A Vehicular DAA Scheme for Unlinkable ECDSA Pseudonyms in V2X

Christopher Hicks, Flavio D. Garcia
School of Computer Science, University of Birmingham, United Kingdom
{*c.hicks, f.garcia*}*@cs.bham.ac.uk*

***Abstract*—Vehicle-to-everything (V2X) communication is a broadcast messaging system intended to improve the efficiency and safety of connected and autonomous vehicles. In this paper we present a new V2X architecture and key management solution that reconciles the strong privacy guarantees of Direct Anonymous Attestation (DAA) with the efficiency, low-latency and accountability that is required for V2X. In contrast with the leading V2X standards, and uniquely in the literature, we prevent long-term vehicle pseudonym tracking despite dishonest and colluding certificate authorities and whilst retaining centralised authority over revocation. Our Vehicular DAA (VDAA) scheme includes a novel construction that optimally limits Sybil attacks by restricting each vehicle to one anonymous pseudonym-request per epoch. We present a new security model for VDAA and show that we can reduce the unforgeability and unlinkability of our Elliptic Curve Digital Signature Algorithm (ECDSA) broadcast messages to the security of the underlying DAA scheme.**

***Index Terms*—V2X, Attestation, Authentication**

1. Introduction

In the near future, vehicles will communicate directly between themselves and with roadside infrastructure. V2X communication, which includes Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) techniques, will drastically improve road safety and efficiency by enabling the next generation of semi-autonomous vehicle safety features such as platooning, collaborative forward collision warning and emergency electronic brake lights [1]. In V2X, vehicles cooperatively broadcast geospatial information to nearby peers using short Cooperative Awareness Messages (CAM) which are sent multiple times per second.

The development of V2X architectures is challenging because there are a number of conflicting requirements. In particular, vehicles are required to broadcast authenticated but unencrypted messages that specify their precise location, speed and heading [2]. At the same time, it is critically important that drivers are protected from the type of long-term tracking that threatens to uniquely identify individual behaviour. For example vehicle location data from Uber and Lyft has been misused for corporate espionage [3], to track important persons and to identify customers engaging in one-night stands [4]. Finally, vehicles must be accountable for the messages they broadcast and it must be possible for an authority to revoke the credentials of misbehaving vehicles.

For maximum impact on road safety V2X should be universally deployed [5]; Correspondingly, there are a number of international standardisation efforts. In particular, the European Telecommunications Standards Institute (ETSI) and the Institute of Electrical and Electronics Engineers (IEEE) are leading the development of V2X standards in Europe and the United States (U.S.), respectively. Both ETSI and the U.S. Department of Transportation (USDOT) have proposed national V2X architectures based on the IEEE Wireless Access in Vehicular Environments (WAVE) suite of standards, pseudonymous Elliptic Curve Digital Signature Algorithm (ECDSA) digital signatures and a hierarchical Public Key Infrastructure (PKI) that manages trust between different road users. ECDSA is well suited to V2X because it offers small signature sizes and low-latency message verification [6], but it is also less flexible than other signature schemes [7] (e.g. Schnorr [8]). In particular, the inability to re-randomise an ECDSA signature makes it impossible to strongly protect the privacy of vehicles that request pseudonyms by presenting a long-term credential.

Whilst the current standards offer some privacy protection from honest-but-curious [9] certificate authorities, they do not protect road users from authorities which are dishonest or that collaborate. Indeed, the European Data Protection Working Party have identified the need for new techniques that adequately protect the privacy of V2X-enabled vehicles from corrupt certificate authorities [10]. One promising technique that paradoxically provides both anonymity and accountability is Direct Anonymous Attestation (DAA), an anonymous group signature scheme that is typically used to attest to the state of a device based on a secure hardware root of trust. In DAA each user platform is enhanced with a Trusted Platform Module (TPM) which isolates a cryptographic key and that, together with the host platform, provides remote authentication in a privacy-preserving way. Unlike with ECDSA, users are able to receive a blind signature on their long-term credential and then authenticate anonymously as a group member. DAA offers strong privacy guarantees that include unforgeability, non-frameability and unlinkability, all of which are desirable for V2X and are maintained despite certificate authority corruption.

Whilst the strong privacy attributes of DAA make it an attractive candidate for use in V2X, the computational costs, large signature sizes and the risk of anonymous credential abuse prohibit its straightforward application. In this paper we seek to have the best of both worlds and reconcile the strong privacy guarantees of DAA with the efficiency, small signature size and standards-compliance of ECDSA broadcast messages.

Our Contribution

- We present our new VDAA architecture which harmonises the strong privacy properties of DAA with the low-latency, small signature size and standards-compliance of ECDSA signatures for V2X broadcast messages. In contrast to the latest standards and uniquely in the literature, VDAA prevents long-term vehicle pseudonym tracking despite dishonest certificate authorities and whilst retaining centralised authority over revocation.
- We introduce a novel construction that optimally limits Sybil attacks by restricting each vehicle to a single anonymous pseudonym request per epoch. Vehicles that attempt to retrieve multiple pseudonyms for a single epoch are denied, forfeit unlinkability and may optionally have their long-term credentials revoked.
- We model the VDAA architecture and formalise its security and privacy notions. We provide a reduction from the unforgeability and unlinkability of our scheme to the properties of the underlying DAA and ECDSA algorithms.

Related Work

Both the European ETSI [11] and USDOT standards [12] for V2X use ECDSA pseudonym certificates as the primary mechanism for providing vehicle privacy. Pseudonyms allow vehicles to send messages without revealing their identity, whilst still remaining accountable. In addition, both standards combine pseudonyms with a role-separated PKI to provide some limited privacy against honest, non-colluding authorities. Modern strategies for provisioning and changing pseudonyms in V2X are comprehensively surveyed by Petit et al. [13] and also by ETSI in their recent pre-standardisation study [14]. Whilst both standards share a common IEEE WAVE standard [15] for V2X message transmission and have comparable PKI architectures, they differ in their precise instantiations. In particular the ETSI standard uses the comparatively minimal PKI developed in [16], while the USDOT standard uses the Security Credential Management System (SCMS) developed by Whyte et al. [12], [17].

The DAA signature scheme was first proposed by Brickell et al. [18] and has since been standardised by the Trusted Computing Group (TCG) who include it in their TPM specification [19]. The latest TPM 2.0 standard [19] uses the efficient Elliptic-Curve based DAA (ECDAA) implementation developed by Chen et al. [20]. A number of security issues in the TPM 2.0 standard are addressed by Camenisch et al. [21] who propose minimal changes that both fix the standard against all known attacks and which allow building DAA schemes that are secure in the Universal Composability (UC) framework. The drawback to DAA is that all of the current schemes suffer from highly inefficient revocation procedures which grow linearly in the size of the revocation list [22]. The standard method for revocation in DAA only operates under the assumption that the long-term TPM secret is compromised and discovered by the verifier [18]; However, Enhanced Privacy ID (EPID) [23], [24] is a DAA scheme that complements the standard with the addition of both signature and issuer-based revocation mechanisms.

