institute (ETSI), and the Security Credential Management System (SCMS) [3] [4], which is part of the IEEE 1609.2 standard. This paper is focused on ETSI standard because it is a part of an ETSI project. The Cooperative Intelligent Transportation Systems (C-ITS) will significantly improve transportation safety, traffic efficiency, driver's comfort, reducing traffic congestion in the near future. Despite the advantages that C-ITS can offer, there are still many critical issues concerning the deployment. The C-ITS relies primarily on a vehicular network to ensure communications between vehicles, and between vehicles and infrastructure. In the C-ITS VPKI architecture, a set of certification authorities (CAs) provides credentials to legitimate vehicles or RSUs. In this way, each message in C-ITS is signed by the sender. The architecture allows vehicles to own two types of certificates; namely, the long-lived (long-term) [5] and the short-lived certificate (pseudonym) [5]. The long-lived certificate is issued only on time by the Root CA for the vehicles at the manufacturing phase. The vehicle must to maintains it for all its life and absolutely must not disclose it. It is credentials that identify the vehicle. The short-lived certificate (pseudonym certificate) is related to the long-lived certificate. It does not provide significant information about vehicle identity, so it is used to authenticate by preserving vehicle anonymity. However, to avoid traceability, each vehicle must use multiple short-lived certificates. Hence, a vehicle can use a new short-lived at each message sent, each minute or each hour. In our work, we consider a C-ITS that relies on the architecture described by the standard ETSI TS 102941, which set certificate lifetime at one hour. Each message sent by a vehicle is signed and contains the short-lived certificate of the sender, to allow authentication and non-repudiation properties on messages. The certificate verification process includes an unavoidable step, the verification of revocation status. Indeed, numerous events such as misbehaving, the loss or modification of private keys, the sale, or the leasing of a vehicle can cause the revocation of vehicles by Root CA. The PKIs use a default model were potentially compromised certificate remains valid until their expiration date or until they are revoked. The unique solution to stop all activities of a compromised vehicle in the network is the revocation of all its certificates. The certificate revocation process and dissemination need to be clear for the CAs and all the nodes of the network. The ETSI standard formerly cited proposes the Certificate Revocation List (CRL) mechanism for disseminating certificate revocation information to all C-ITS nodes. The CRL is a blacklist file which contains the list of all revoked short-lived certificate by PCA. Each vehicle can own more than three thousand certificates, so the revocation of only one vehicle cause de adds of maybe three thousand certificates to the CRL. It is well-known that this fact induces the exponential growth of CRL size [6] and hence makes it unable to address future scalability challenges. In this paper, we propose (1) a new CRL dissemination method to solve CRL oversizing issues. (2) we evaluate the performances of our approach in comparison to

the standard CRL mechanism. (3) we propose a quantitative performance analysis and applicability of our approach in the current C-ITS standardized architecture.

II. Related work

There are some approaches for revocation information dissemination, which are standardized in C-ITS. We show it below:

Certificate Revocation Lists (CRL) [7] represent the standard mechanism. It is a blacklist file which contains the list of all revocated certificate by a CA. It's updated periodically according to the CA policy and downloadable from a dedicated revocation server. To be used, the end-user (e.g., device, vehicle, etc.) must download the whole CRL file at each update time. Thus, the CRL method is not scalable because the CRL file size is directly proportional to the number of revoked elements. It became a big file with the growth of revoked certificates.

There is another type of revocation dissemination approach used in internet network under Transport Layer Security (TLS), which is not standardized in C-ITS, such as CRL distribution point. The CRL Distribution Points (CRLDP) is an improvement of CRL, which is specified in [7]. The main idea is to face the scalability issue fragmenting the CRL into the smallest parts. The fragments are organized logically, to allow the user to download only parts that contain pieces of information about the owned certificate. Indeed, each issued certificate is linked to a distribution point. Each fragment is defined as a distribution point. The solution presents some drawbacks, like the non-uniform growth of different pieces [8]. The CRL Distribution point is the most used revocation method in the internet network.

In the last years, V-to-X security related to revocation management in the PKI has gained extensive attention. Many works have addressed the revocation dissemination issue, which we are starting to quote.

Jason j. Haas and Yih-chun Hu propose in [9] the uses of delta-CRL in an optimized form for disseminating revocation information. Indeed, a vehicle is identified by many pseudonyms and a key *sk*. Each certificate contains a certificate-ID obtained by encrypting an integer with a key *sk*. The key *sk* is shared between the vehicle and CA. The list of integers to encrypt is well-known for each vehicle. When the CA is revoking a certificate, it adds the critical *sk* in the CRL. Each vehicle has to retrieve the keys *sk* from CRL and encrypt the list of well-known integers, to obtain all certificate-ID of the revoked vehicles. These certificate-IDs are inserted in a bloom filter, which is stored locally, so any certificate-ID is not stored in the way to save storage. The revocation status of a certificate can be ensured, only checking the belonging of certificate-ID to the local bloom filter. Finally, they propose the pre-loaded certificates to address the bloom filter false positive issue. The solution proposed is not scalable, because it moves the scalability issue from CRL to the local Bloom filter. The CRL dimension will grow not exponentially but linearly, and the Bloom filter exponentially at each vehicle revocation.