Whitefield et al. [25] apply DAA to V2X using a decentralised approach that removes the need for a pseudonym authority and which makes each vehicle responsible for managing its own pseudonym certificates. The REWIRE V2X revocation protocol [26] uses trusted computing to enable revocation without pseudonym resolution and the OTOKEN protocol [27] enhances REWIRE using the results of symbolic protocol analysis. Most similar to our contribution, Förster et al. [28] propose the PUCA scheme which builds upon the ETSI PKI [16] and the REWIRE revocation protocol. Vehicles request pseudonyms using periodic n-times anonymous credentials [29], yet retain ECDSA signatures for V2X communication. In contrast to our proposal, these schemes all critically depend on having vehicle hosts which will properly forward revocation messages to the trusted platform. It is unclear whether this is tenable when considering that an important stimulus for revocation is that the vehicle host is compromised and therefore broadcasts fictitious traffic information. Chen et al. [30] propose a DAA-based V2X scheme that, whilst retaining centralised revocation, provides a mechanism for detecting vehicles that abuse their anonymity to send multiple messages relating to the same event. Unlike VDAA the scheme of Chen et al. does not use efficient standards-compliant ECDSA signatures on broadcast messages, incurs significant communication overheads and does not prevent Sybil attacks. Finally and in contrast to all of these proposals, VDAA maintains vehicle privacy even under the much weaker assumption that the certificate authorities collaborate and that the TPM is compromised.

Outline

For clarity the goal of this work is to address the need for systems which meet the standard requirements for V2X, whilst also protecting vehicles from corrupt or colluding certificate authorities. Essentially, we substitute the long-term ECDSA vehicle certificates used by the leading standards with DAA credentials. This approach still allows vehicles to request regular, standards-compliant ECDSA pseudonym certificates whilst additionally providing the unlinkability of these requests. We use signature-based DAA revocation, and lists linking pseudonym values to DAA signatures, to retain centralised revocation capabilities. We also introduce a new secret attribute in each DAA credential which prevents Sybil attacks. The full details of our scheme are first introduced in Section 6, followed by a formalisation of the standard security and privacy requirements in Section 7 and a reduction to the underlying DAA scheme in Section 8.

2. Requirements

The core security and privacy requirements for V2X which are converged upon in the main standards [12], [31] and the literature [17], [25], [25], [28], [30] are as follows:

Authentication Every vehicle must be able to determine the authenticity and integrity of each broadcast message.

Unlinkability The messages broadcast by one vehicle, using two different pseudonyms and during two non-overlapping periods, should be indistinguishable from messages that have been broadcast by two distinct vehicles during the same two epochs.

Corrupt CA Resistance The repercussions of certificate authority compromise or collusion should be minimised. In particular no dishonest, colluding subset of authorities should be able link together two or more non-overlapping vehicle pseudonym certificates.

Revocation It must be possible to remove vehicles from the scheme. Specifically, V2X requires two types of revocation

i. (Vehicle Based Revocation) It must be possible to revoke any vehicle based on its canonical registration information, for example when it is 'written off' by an insurer.

ii. (Signature Based Revocation) Given a (malicious) signed message it must be possible to revoke the vehicle that sent it, for example if a road user modified their vehicle to send misleading messages to other road users.

Sybil Resistance Related to malicious vehicles which send misleading messages, V2X schemes should limit the number of concurrently valid pseudonyms that can be used to sign messages. One road user may try to imitate many different vehicles in order to manipulate traffic. When allowing for small variations in local time between different vehicles, the optimal number of concurrent identities is two.

In addition there are a number of established performance requirements for V2X. Vehicles have relatively limited computational, bandwidth and storage capabilities and V2X schemes should be developed accordingly. In particular, vehicles require very low-latency message verification which has been defined as exceeding 1000 verification operations per second [32] and must also be able to sign 10 messages per second.

3. System and Threat Model

We adopt the ETSI standard PKI model [11] for V2X, shown in Figure 1, which comprises one or more vehicles, Enrolment Authorities (EAs), Authorisation Authorities (AAs) and Revocation Authorities (RAs). For simplicity and without loss of generality we consider a single EA which also assumes the role of RA, a single AA and an implicit Root Certificate Authority (RCA). The EA manages long-term enrolment credentials and the AA authorises vehicles to use a particular service by issuing pseudonym certificates. In VDAA, the standard vehicle On-Board Unit (UBU) and Trusted Element (TE) in the ETSI reference architecture are replaced by a host platform and a TPM, respectively.

Threat Model

For our formalisation of unlinkability (privacy), and in common with both the TPM [19] and V2X standards [11], [12], we make the necessary assumption that the vehicle host is honest. A compromised host can always send arbitrary privacy-compromising information. Beyond the vehicle host, we also make weak assumptions about every other entity and improve significantly on the latest standards and literature. Crucially we advance the standards by allowing for curious EA and AA certificate authorities which may also collaborate. In addition, we improve upon the closest works in the literature [25], [28] by allowing for a subverted TPM when considering unlinkability and yet also allowing for an uncooperative host when considering revocation.

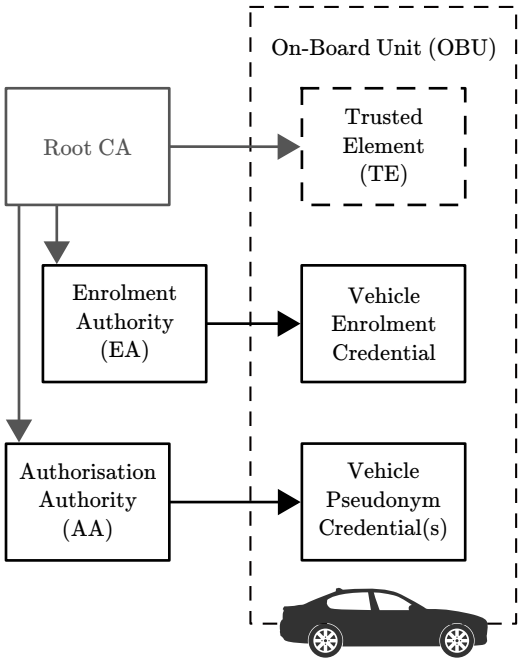

Figure 1: The ETSI standard PKI architecture.

For authentication (security) we require that the TPM is uncompromised but allow for a corrupted vehicle host. The EA and the AA must be trusted for authentication as they can register any compromised vehicle they desire, but we require that they cannot forge messages from any uncompromised TPM.

4. Preliminaries

This section introduces our notation and the cryptographic building blocks from which our scheme is developed. Specifically, we define the ECDSA and DAA signature schemes and the high-level TPM interface that are used in the construction of our scheme.

Notation

We use $x \leftarrow S$ to denote some x chosen uniformly at random from a set S. We let $|x|$ denote the bit size of x, let $x \parallel y$ express the concatenation of x and y and let $x \times G$ denote the scalar multiplication of point G by x. We distinguish between DAA and ECDSA public key pairs using the notation $(\mathsf{pk}, \mathsf{sk})$ and (P, x), respectively.

In addition, we let $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ denote groups of large prime order q and we let G, g_1, $\bar{g}$ and g_2 denote the generators such that $\mathbb{G}_1 = \langle G \rangle = \langle \bar{g} \rangle = \langle g_1 \rangle$ and

$\mathbb{G}_2 = \langle g_2 \rangle$. We let e be a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ such that:

- $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ is an efficiently computable homomorphism from $\mathbb{G}_2$ to $\mathbb{G}_1$ with $\psi(g_1) = g_2$.
- $\forall x \in \mathbb{G}_1, y \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_q$, $e(x^a, y^b) = e(x, y)^{ab}$.
- e is non-degenerate, in other words $e(g_1, g_2) \neq 1$.

For denoting signature proofs of knowledge of discrete logarithms, and signature proofs of the validity of statements about discrete logarithms, we use the standard notation introduced by Camenisch and Stadler [33]. For example, $\mathsf{SPK}[\alpha, \beta : \bar{y} = \bar{g}^{\alpha} \wedge y_1 = g_1^{\beta}](m)$ denotes the "signature proof of knowledge" upon m and of integers α and β such that $\bar{y} = \bar{g}^{\alpha}$ and $y_1 = g_1^{\beta}$ holds. To distinguish between proofs with TPM contribution and those without we use SPK^* and SPK, respectively. We use the notation $\mathsf{NIZK}[(w) : \text{statement}(w)](\mathsf{ctxt})$ from [21] to denote any non-interactive zero-knowledge proof that is bound to a context ctxt and proves knowledge of a witness w such that statement(w) is true.

Finally, in the formal security setting we use the term efficient to mean solvable using a probabilistic polynomial-time (PP) Turing machine with an error probability of less than $1/2$.

ECDSA Signature Scheme

In both the leading European and U.S. standards, and across much of the literature, ECDSA signatures are used to provide authentication and authorisation of the CAM broadcast by vehicles. Where $H_q : \{0,1\}^* \rightarrow \{0,1\}^{|q|}$ is a hash function, the ECDSA scheme comprises the following three algorithms:

DSAGen On input the security parameter 1^{η}, the algorithm selects secret key $x \leftarrow \mathbb{Z}_q^*$ and computes public key $P = x \times G$. The output is (P, x).

DSASign On input secret key x, the algorithm selects instance key $k \leftarrow \mathbb{Z}_q^*$ and computes curve point $R = (r_x, r_y) = k \times G$, element $r = r_x \bmod q$ and signature $s = k^{-1} \cdot (H_q(m) + x \cdot r) \bmod q$. The output is $\tau = (r, s)$.

DSAVerify On input the public key P, message m and signature $\tau = (r, s)$, the algorithm computes $w = s^{-1} \bmod q$, $u_1 = H_q(m) \cdot w \bmod q$, $u_2 = r \cdot w \bmod q$ and curve point $R' = (x_1, y_1) = u_1 \times G + u_2 \times P$. If $P, R' \in \langle G \rangle$, $r, s \in \mathbb{Z}_q^*$ and $r \equiv x_1 \bmod q$ then the output is **true** (accept) otherwise it is **false** (reject).

The provable security and known weaknesses of ECDSA is surveyed by Vaudenay [34]. In this work we assume that ECDSA is an EUF-CMA secure signature scheme as defined in the Appendix A.

DAA Formalisation

In our scheme, vehicles establish long-term DAA credentials which are used to request the short-lived ECDSA pseudonym certificates that authenticate each CAM. A DAA scheme, essentially an anonymous group signature scheme, entails a set of Issuers $\mathfrak{I}$, a set of signers $\mathfrak{S}$ and a set of verifiers $\mathfrak{V}$. Each signer $(\mathfrak{t}, \mathfrak{h}) \in \mathfrak{S}$ comprises a host platform $\mathfrak{h}$ and its TPM $\mathfrak{t}$. A DAA scheme $\mathcal{DAA}$ consists of the following five efficient algorithms and protocols:

Setup On input the security parameter 1^{η} the issuer $\mathfrak{i} \in \mathfrak{I}$ generates a random secret key isk, the group public key ipk and the public parameters par.

Join The signer $(\mathfrak{t}, \mathfrak{h}) \in \mathfrak{S}$ generates a secret key tsk on the TPM $\mathfrak{t}$ and then communicates with the issuer $\mathfrak{i} \in \mathfrak{I}$ to establish the DAA credential cre on the host $\mathfrak{h}$. cre optionally certifies a number of attributes $\mathsf{attr} = (a_1, \ldots, a_L)$.

Sign On input the TPM secret key tsk, basename bsn, message m and optionally attributes attr or verifier nonce n_v, the signer $(\mathfrak{t}, \mathfrak{h}) \in \mathfrak{S}$ outputs the DAA signature σ on m under $(\mathsf{tsk}, \mathsf{cre}, \mathsf{attr})$ associated with bsn.

Verify On input a message m, basename bsn, DAA signature σ and the signature revocation list $\mathsf{Sig\text{-}RL}$, the algorithm returns either **true** (accept) or **false** (reject).

Link On input two DAA signatures σ_a and σ_b, this algorithm returns either **linked** if the signatures have the same basename, **unlinked** or $\perp$ (**invalid**).

The basename bsn that is input to each DAA signature is used to provide user-controlled linkability, a key feature of DAA. In this work we seek unconditional unlinkability of the pseudonym certificates requested by each vehicle and so the same basename bsn will never be used twice and our DAA Link algorithm will always return false. For services other than cooperative awareness, such as automatic toll-road payments, the user may opt to re-use a specific basename when interacting with certain AAs.

TPM Interface

The assumption of trusted hardware on-board each vehicle is made by both of the leading V2X standards and minimises the attack surface of security-critical components. The TPM is an international standard for a hardware security chip that can be used to manage cryptographic keys and for remote attestation. TPMs provide a standard interface, which we detail here for completeness, that a host platform interacts with when executing the DAA protocol. Specifically the TPM has a fixed generator $\bar{g}$, two random oracles $H : \{0,1\}^* \rightarrow \mathbb{Z}_q$, $H_{\mathbb{G}_1} : \{0,1\}^* \rightarrow \mathbb{G}_1$, the set $\mathsf{Committed} = \emptyset$, counter $\mathsf{commitId} = 0$ and provides an interface to the following four algorithms:

TPM.Create Selects $\mathsf{tsk} \leftarrow \mathbb{Z}_q$, computes $\mathsf{tpk} = \bar{g}^{\mathsf{tsk}}$ and outputs the public key tpk. The private key tsk is stored.