Authors in [6] propose a design of three new protocols for revocation management. The Compressed Certificate revocation List (C^2RL) is a bloom filter used for CRL overhead reduction. The RTPD (Revocation of the Tamper-Proof Device) protocol is a solution to make the revoked vehicle to be aware of its revoked status. The approach is to perform revocation, just preventing a vehicle from using owned certificates when it is revoked. The last is the DRP (Distributed Revocation Protocol), which proposes an alternative method to revoke a certificate when the CA cannot do it because not reachable. The RTPD works only if the vehicle to revoke is not corrupted by an attacker or is not the attacker itself. The (C^2RL) approach is not scalable because the Bloom filter calculation difficulty is related to the CRL dimension.

Similarly to the [6], the [10] proposes the Compressed Certificate revocation List (C^2RL) solution that uses bloom filter to compress the common CRL. Then the RTC protocol is based on the fact that (Trusted Component) is active in all vehicles. So the CA revoke a vehicle sending a message to TC, which erased all owned certificates. If communication is not possible between CA and TC, the (C^2RL) is published from RSUs. The paper also proposes the Misbehavior Detection System (MDS) protocol and the Local Eviction of Attackers by Voting Evaluators(LEAVE), a collective warning system against misbehaving nodes. As mentioned yet, the (C^2RL) is not scalable, and the RTC protocol cannot face corrupted vehicles and vehicle attackers.

Rigazzi and Tassi also propose in [11] an evaluation of the efficiency of (C^2RL) in the C-ITS network to introduce an optimization framework to jointly minimize the filter size and the number of hash functions employed according to a predefined probability of false positives.

The paper [12] proposes an encoding of CRLs into numerous cryptographically self-verifiable pieces, allowing revocation verification having only a considerable part of CRL pieces. Moreover, this reduces the bandwidth requirement for CRL dissemination by RSUs to keep efficient distribution without interfering with other C-ITS traffic. The number of self-verifiable pieces will grow according to the CRL growing, and many parts can become difficult to disseminate and handle efficiently.

Authors in [13] proposes a solution to make CRL dissemination faster, making it in an epidemic distribution way. The solution uses the delta-CRL method, where vehicles have to download the entire CRL only once and at the beginning. After that, only the CRL updates are downloaded. As the previous solution, the storage constraint and scalability have not addressed.

The contribution [14] proposes a Blockchain-based authentication mechanism for inter-vehicles communication. The authors aim to eliminate the single point of failure and reduce the communication and verification overheads in the PKI while ensuring authentication. The revocation issue is addressed moving the CRL in the Blockchain. The revocation status verification is resumed to a query to the Blockchain. This approach is not reliable because Blockchain needs constant

communication and synchronization between all nodes. This need is a difficult challenge to meet in a network like C-ITS.

Authors in paper [15] propose a certificate revocation scheme based on Blockchain concept. The Blockchain is introduced to store and to simplify the network structure and distributed maintenance of the certificate revocation list (CRL).

There exist others works [16] [17] [18] [19] which address security in C-ITS with PKI and digital signatures whithout proposing any mecchanism for certificate revocation.

Some aspects of revocation were discussed in [20] [21] [22] without a complete solution provided.

III. System Model and Design

We describe herein the network and the main entities involved in our system model. We also discuss our solution design as well as the procedure to reduce CRL overhead and to perform certificate verification.

A. Network Model

In a typical vehicular architecture, there exists a root entity called Root CA (RCA), which coordinates vehicles and infrastructure authentication within a predefined jurisdictional area, like a city, region, country, and so on, registering vehicles and assigning long-lived certificates. Only the RCA can revoke the long-lived certificates. There are also subordinate authorities called Pseudonym CAs (PCA) responsible for issuing and revoking a pseudonyms certificate (short-lived certificate) and CRLs. The RCA maintains the mapping between short-lived and long-lived certificates of vehicles securely. We assume that potential attackers cannot compromise RCA and PCAs. The RSUs are deployed along the roads, and each RSU is connected to a single PCA via a wired network. We assume that RSUs are well deployed along all streets. The connectivity Vehicle-to-infrastructure is achieved through Dedicated Short Range Communications (DSRC) interfaces such as ETSI ITS-G5. The vehicles involved in the network sign and broadcast safety-related messages like Cooperative Awareness Messages (CAM) and Decentralized Environmental Notification Messages (DENM), attaching the sender certificate. We also assume that vehicles are provided with a tamper-resistant Hardware Security Module (HSM) storing the cryptographic material.