TPM.Commit Takes as input secret key tsk, the signature basename bsn_L and the generator basename bsn_E. The algorithm computes the first part of the signing operation as follows:

1) If $\mathsf{bsn}_E \neq \perp$, set $\bar{g} \leftarrow H_{\mathbb{G}_1}(\mathsf{bsn}_E)$
2) Select $r \leftarrow \mathbb{Z}_q$, $n_t \leftarrow \{0,1\}^{\eta}$ and append $(\mathsf{commitId}, r, n_t)$ to $\mathsf{Committed}$.
3) Set $\bar{n}_t = H(\text{"}nonce\text{"}, n_t)$, $E \leftarrow \bar{g}^r$ and $K, L = \perp$.

4) If $\mathsf{bsn}_L \neq \perp$, set $j = H_{\mathbb{G}_1}(\mathsf{bsn}_L)$, $K = j^{\mathsf{tsk}}$ and $L = j^r$

The TPM outputs commitment $(\mathsf{commitId}, \bar{n}_t, E, K, L)$ and increments $\mathsf{commitId}$.

TPM.Hash Takes as input messages m_t and m_h. If $m_t \neq \perp$, the TPM checks if it wants to attest to m_t. The algorithm computes $c = H(\text{“}TPM\text{”}, m_t, m_h)$, the digest c is marked ‘safe to sign’ and the output is c.

TPM.Sign Takes as input $\mathsf{commitId}$, a digest c, a host nonce contribution n_h and completes the signing operation as follows:

1) Retrieve and remove $(\mathsf{commitId}, r, n_t)$ from $\mathsf{Committed}$.
2) Set $c' = H(\text{“}FS\text{”}, n_t \oplus n_h, c)$ and $s = r + c' \cdot \mathsf{tsk}$.

Upon completion of the $\mathsf{TPM.Sign}$ algorithm, the TPM outputs the nonce contribution n_t and the signature s.

5. VDAA Formalisation

The formal definition of a VDAA scheme which we use to both describe and evaluate our instantiation is as follows. With reference to DAA in Section 4 and the ETSI-standard PKI we present in Section 3, each vehicle $V_i = (\mathfrak{h}_i, \mathfrak{t}_i)$ is a DAA signer $V_i \in \mathfrak{S}$. The EA is both the DAA issuer $\mathfrak{i} \in \mathfrak{I}$ and the revocation manager, meanwhile the AA is the DAA verifier $\mathfrak{v} \in \mathfrak{V}$. The AA maintains a list of DAA signature and ECDSA pseudonym tuples $\mathsf{Auth\text{-}L}$ and a list of token serials $\mathsf{Ser\text{-}L}$. In addition, the EA manages three revocation lists: the vehicle revocation list $\mathsf{Pub\text{-}RL}$, the signature revocation list $\mathsf{Sig\text{-}RL}$ and the private-key revocation list $\mathsf{Priv\text{-}RL}$.

A VDAA scheme comprises three efficient algorithms Setup, Verify, Revoke and three protocols Join, Issue and Sign which are defined as follows:

Setup takes as input the security parameter 1^η. The EA outputs the DAA group public key pair $\mathsf{ik} = (\mathsf{ipk}, \mathsf{isk})$ and the global public parameters par which include $\mathsf{Pub\text{-}RL} = \emptyset$ and $\mathsf{Priv\text{-}RL} = \emptyset$.
The AA outputs the ECDSA public key pair $\mathsf{ak} = (P_{\text{AA}}, x_{\text{AA}})$, signature revocation list $\mathsf{Sig\text{-}RL} = \emptyset$, attestation list $\mathsf{Auth\text{-}L} = \emptyset$ and token serial list $\mathsf{Ser\text{-}L} = \emptyset$.

Join is run between the EA and a vehicle $V_i = (\mathfrak{h}_i, \mathfrak{t}_i)$. The EA is given the group public key pair $\mathsf{ik} = (\mathsf{ipk}, \mathsf{isk})$ and V_i is given ipk. Eventually, $\mathfrak{t}_i$ outputs a private key tsk. The host $\mathfrak{h}_i$ will output a secret key hsk, DAA credential cre and sybil secret s. A revoked vehicle will output nothing $\perp$.

Issue is run between the AA and vehicle $V_i = (\mathfrak{h}_i, \mathfrak{t}_i)$. The AA is given the group public key ipk, the private key x_{AA}, attestation list $\mathsf{Auth\text{-}L}$ and token serial list $\mathsf{Ser\text{-}L}$. The TPM $\mathfrak{t}_i$ is given the private key tsk and the host $\mathfrak{h}_i$ is given ipk, the private key hsk, the DAA credential cre, sybil secret s and epoch ep. Eventually, the AA will output the updated $\mathsf{Auth\text{-}L'}$, the updated $\mathsf{Ser\text{-}L'}$ and either the pseudonym signature τ or $\perp$ (vehicle revoked).

Sign is run by a vehicle V_i between the TPM $\mathfrak{t}_i$ and the host $\mathfrak{h}_i$. $\mathfrak{t}_i$ takes as input the private key $x_{\mathfrak{t}}$ and $\mathfrak{h}_i$ takes the private epoch key x_{ep}, message m and epoch ep. Eventually, V_i outputs the ECDSA signature $\tau = (r, s)$ on m with respect to the public key $P_{\mathsf{ep}} = (x_{\mathsf{ep}} \times x_{\mathfrak{t}} \times G)$.

Verify is run by a vehicle V_i, takes as input the ECDSA signed message (m, τ) and outputs either **true** (accept) or **false** (reject).

Revoke has three different implementations. For vehicle based revocation, the EA takes as input the vehicle public key vpk and outputs the updated vehicle revocation list $\mathsf{Pub\text{-}RL'}$. Signature based revocation is run between the AA and the EA. The AA takes as input the group public key ipk, signed message (m, τ), signature revocation list $\mathsf{Sig\text{-}RL}$ and the attestation list $\mathsf{Auth\text{-}L}$. The AA sends the corresponding DAA signature (m', σ) to the EA which outputs the updated $\mathsf{Sig\text{-}RL'}$. For private-key based revocation the EA takes as input ipk, vehicle private key vsk and outputs $\mathsf{Priv\text{-}RL'}$.