B. Certificate Organisation

In order to communicate securely, each node needs a pre-installed certificate in vehicles or updated during production, periodic service appointments, or sale. We do not consider the architecture where short-lived certificates are issued one by one by PCA after each one expiration. The C-ITS aims to achieve a tradeoff between privacy and authentication while ensuring the non-traceability of vehicles. Received messages must be trusted to be accepted, and this requires authentication of the sender, preserving its privacy. Indeed, senders wish to retain as much privacy as possible. The use of only one pseudonym certificate can authenticate and protect the vehicle identity privacy, but it cannot protect against traceability in the network. Indeed, each message sent contains the sender's pseudonym credential, so an attacker can trace all messages sent by a victim and discover the location of interest (e.g. home location of the victim) by listening to DSRC.

C. Certificate Group

In the goal of protecting a vehicle from traceability, the PCA issue a group of pseudonym certificates that are not related to each other. However, the set of pseudonym certificates must be linked to the long-lived certificate of the vehicle. It must be difficult for an attacker to go back to the long-lived certificate from any pseudonym certificate. Only the RCA has the privilege to realize such operation. Hence, vehicles can frequently change their certificates in order to be challenging to trace. Its are free to choose its policy about certificate change frequency. Its can change certificate each hour or at each message sent without exceeding the certificate lifetime set by the standard. The number of short-lived certificates to issue to each vehicle and the certificate group renewal frequency depends on the PCA internal policy or the PKI architecture. Nonetheless, there are some parameters to considers, like internal vehicle storage, bandwidth use, network topology, and so on.

The European normative imposes a technical control on a new vehicle every four years and every two years for vehicles with more than four years old. We will use four years as the certificate renewal period.

IV. Short-lived Certificate aggregation

We assume that a vehicle V owns n_v pseudonym certificates, and that certificate IDs are unrelated, in the way that is complicated for an attacker to discover vehicle identity from a short-lived certificate. All certificates are securely and locally stored by the vehicles. We introduce the encryption/decryption symmetric key S_k shared only among PCA and vehicles. Each vehicle shares an S_k with the PCA. The S_k is used to produce a new field in the vehicle short-lived certificate, the *verification-text* (v_t). The *verification-text* takes place in the *certificate ID (certID)* field in order to allow compatibility with the standard as described by Equation 1.

$$size(v_t) + size(certID) < MaxSize(CertID) \quad (1)$$

The S_k is a symmetric key; it is generated by the PCA for each vehicle. It represents an encryption/decryption key which takes fixed length inputs and permutes them into a static output. The same input must to always produces the same output.

The PCA generates a list of integers *(v,..,n)*, *v* and *n* are random values generated by PCA for each vehicle. They do not have to be necessarily big integers, because the size of the integer doesn't influence our system security. Indeed the system security depends on the encryption algorithm. *v* and *n* are known by everyone and the integers list length must be equal to the number of pseudonyms issued to the vehicle. Indeed, each integer i in the range $(v...n)$ is related to a single certificate, but it is not provided clearly in the certificate. The

verification-text is the encryption output of the certificate's related integer with *sk* ($v_t = E_{sk}(i)$). The *verification-text* is different for each certificate in order to preserve its privacy. For each i, we can resume the certificate as described by Equation 2:

$$Cert_i = K_{pub}, E_{sk}, sig_{PCA} \quad (2)$$

Where K_{pub} is the public key of the vehicle, the E_{sk} is the verification-text, and the sig_{PCA} is de signature of PCA on the whole certificate. All these fields are generated by PCA k times for a vehicle. Now, the S_k resume all the vehicle's certificates. Indeed, we exploit this feature to enhance the revocation process and to reduce the CRL size.

V. ICRLDP Approach

We can now introduce our new approach called Improved CRL Distribution Points (ICRLDP). Our method fragments the full CRL into sub-sets such that each fragment becomes a smaller CRL. We define each fragment as a distribution point. The fragment size and the set of the related certificate issued depend on PCA policy. Each vehicle's set of short-lived certificates are related to a single distribution point. The ICRLDP is, we can say, the adaptation of internet CRLDP to the vehicular network. It structure is described in the picture 1.

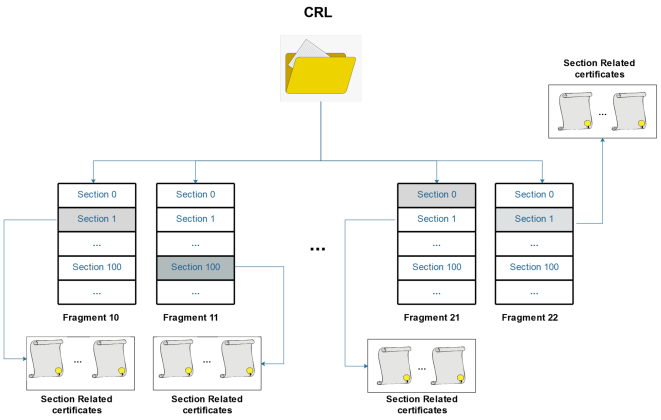


Fig. 1: ICRLDP structure

We want to underline that our system is characterized by some parameters: (1) F_d which is the single fragment dimension expressed in the quantity of S_k contained, (2) F_n the fragments quantity and (3) C_f Certificates renewal frequency. These parameters help us to find a trade-off to make reliable our system. We made a statistical estimation of the CRL dimension to show the need and the relevance of a new revocation approach. Our evaluations are based on statistical data published by the European Union [23] and the French government [24]. As described previously, car theft is one of many reasons causing the revocation of car credentials. Since realistic data are not yet available, for this study, we reduce revocation causes to only vehicles' theft. For geographical areas, we focus on France. Indeed, the used statistical data have been obtained from the French government. The vehicle theft information is published yearly by the French home office [24] and the European Commission [23]. In 2018 Police recorded an average of *177300* vehicle thefts in France. We model it as τv. The number of vehicles in circulation is estimated to *33,020,000* [25] which we model as V_c. The revocation percentage of vehicles in circulation is around of *0.43%*.

We define as α sum of vehicles revoked for apparent reasons on which we don't have any pieces of information. So the size of the CRL value can be:

$$size(CRL) \sim [(\tau v * n_v) + \alpha] * size(CertID) \quad (3)$$

A numerical estimation can be provided if we take as described in the standard [2] one hour as a pseudonym certificate lifetime. So we have a group of 35040 certificates assigned to each vehicle for four years. We accept four years as C_f as argued firstly, and a CertID size of 16 Bytes. The CRL size can be approximated to $(177300 * 35040 * 16)/1024 = 97.071.750 KBytes$, in other words, 97 GBytes. This size value is too high, so not acceptable to avoid non-negligible CRL download latency and network bandwidth usage. Fragmenting the huge CRL in small independent pieces will solve CRL size issues and reduces CRL download latency. In this example case, the CRL can be divided, for example, into 500 fragments. This fragmentation produces a fragments size of 19 Mbytes. This quantity of ICRLDP fragments is realistic because it is the actual quantity of distribution points in the internet network, as identified by paper [26].

The ICRLDP mechanism is not standardized, so we have to make some changes in the vehicle's certificates to allows its use. The CRL ASN.1 structure is described in the ETSI standard TS 102 941 V1.2.1 [27]. That structure will be slightly modified, allowing retro compatibility, as shown in picture I. We only added a new field called ICRLDP ID, which is the fragment identifier. It has a size of 16 bytes.

Field	type
Version	INTEGER
CRLDP ID	OCTET STRING (SIZE(16))
This update	Time32
Next update	Time32
Sequence of CRL Entry	OCTET STRING

TABLE I: ASN.1 Structure of CRLDP

We assume that RSUs are trusted and have enough storage capacity to store all CRL distribution points.

A. Certificate new Field

Some changes must be made in the certificate structure to allow the use of our revocation mechanism. These changes are minor to ease integration and to allow retro-compatibility. The certificate field [27] *crlseries* assume a new value *u* to states

that revocation check is performed through CRLDP mechanism. According to this, vehicles must check the *crlseries* field to be aware of the revocation method type to uses.

B. Fragment features

We introduce a logical order in the fragment to ease keys research, so we subdivide the fragment into logical sections. Each section is related to a group of vehicles S_{veh}. Each fragment have a dimension F_d. The PCA must define the value of C_f and F_d. The definition of these values allow an estimation of F_n as described by Equation 4:

$$F_n = \left\lceil \frac{V_q * C_f}{F_d} \right\rceil \tag{4}$$

We define section dimension (Sec_k) as the maximal number of (S_k) that a section can contain. We also define Tot_{veh} as the total quantity of vehicles in circulation in a selected geographical area. Hence, it is possible to deduce the maximal quantity of vehicle covered by each fragment (Ff_{cov}) so:

$$Ff_{cov} = \frac{Tot_{veh}}{F_n} \tag{5}$$

The maximal quantity of vehicle covered by each section Fs_{cov} is computed according to Equation 5:

$$Fs_{cov} = \frac{Ff_{cov} * Sec_k}{F_d} \tag{6}$$

We define v_{tot} the number of vehicle related to a section and c_{tot} the number of pseudonym certificate owned by each vehicle. We also define rev_{tot} the section revoked vehicles. As we have previously mentionned the relation between vehicle and ICRLDP lines is *1:1*. Revoking a vehicle means write its S_k in the related ICRLDP fragment. The number of certificates related to a section $cert_{section}$ is expressed:

$$cert_{section} = v_{tot} * c_{tot} \tag{7}$$

We propose this fragment subdivision to reduce the number of keys S_k to check during the revocation control process. Hence, the certificate revocation status check implies to try only the keys in the certificate section and not all keys in the fragment. This fragment subdivision has two extreme scenarios. The best scene is to have only one revoked vehicle in each section ($rev_{tot} = 1$), so the revocation check will imply only one decryption. Unfortunately, there is also the worst scenario, which happens when all vehicles in a section are revoked ($rev_{tot} = v_{tot}$). The last one is the scenario, which must be avoided. The two extreme scenarios are statistically unlikely because it is not possible to predict vehicle revocation. We recommend assigning vehicles to distribution points progressively according to the evolution of CRLDP's size.