6. VDAA Scheme

This section presents the full details of our VDAA scheme. VDAA harmonises the strong privacy guarantees of DAA with the low-latency, small signature size and standards-compliance of ECDSA signatures. In VDAA, vehicles are fitted with a TPM and use DAA as the basis of their long-term enrolment. Uniquely in our scheme, the privacy of vehicles is preserved despite colluding certificate authorities and a subverted vehicle TPM. We maintain privacy under a very strong model in which only the vehicle host needs to be fully trusted. In addition, this is accomplished whilst retaining the centralised control over vehicle revocation that is necessary for V2X. To relax the requirements for clock synchronisation, we assume a globally defined pseudonym change policy that divides the future into a number of epoch periods ep and a global pseudonym overlap period T_{overlap} during which the pseudonym of both the current and next epoch is valid.

The intuition for our scheme is as follows. Every vehicle comprises a TPM and a host which jointly generate a split DAA key pair $\mathsf{vk} = (\mathsf{vpk}, \mathsf{vsk})$. Vehicles join the scheme by obtaining a partially blind DAA signature $\mathsf{cre} = \mathsf{PBSIG}(\mathsf{isk}, \mathsf{vpk})$ on the split public key vpk from the EA. To obtain ECDSA pseudonym certificates, vehicles make anonymous requests for each epoch ep by using the DAA algorithm to authenticate to the AA. To prevent the abuse of anonymous DAA credentials, each request includes a unique serial token ser. Each serial token is derived from a Sybril secret s that is unique to each vehicle and the requested pseudonym epoch ep. Serial tokens prevent Sybil attacks as any vehicle that makes multiple requests for pseudonyms in the same epoch is forced to do so with the same token and therefore forfeits unlinkability and can be denied additional credentials. The AA maintains a list of DAA signature and ECDSA pseudonym tuples, $\mathsf{Auth\text{-}L}$, which enables vehicles that send malicious messages to be removed by denying them new credentials in the future. Broadcast message signing and verification are just the standard ECDSA operations from Section 4, which both maintains the performance that is necessary for safety-critical V2X applications and

ensures that a subverted TPM cannot compromise the privacy of the vehicle.

The VDAA scheme consists of 3 algorithms and 3 protocols. The Setup algorithm is run once by the EA and the AA to generate the scheme public and private parameters. The Join protocol is typically executed only once for each vehicle that joins the scheme and the Issue protocol is run each time a vehicle requires a pseudonym certificate for a particular epoch. Vehicles sign and verify broadcast messages using the ECDSA Sign and Verify algorithms, respectively, and the Revocation protocol is run when removing misbehaving vehicles. Whilst VDAA can be instantiated using either a LRSW [35] or q-Strong Diffie Hellman (q-SDH) [36] based DAA scheme, in the remainder of this section and our analysis we focus on the q-SDH based scheme of Camenisch et al. [21]. The q-SDH DAA scheme we use has a more efficient attribute certification mechanism which we use to prevent Sybil attacks.

Setup

The VDAA setup algorithm is run once to initialise the parameters of the scheme. On input the security parameter 1^η, the EA selects the group public key pair $\mathsf{ik} = (\mathsf{ipk}, \mathsf{isk})$ and the public parameters par which comprises ipk, the vehicle revocation list $\mathsf{Pub\text{-}RL} = \emptyset$ and the private-key revocation list $\mathsf{Priv\text{-}RL} = \emptyset$. Specifically, ik is a BBS+ signature scheme [37] key pair which is generated as follows:

1) Choose uniformly at random generator $h \leftarrow \mathbb{G}_1$ and the private key $x \leftarrow \mathbb{Z}_q$.
2) Set $X = g_2^x$ and $X' = g_1^x$.
3) Prove $\pi_{\mathsf{ipk}} = \mathsf{SPK}[x : X = g_2^x \wedge X' = g_1^x](\text{“setup”})$.
4) Let $\mathsf{ipk} = (h, X, X', \pi_{\mathsf{ipk}})$ and $\mathsf{isk} = x$.

The AA selects the ECDSA public key pair $\mathsf{ak} = (P_{\mathrm{AA}}, x_{\mathrm{AA}})$ and creates the signature revocation list $\mathsf{Sig\text{-}RL} = \emptyset$, the attestation list $\mathsf{Auth\text{-}L} = \emptyset$ and the serial token list $\mathsf{Ser\text{-}L} = \emptyset$. In particular, ak is an ECDSA key pair that is output by the DSAGen algorithm defined in Section 4.

Join

The first step of VDAA is the Join protocol, shown in Figure 2, during which a vehicle joins the scheme for the first time. Our Join protocol is based on the DAA Join protocol of Camenisch et al. [21], which we adapt to include our Sybil attack resistance mechanism and revocation capabilities. For simplicity we assume that the vehicle host manufacturer is also the EA and so can be certain that it is executing the protocol with a genuine TPM. The Join protocol can also be run after the vehicle host has been shipped, for which we assume that a certified endorsement key is installed and that the corresponding certificate is available to the EA. The EA takes as input the group public key pair $\mathsf{ik} = (\mathsf{ipk}, \mathsf{isk})$ and the public and private revocation lists Pub-RL and Priv-RL, respectively.

The vehicle takes as input ipk and then the Join protocol is as follows:

1) The vehicle host requests to join the VDAA group and the EA responds with a nonce $n \leftarrow \{0,1\}^{|n|}$ for freshness.
2) The vehicle host requests the TPM to create a new DAA key pair. The TPM selects the DAA key pair $\mathsf{tk} = (\mathsf{tpk}, \mathsf{tsk})$, stores the private key tsk and sends the public key tpk to the host.
3) The vehicle host forwards the nonce n to the TPM and then requests the split vehicle key contribution $\mathsf{tpk} = \bar{g}^{\mathsf{tsk}}$ and the proof $\pi_{\mathsf{tpk}} = \mathsf{SPK}^*[\mathsf{tsk} : \mathsf{tpk} = \bar{g}^{\mathsf{tsk}}](\text{“join”}, n)$ which asserts that:
   i. The TPM has the private key tsk corresponding to the public key tpk.
   ii. The TPM generated the split vehicle key contribution $\mathsf{tpk} = \bar{g}^{\mathsf{tsk}}$ such that it corresponds to tsk.

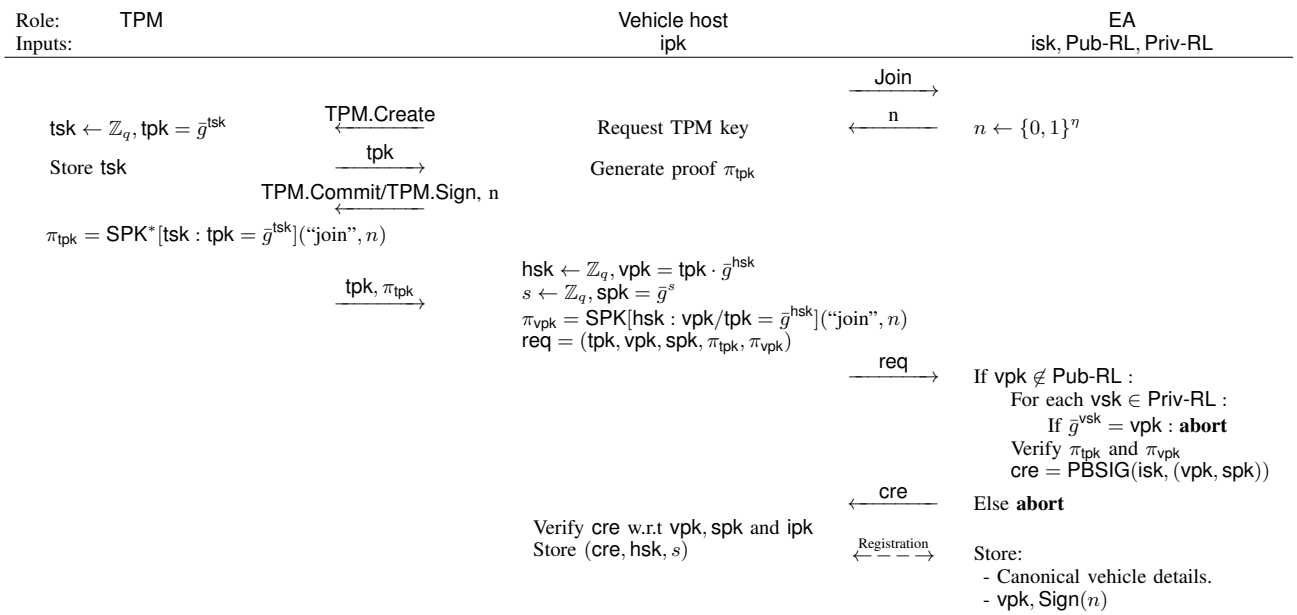

Figure 2: The VDAA Join protocol.

The TPM computes tpk, the proof π_{tpk} and sends them to the vehicle host.

4) The vehicle host selects the split key contribution $\mathsf{hsk} \leftarrow \mathbb{Z}_q$, computes the public key $\mathsf{vpk} = \mathsf{tpk} \cdot \bar{g}^{\mathsf{hsk}}$ and the proof $\pi_{\mathsf{vpk}} = \mathsf{SPK}[\mathsf{hsk} : \mathsf{vpk}/\mathsf{tpk} = \bar{g}^{\mathsf{hsk}}](\text{“join”}, n)$ which asserts that vpk is a signature proof of knowledge SPK on n. The vehicle host also selects the Sybil secret s and computes the public key $\mathsf{spk} = \bar{g}^s$ which is included in the request for group membership. The vehicle host sends $\mathsf{tpk}, \mathsf{vpk}, \mathsf{spk}, \pi_{\mathsf{tpk}}$ and π_{vpk} to the EA.
5) The EA verifies that vpk is not in the vehicle revocation list $\mathsf{Pub\text{-}RL}$ and that it does not correspond to any revoked private key in $\mathsf{Priv\text{-}RL}$. Next, the EA verifies the proofs $\pi_{\mathsf{tpk}}, \pi_{\mathsf{vpk}}$ and then computes the membership credential cre using a partially blind signature PBSign that certifies vsk by signing vpk. The resulting DAA credential $\mathsf{cre} = \mathsf{PBSign}(\mathsf{isk}, (\mathsf{vpk}, \mathsf{spk}))$ is sent to the vehicle host. In particular, cre is a blindly-signed BBS+ signature on the message (vsk, s) which is computed as follows:
   i. Choose $(e, r) \leftarrow \mathbb{Z}_q^2$.
   ii. Compute $A = (g_1 \cdot h^r \cdot \mathsf{vpk} \cdot \mathsf{spk})^{\frac{1}{e+x}}$.
   iii. Set $\mathsf{cre} = (A, e, r)$.
6) The vehicle host verifies the DAA credential cre with respect to vpk, the Sybil public attribute spk and the group public key ipk. Specifically, the vehicle host computes $b = g_1 \cdot h^r \cdot \mathsf{vpk} \cdot \mathsf{spk}$ and checks that $e(A, X \cdot g_2^e) = e(b, g_2)$. The vehicle host stores $\mathsf{cre}' = ((A, e, r), b)$, the host stores the secret key hsk and the Sybil secret s.

Issue

The VDAA Issue protocol, shown in Figure 3, is run each time that a vehicle requires a signed pseudonym certificate for a particular epoch ep. Initially, the TPM has the private key tsk and the vehicle host has the DAA credential cre, the secret key hsk, the Sybil secret s, the AA group public key ipk, an epoch ep and the signature revocation list $\mathsf{Sig\text{-}RL}$. The AA has the ECDSA private key x_{AA}, ipk, $\mathsf{Sig\text{-}RL}$, the attestation list $\mathsf{Auth\text{-}L}$ and the token serial list $\mathsf{Ser\text{-}L}$. The Issue protocol is as follows:

1) The vehicle host selects a random epoch key x_{ep} and then computes the pseudonym public key $P_{\mathsf{ep}} = x_{\mathsf{ep}} \times G$ and the serial token $\mathsf{ser}_{s,\mathsf{ep}} = H_{\mathbb{G}_1}(1 \parallel \mathsf{ep})^s$.
2) The vehicle host and TPM jointly compute the signature revocation token $\mathsf{rev} = H_{\mathbb{G}_1}(1 \parallel \mathsf{bsn})^{\mathsf{vsk}}$, the proof of membership credential π_{cre} and, for each tuple $(\mathsf{bsn}_i, \mathsf{rev}_i)$ in $\mathsf{Sig\text{-}RL}$, the non-revocation proof $\pi_{\mathsf{Sig\text{-}RL},i}$. The proof of membership credential π_{cre} is computed using following method of Camenisch et al. [21] which is zero-knowledge even if the TPM is subverted:
   i. The vehicle host re-randomises the BBS+ credential $\mathsf{cre}' = ((A, e, r), b)$ established in the Join protocol. The vehicle host chooses $q_1 \leftarrow \mathbb{Z}_q^\star, q_2 \leftarrow \mathbb{Z}_q, q_3 \leftarrow \frac{1}{q_1}$, sets $A' = A^{q_1}, \bar{A} = A'^{-e} \cdot b^{q_1}, b' = b^{q_1} \cdot h^{-q_2}$ and $r' = r - q_2 \cdot q_3$. The re-randomised credential is $\mathsf{cre}' = (\bar{A}, A', b')$.
   ii. The vehicle host and TPM jointly compute the