A PCA must do a thorough study of parameters like geographical vehicle distribution, authority area, S_{veh}, C_f, and F_d to avoid traceability of vehicles and improve efficiency. The PCA choice of parameters values must be made keeping present these constraints:

- The CRL fragments must be smaller than the whole CRL and bigger than the size of a short-lived certificate.
- The CRL fragmentation must follow vehicle distribution on the geographical area of interest.
- Each fragment must be signed by PCA.
- The fragments must be independent of each other.
- The fragments are distributed by RSUs and vehicles in some cases.

All these fragments are stored by infrastructure nodes like RSU, which are enough memory capacity to store all CRL fragments of all PCA independently of a well-known geographical area.

C. Fragment Dissemination

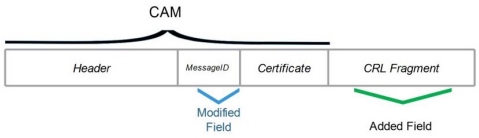


Fig. 2: New CAM Message Structure

All sent messages must contain a certificate as described in the ETSI [27] standard and now a ICRLDP fragment. The inclusion of ICRLDP fragments in messages helps its dissemination in an area where RSUs are not well distributed. Vehicles can download fragment which it belongs at each update time when it is in RSUs communication range. The vehicles can also exchange each other new or updated fragment version directly in a securely way granted since the PCA signs all fragments. A vehicle that wants to update its CRL fragment must send a ICRLDP request. It does not represent a new message, but a ordinary CAM message which has a particularly *messageID* (Fig 2). We propose a new *messageID* value as *p*. The vehicle request will contain an expired fragment because it is just looking to update an expired one. This message will be ignored by vehicles that are not attached to the same distribution point with the sender. The request will be accepted only by RSUs or vehicles that belong to the same distribution point or which have the right fragment. Regarding the receiver side, there are two cases. (1) If the receiver is a vehicle, it will respond with an ordinary CAM message. Then, the sender will retrieve the fragment attached to the message. (2) If the receiver is an RSU, the last one will respond with a CAM message, attaching the requested fragment. All vehicles that belong to the same distribution point and in the responder communication range will update their fragment if needed, avoiding sending news requests. The other vehicles will ignore the message. The RSU will satisfy the request if there no prior response from another vehicle. Therefore, it will be the first receiver of the request, which has the right to respond independently of receiver nature. As we can see, our system also works in an area where RSUs are not well distributed.

VI. Revocation Checking

First, we underline the basic rules related to our system about CRL fragment dissemination.

- Vehicles can download fragments from RSUs or other vehicles.
- RSUs can deliver ICRLDP fragment issued by differents PCA in the geographical area assigned.
- Vehicles must add to all sent messages the own fragment.

According to ETSI TS 102.941 standard [27], only the RCA has the capacity of a vehicle's revocation. The RCA proceeds to revoke the vehicle's long-term certificate; then, all owned short-term certificates are invalidated at the PCA level. To revoke a vehicle, the PCA publish in the vehicle-related fragment the symmetric key S_k shared with the revoked vehicle, which can decrypt all differents *verification-text* of the revoked vehicle pseudonym certificates. So it will appear for each revoked certificate the line:

$$line_i = \{S_k\}$$

Where S_k is the encryption key used to produce the *verification-text*. When a message is received, the vehicle or RSU checks the validity of the attached certificate and its revocation status. Then, verify that the fragment attached is the right one. The vehicle proceeds opening the fragment, then takes the right section and use key by key a.k.a S_k for decrypt the cipher a.k.a *verification-text* of the peer certificate. The certificate is not revoked if there is no S_k in the fragment section able to decrypt its *verification-text*.

Algorithm 1: Revocation status check

Input : Ct= Certificate, Cl=ICRLDP Fragment
Output: S=Bool (True = revoked, False= Not revoked, Certificate revocation status

```
Check revocation (P,S)
    N ← Ct
    C ← Cl
    if N ∉ ∅ ∩ C ∉ ∅ then
        if cert_frag_belonging(N, C) then
            Vt = load_verification_text(N);
            dt = Load_CRLSection(C);
            for i = 0; i < length(dt); i++ do
                tx = decrypt(dt[i], Vt);
                if tx == True then
                    return true;
                end
            end
        end
        else
            return Error;
        end
    end
    else
        return Error;
    end
    return False;
```

The algorithm 1 explain deeply how the revocation status is checked by a node when it receives a message. The algorithm takes as input the peer certificate and the attached ICRLDP fragment. It first checks the membership of the certificate to the attached ICRLDP fragment. Then the function *load_verification_text* retrieves the *verification_text* from certificate. The function *Load_CRLSection()* retrieves the right section. The function return bool variable, which is *True* if the *verification-text* is decrypted which means that the vehicle is revoked. The False value states that the vehicle is not revoked.

The ICRLDP fragments are signed by PCA, so it will always be possible to ensure its validity. The ICRLDP fragments are updated periodically, so it contains an expiration date, which is checked by the users. There is no way for the attacker to use an old fragment.

VII. Implementation and Performance Evaluation

In this section, we first show our implementation scenario, then we analyze our distribution system with tests and comparing the obtained results with the standard CRL mechanism.

We used the network simulator called veins [28] to perform our tests. Veins is an open-source framework for running vehicular network simulations. It is based on two well-established simulators: OMNeT++[29], an event-based network simulator, and SUMO [30], a road traffic simulator. Our implementation is written in C++ and executed in the simulator. The simulator was installed in a desktop PC with these features: core i7-3770, X86_64 64 bits, 3.4 GHz, 8 GB memory RAM, Ubuntu 16.04. We used Veins to reproduce real road traffic conditions, using a part of the 15th district Map. Indeed we used 600 m * 600 m around of our laboratory in Télécom Paris. We collected data from *400* different vehicles including RSUs.

We only use CAM Beacon messages to request and receive fragment as described previously. No new message types are created. The CRL fragments are directly created in RSUs; we disregarded exchanges between PCA and RSUs. The mobile entities of our simulation are vehicles, and the static nodes are RSUs. We evaluate three metrics: Time to acquire or update a CRL Fragment, the number of sent requests and the number of received messages by each vehicle. We study the minimal time required to get or update a CRL fragment as the first metric. We consider the amount of received message by single vehicle to estimate the overproduction of data and data distribution due to our solution. We also consider the number of sent request by a single vehicle to estimate the number of requests which must be sent to get or update a CRL fragment.

We perform multiple tests of one-hour duration each. The simulation parameters are described in Table II.

t_{req} = *Time of the first ICRLDP or Crl request.*
t_{res}= *The first reception of requested fragments (s) .*
σ = *Standard Deviation.*
τ_{updt}= *Requested time to update CRL or ICRLDP status (s) .*
$\bar{m}$= *Average of received messages by each vehicle (All receivable Messages).*
$\bar{r_m}$= *Average of request messages sent.*
B_r=*Beacon interval (s)*

Info	Value
Vehicle count	80
Through Traffic factor	40
Accident Count	4
Accident Duration	50
ICRLDP Dimension	500 lines
ICRLDP Lifetime	1 h
Max vehicle speed	50 km/h

TABLE II: Simulation Parameters

$\tau_{updt} = t_{res} - t_{req}$

Fields	F_n	B_r (s)	τ_{updt}		$\bar{m}$		$r_{\bar{m}}$	
			$\bar{A}$	σ	$\bar{A}$	σ	$\bar{A}$	σ
CRL	1	0.25	7.1	11.7	3062	2596.1	54.5	78.1
		0.5	3.6	5.4	1640	1342.5	31.3	42.4
		0.75	2.7	5.4	1123	973	18.1	23.6
		1	2.3	3.1	842	686.8	16.5	20.2
		5	0.8	1.8	198	181.2	5.08	10.3
ICRL DP	2	0.25	0.8	3	2805	2088	10.8	32.1
	5		0.5	2	2962	2351	6.6	21.1
	10		0.5	2.8	3757	2478.5	5.1	20.9
	15		0.03	0.2	4771	2972.2	3.5	11.8
	20		0.05	0.2	3135	2328.2	5.3	18.4
	2	0.5	1.1	4	2040	1409	1.7	10.5
	5		0.6	2.9	2018	1302.1	3.8	12.3
	10		0.2	0.7	1576	1172.4	2.6	7.4
	15		0.2	1.7	1878	1273	2.3	8.6
	20		0.2	0.9	1465	1038.8	3	8.3
	2	0.75	0.07	0.4	1095	959	5.9	25.4
	5		0.03	0.2	958	762.4	2.9	6.9
	10		0.03	0.2	928	777.2	0.1	0.9
	15		0.06	0.3	948	807.6	5	17.8
	20		0.03	0.2	908	747.3	0.1	0.8
	2	1	0.2	0.6	613	697	11.2	28.6
	5		0.06	0.2	893	772.9	2.7	6.2
	10		0.04	0.2	795	585.5	2.1	4
	15		0.06	0.2	719	646.4	2.8	5.6
	20		0.08	0.5	641	620.3	4.5	16.5
	2	5	0.3	0.8	171	142.8	2.5	4.6
	5		0.16	0.4	168	131.3	1.7	2
	10		0.16	0.4	229	143.4	1.4	1
	15		0.19	0.6	163	132	1.7	2.8
	20		0.2	0.6	210	138	1.6	1.6

TABLE III: Simulation Results

We performed numerous tests which we divide into two parts. In the first part, we tested the traditional CRL method in the simulator environment and then, in the second part, our ICRLDP approach. The performed tests gave back results that we use as a comparison basis. We use the simulation parameters described in table II. We performed many tests varying the CAM beacon interval in the way to check if it has an impact on results. We choose the beacon interval value in the data range recommended by the standard [2] which is $100ms < x < 10000ms$. Where *x* is the beacon interval. The simulation results are described in table III. The table shows the average and the standard deviation of data collected from each vehicle involved in the simulation. We can saw in Table III that received messages $\bar{m}$ by a vehicle are proportional to the beacon interval. That has sense because if we increase the frequency of sent messages, there will be more exchanged data in the network and, therefore, more received messages by vehicles. We can also observe that weaker is the beacon interval, worse is the message reception. We can explain it as a consequence of interference and collision among messages.