Role:	TPM	Vehicle host	AA
Inputs:	tsk	$(\mathsf{cre}, \mathsf{hsk}, s), \mathsf{ipk}, \mathsf{ep}, \mathsf{Sig\text{-}RL}$	$x_{\text{AA}}, \mathsf{ipk}, \mathsf{Sig\text{-}RL}, \mathsf{Auth\text{-}L}, \mathsf{Ser\text{-}L}$
Sign:		$\mathsf{bsn} = \perp$ $x_{\mathsf{ep}} \leftarrow \mathbb{Z}_q^*, P_{\mathsf{ep}} = x_{\mathsf{ep}} \times G$ $\mathsf{ser}_{s,\mathsf{ep}} = H_{\mathbb{G}_1}(1 \parallel \mathsf{ep})^s$ $\mathsf{rev} = H_{\mathbb{G}_1}(1 \parallel \mathsf{bsn})^{\mathsf{vsk}}$ $\pi_{\mathsf{cre}} = \mathsf{NIZK}^*[(\mathsf{vsk}, s, \mathsf{cre}) : \mathsf{ser}_{s,\mathsf{ep}} = H_{\mathbb{G}_1}(1 \parallel \mathsf{ep})^s$ $\wedge\ \mathsf{rev} = H_{\mathbb{G}_1}(1 \parallel \mathsf{bsn})^{\mathsf{vsk}}$ $\wedge\ 1 = \mathsf{PBVf}(\mathsf{ipk}, \mathsf{cre}, \mathsf{vsk})](\text{“sign”}, (P_{\mathsf{ep}}, \mathsf{ep}), \mathsf{Sig\text{-}RL}))$ For each $(\mathsf{bsn}_i, \mathsf{rev}_i) \in \mathsf{Sig\text{-}RL}$: $\pi_{\mathsf{Sig\text{-}RL},i} = \mathsf{SPK}^*[\mathsf{vsk} : H_{\mathbb{G}_1}(1 \parallel \mathsf{bsn}_i)^{\mathsf{vsk}} \neq \mathsf{rev}_i \wedge \mathsf{rev} = H_{\mathbb{G}_1}(1 \parallel \mathsf{bsn}_i)^{\mathsf{vsk}}](\text{“sign”})$ $\sigma = (\mathsf{ser}_{s,\mathsf{ep}}, (\mathsf{bsn}, \mathsf{rev}), \pi_{\mathsf{cre}}, \{\pi_{\mathsf{Sig\text{-}RL},i}\})$	
		$\xrightarrow{(\sigma, P_{\mathsf{ep}} \parallel \mathsf{ep})}$	
Verify:			Parse $\sigma = (\mathsf{ser}_{s,\mathsf{ep}}, (\mathsf{bsn}, \mathsf{rev}), \pi_{\mathsf{cre}}, \{\pi_{\mathsf{Sig\text{-}RL},i}\})$ Verify $\pi_{\mathsf{cre}}, \pi_{\mathsf{Sig\text{-}RL},i}$ w.r.t $\mathsf{ipk}, m, \mathsf{Sig\text{-}RL}$ For each $\mathsf{ser} \in \mathsf{Ser\text{-}L}$: If $\mathsf{ser}_{s,\mathsf{ep}} = \mathsf{ser}_{s,\mathsf{ep}}$: **abort** Add $\mathsf{ser}_{s,\mathsf{ep}}$ to $\mathsf{Ser\text{-}L}$, Add σ to $\mathsf{Auth\text{-}L}$ For each $\mathsf{vsk} \in \mathsf{Priv\text{-}RL}$: If $H_{\mathbb{G}_1}(1 \parallel \mathsf{bsn})^{\mathsf{vsk}} = \mathsf{rev}$: **abort**
DSASign :		$\xleftarrow{\tau}$	$\tau = \mathsf{DSASign}(x_{\text{AA}}, P_{\mathsf{ep}} \parallel \mathsf{ep})$
DSAVerify :		If $\mathsf{DSAVerify}(P_{\mathsf{ep}}, (P_{\mathsf{ep}} \parallel \mathsf{ep}), \tau) \neq$ **true** : **abort** Store $(\mathsf{ep}, x_{\mathsf{ep}}, \tau)$	

Figure 3: The VDAA Issue protocol.

proof of membership credential π_{cre}:

$$\begin{aligned}\pi_{\mathsf{cre}} = &\mathsf{SPK}^{\star}\{(\mathsf{vsk}, s, e, q_2, q_3, s') : \\ &g_1^{-1} \cdot \bar{g}^{-s} = b'^{-q_3} \cdot h^{r'} \cdot \mathsf{vpk} \cdot \mathsf{spk} \ \wedge \\ &\mathsf{rev} = H_{\mathbb{G}_1}(1 \parallel \mathsf{bsn})^{\mathsf{vsk}} \ \wedge \\ &\bar{A} \cdot b'^{-1} = A'^{-e}\}(\text{“sign”}, \mathsf{Sig\text{-}RL}))\end{aligned}$$

The final DAA signature is $\sigma = (\mathsf{ser}_{s,\mathsf{ep}}, (\mathsf{bsn}, \mathsf{rev}), \mathsf{cre}', \pi_{\mathsf{cre}}, \{\pi_{\mathsf{Sig\text{-}RL},i}\})$. The vehicle host sends $(\sigma, P_{\mathsf{ep}} \parallel \mathsf{ep})$ to the AA.

3) The AA parses the DAA signature σ and verifies the proofs π_{cre} and $\{\pi_{\mathsf{Sig\text{-}RL},i}\}$ with respect to the group public key ipk, the message m and the revocation list Sig-RL. In particular, the AA checks that $A' \neq 1$ and $e(A', X) = e(\bar{A}, g_2)$ with respect to the randomised DAA credential $\mathsf{cre}' = (\bar{A}, A', b')$ and $\mathsf{ipk} = (h, X, X', \pi_{\mathsf{ipk}})$. The AA also ensures that the serial token $\mathsf{ser}_{s,\mathsf{ep}}$ is novel and that vsk has not been revoked. The DAA signature and ECDSA pseudonym tuple $(\sigma, P_{\mathsf{ep}} \parallel \mathsf{ep})$ is added to the attestation list Auth-L, the serial token $\mathsf{ser}_{s,\mathsf{ep}}$ is added to Ser-L and the host is sent the ECDSA signature τ on the requested pseudonym public key and epoch $(P_{\mathsf{ep}} \parallel \mathsf{ep})$.
4) The vehicle host verifies the AA signature τ on $(P_{\mathsf{ep}} \parallel \mathsf{ep})$ and then creates a record that links the epoch ep with the signature τ and the signing key x_{ep}.

Sign

The VDAA Sign algorithm is run each time that a vehicle host signs a broadcast message. The Sign algorithm is simply the standard ECDSA signing algorithm from Section 4, and is run by the vehicle host only. Since the TPM does not take part, it is unable to compromise the privacy of the vehicle host.