In the second part, we performed many tests using our ICRLDP approach as the revocation distribution system this time. We varied the fragment quantity for each beacon interval used in the CRL simulation, to compare ICRLDP to CRL performance and to deduce the optimal fragment quantity. The tests show us in the table III that fragment quantity impact the parameters τ_{updt}, $r_{\bar{m}}$ and not $\bar{m}$. We can observe that more fragmented is the CRL better are the values of τ_{updt} and $r_{\bar{m}}$. On the other hand, there are not significative variations of $\bar{m}$ parameters regardless of fragment quantity. This result confirms the scalability of our solution.

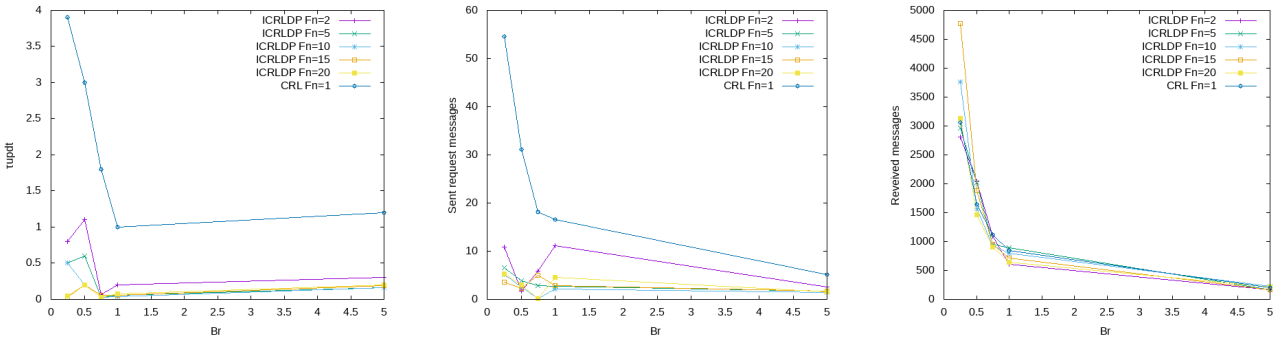
We have compared the time to update of CRL standard method with the time to update of ICRLDP as described in the graph 3a. We can observe that our approach produces better results regardless of fragment quantity. Moreover, revocation information retrieving is faster and comfortable. Indeed, vehicles can access quickly to the revocation information when available. According to the second metric, as described by graph 3b, the number of request messages needed to give back a fragment is lower using the ICRLDP and still regardless of the fragment quantity. Hence, better is the data distribution, lesser is the number of request messages required to recover the fragment. We compared in the graph 3c the two methods according to the received message (all received messages). We observe that the results are practically equal for the two methods. We deduce that our approach doesn't influence the bandwidth usage significantly. Our solution optimizes substantially the time to update and reduce the data to store by each vehicle without suffocating the bandwidth. Our results also show the optimal fragment quantity reduces our solution impact on the bandwidth usage. We focused on these three metrics because we think that they efficiently explain and efficiently summarize the advantages of a revocation solution.

VIII. Conclusion

The revocation distribution is a recurring issue that continues to get worse with the growth of connected devices. The standardized approach, the CRL, will arrive soon to a breaking point. In this paper, we proposed an original plan to solve the CRL problem by using a distributed and scalable mechanism for the revocation issue. We proposed a new data structure for the CRL, a compressed version that is exchanged directly between vehicles to avoid mandatory connection with RSU. All these are reached without introducing a new message type and without increasing traffic overload. The evaluation results obtained show that our approach is realistic and meets the requested requirements, such as the scalability more than the other existing methods.

References

[1] Mohammad Khodaei and Panos Papadimitratos. The key to intelligent transportation: Identity and credential management in vehicular commu-



(a) The graph shows the average time necessary for each vehicle to receive a good piece of CRL. This for ICRLDP method with different fragment quantity and CRL method

(b) The graph shows the number of request messages sent by each vehicle before receiving a good piece of CRL. This for ICRLDP with different fragment quantity and standard CRL methods.

(c) The graph shows the number of messages received by each vehicle in one hour. This for ICRLDP with different fragment quantity and standard CRL methods.

Fig. 3: Performance tests results comparison

nication systems. *IEEE Vehicular Technology Magazine*, 10(4):63–69, 2015.

[2] TS ETSI. 102 941-v1. 1.1 (2012-06)-intelligent transport systems; security; trust and privacy management.

[3] William Whyte, André Weimerskirch, Virendra Kumar, and Thorsten Hehn. A security credential management system for v2v communications. In *2013 IEEE Vehicular Networking Conference*, pages 1–8. IEEE, 2013.

[4] Marcos A Simplicio, Eduardo Lopes Cominetti, Harsh Kupwade Patil, Jefferson E Ricardini, and Marcos Vinicius M Silva. The unified butterfly effect: Efficient security credential management system for vehicular communications. In *2018 IEEE Vehicular Networking Conference (VNC)*, pages 1–8. IEEE, 2018.

[5] Emin Topalovic, Brennan Saeta, Lin-Shung Huang, Collin Jackson, and Dan Boneh. Towards short-lived certificates. *Web 2.0 Security and Privacy*, 2012.

[6] Maxim Raya, Daniel Jungels, Panos Papadimitratos, Imad Aad, and Jean-Pierre Hubaux. Certificate revocation in vehicular networks. *Laboratory for computer Communications and Applications (LCA) School of Computer and Communication Sciences, EPFL, Switzerland*, 2006.

[7] Internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile. url=https://tools.ietf.org/html/rfc5280, 2008.

[8] Adam Slagell, Rafael Bonilla, and William Yurcik. A survey of PKI components and scalability issues. In *2006 IEEE International Performance Computing and Communications Conference*, page 64. IEEE, 2006.

[9] Jason J Haas, Yih-Chun Hu, and Kenneth P Laberteaux. Design and analysis of a lightweight certificate revocation mechanism for vanet. In *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, pages 89–98. ACM, 2009.

[10] Maxim Raya, Panagiotis Papadimitratos, Imad Aad, Daniel Jungels, and Jean-Pierre Hubaux. Eviction of misbehaving and faulty nodes in vehicular networks. *IEEE Journal on Selected Areas in Communications*, 25(8), 2007.

[11] Giovanni Rigazzi, Andrea Tassi, Robert J Piechocki, Theo Tryfonas, and Andrew Nix. Optimized certificate revocation list distribution for secure v2x communications. In *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, pages 1–7. IEEE, 2017.

[12] Panagiotis Panos Papadimitratos, Ghita Mezzour, and Jean-Pierre Hubaux. Certificate revocation list distribution in vehicular communication systems. In *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*, pages 86–87. ACM, 2008.

[13] Kenneth P Laberteaux, Jason J Haas, and Yih-Chun Hu. Security certificate revocation list distribution for vanet. In *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*, pages 88–89. ACM, 2008.

[14] Noureddine Lasla, Mohamed Younis, Wassim Znaidi, and Dhafer Ben Arbia. Efficient distributed admission and revocation using blockchain for cooperative its. In *2018 9th IFIP international conference on new technologies, mobility and security (NTMS)*, pages 1–5. IEEE, 2018.

[15] Ao Lei, Yue Cao, Shihan Bao, Dasen Li, Philip Asuquo, Haitham Cruickshank, and Zhili Sun. A blockchain based certificate revocation scheme for vehicular communication systems. *Future Generation Computer Systems*, 2019.

[16] Maxim Raya and Jean-Pierre Hubaux. Securing vehicular ad hoc networks. *Journal of computer security*, 15(1):39–68, 2007.

[17] Maxim Raya and Jean-Pierre Hubaux. The security of vehicular ad hoc networks. In *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, pages 11–21. ACM, 2005.

[18] Bryan Parno and Adrian Perrig. Challenges in securing vehicular networks. In *Workshop on hot topics in networks (HotNets-IV)*, pages 1–6. Maryland, USA, 2005.

[19] Magda El Zarki, Sharad Mehrotra, Gene Tsudik, Nalini Venkatasubramanian, et al. Security issues in a future vehicular network. In *European Wireless*, volume 2, 2002.

[20] Giorgio Calandriello, Panos Papadimitratos, Jean-Pierre Hubaux, and Antonio Lioy. Efficient and robust pseudonymous authentication in vanet. In *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, pages 19–28. ACM, 2007.

[21] Panagiotis Papadimitratos, Levente Buttyan, Jean-Pierre Hubaux, Frank Kargl, Antonio Kung, and Maxim Raya. Architecture for secure and private vehicular communications. In *2007 7th International Conference on ITS Telecommunications*, pages 1–6. IEEE, 2007.

[22] Maxim Raya, Panos Papadimitratos, and Jean-Pierre Hubaux. Securing vehicular communications. *IEEE wireless communications*, 13(5):8–15, 2006.

[23] Eurostat. Criminal acts in europe 2008-2017. 2019.

[24] G Pasaoglu, D Fiorello, A Martino, G Scarcella, A Alemanno, A Zubaryeva, and C Thiel. Insécurité et délinquance 2018: premier bilan statistique. 2018.

[25] Eurostat. The french automotive industry, analysis and statistic. 2019.

[26] Yabing Liu, Will Tome, Liang Zhang, David Choffnes, Dave Levin, Bruce Maggs, Alan Mislove, Aaron Schulman, and Christo Wilson. An end-to-end measurement of certificate revocation in the web's pki. In *Proceedings of the 2015 Internet Measurement Conference*, pages 183–196. ACM, 2015.

[27] TS ETSI. 103 097-v1.3.1 (2017-10)-intelligent transport systems; security header and certificate formats.

[28] Veins. The open source vehicular network simulation framework.

[29] Omnet. Discrete event simulator.

[30] Sumo. Simulation of urban mobility.