To sign a message m, the vehicle host has the ECDSA signing key x_{ep} for the current epoch ep in which the signature should be valid. The vehicle host runs the DSASign algorithm from Section 4 which computes the following:

1) Choose an instance key $k \leftarrow \mathbb{Z}_q^{\star}$.
2) Compute the instance curve point $R = (r_x, r_y) = k \times G$ and the integer $r = r_x \mod q$.
3) Compute $s = k^{-1} \cdot (H_q(m) + x_{\mathsf{ep}} \cdot r) \mod q$.
4) The signature on m is $\tau = (r, s)$.

The signing could be split between the vehicle host and the TPM using the IFAL public-key derivation technique of Verheul et al. [38], however the TPM would be able to compromise the privacy of the vehicle by choosing bad instance keys. An alternative technique that would allow the signing to be split between the vehicle host and the untrusted TPM is the efficient two-party ECDSA signing protocol of Lindell [7]. This would have the advantage of requiring the involvement of the TPM for creating broadcast message signatures. However, in VDAA since the TPM is necessary to request a signed pseudonym certificate, there is little to be gained unless the epoch periods are very long. In addition, two-party ECDSA is far more computationally expensive than the single party case.

Verify

The Verify algorithm, which is used to verify every broadcast CAM that is received by a vehicle, is unchanged from the standard ECDSA verification algorithm in Section 4. The vehicle simply runs the DSAVerify algorithm which takes as input the pseudonym public key P_{ep}, the signed message tuple (m, τ) and outputs either **true** (accept) or **false** (reject). For every unique pseudonym public key P_{ep} that is used to authorise a received message, the vehicle additionally verifies that there is a signature τ_{ep} on P_{ep} which is valid with respect to the AA public key P_{AA}.

Revocation

In VDAA there are three different mechanisms for revocation and correspondingly, three different protocols. VDAA supports identity-based, message-based and private-key based revocation with the following three protocols

Identity-based revocation The identity-based revocation protocol is initiated when the EA is provided with the canonical registration information of a vehicle that should be removed from the scheme. The EA has the vehicle identity and then looks up the corresponding public key vpk and signature σ that were provided during the Issue protocol. The EA adds vpk to the vehicle revocation list Pub-RL and sends σ to the AA. The AA adds σ to the signature revocation list Sig-RL.

Message-based revocation It is critical for V2X that dishonest vehicles which send false information can be removed from participation. In message-based revocation the AA is provided with a signature τ on a message m, the attestation list Auth-L and Sig-RL. The AA uses Auth-L to identify the DAA signature σ that was used to request the pseudonym P_{ep} w.r.t τ in the Issue protocol. The AA adds σ to Sig-RL. When requesting new pseudonyms, vehicles prove in zero knowledge that they did not create any of the signatures in Sig-RL; Vehicles that have been revoked are denied future pseudonym signature requests.

Private-key revocation The final revocation mechanism is based on a compromised vehicle private key that has been discovered. Both the EA and the AA take the compromised private key as input. In the Join protocol, the EA checks that vpk does not correspond to any revoked vsk; If vsk is revoked, then the corresponding vpk is added to Pub-RL. In the Issue protocol, the revocation tuple $(\mathsf{bsn}, \mathsf{rev} = \mathbb{H}_{\mathbb{G}_1}(1 \parallel \mathsf{bsn})^{\mathsf{vsk}})$ is used to check that a vehicle is not using the revoked private key vsk. Vehicles with revoked private keys are denied at both the Join and Issue stages of the VDAA scheme.

7. Formal Security & Privacy Requirements

In this section we formalise the security and privacy of a VDAA scheme based on the V2X requirements in Section 2.

7.1. Security

Intuitively we require that, provided all vehicle TPMs are uncorrupted, no adversary should be able to create a valid signature on any V2X broadcast message. We capture this requirement by defining the unforgeability game **Forge-Game**, which is played between an efficient adversary $\mathcal{A}$ and a challenger $\mathcal{C}$, as follows:

Forge-Game$_{\mathcal{C}}(1^{\eta}, \mathcal{A})$:

1) The challenger $\mathcal{C}$ simulates the $\mathsf{Setup}(1^{\eta})$ algorithm which outputs the EA and AA public key pairs $\mathsf{ik} = (\mathsf{ipk}, \mathsf{isk})$, $\mathsf{ak} = (P_{\mathrm{AA}}, x_{\mathrm{AA}})$ and the parameters par. $\mathcal{C}$ simulates N_V vehicles with identities $\{V_1, \ldots, V_{N_V}\}$ and also simulates the EA and the AA including the Join and Issue protocols. Finally, $\mathcal{C}$ provides the adversary $\mathcal{A}$ with the public parameters $(\mathsf{ipk}, \mathsf{par}, P_{\mathrm{AA}})$ and a reference to each vehicle $V_i \in \{V_1, \ldots, V_{N_V}\}$.
2) The challenger $\mathcal{C}$ simulates each vehicle $V_i \in \{V_1, \ldots, V_{N_V}\}$ by selecting the vehicle secret key vsk_i, the Sybil secret s_i and by simulating the Join protocol and the EA so that each vehicle V_i has the DAA credential cre_i on vsk_i and s_i.
3) Challenge: Polynomially many times, adversary $\mathcal{A}$ requests challenger $\mathcal{C}$ to sign a message m in epoch ep on behalf of vehicle V_i. $\mathcal{C}$ simulates the vehicle V_i and then:
   i. If V_i does not have the epoch key x_{ep} then $\mathcal{C}$ simulates the Issue protocol by selecting the random epoch key x_{ep} and then computing the ECDSA signature τ_i on the pseudonym public key $P_{\mathsf{ep},i}$ with respect to the AA public key P_{AA}.
   ii. $\mathcal{C}$ simulates the $\mathsf{DSASign}$ algorithm and provides $\mathcal{A}$ with the ECDSA signature τ on message m with respect to pseudonym $P_{\mathsf{ep},i}$.
4) Output: The adversary $\mathcal{A}$ outputs the ECDSA signature $\tau^{\star}$, the message $m^{\star}$, the pseudonym public key $P_{\mathsf{ep}}{}^{\star}$ and the AA signature $\tau_{\mathsf{ep}}{}^{\star}$.

An adversary $\mathcal{A}$ wins the unforgeability game if:

1) $\mathsf{DSAVerify}(P_{\mathsf{ep}}{}^{\star}, m^{\star}, \tau^{\star}) =$ **true**.
2) $\mathsf{DSAVerify}(P_{\mathrm{AA}}, P_{\mathsf{ep}}{}^{\star}, \tau_{\mathsf{ep}}{}^{\star}) =$ **true**.
3) The message $m^{\star}$ does not correspond to any query made by the adversary $\mathcal{A}$ to the challenger $\mathcal{C}$.

Definition 1. Let $\mathcal{A}$ denote an adversary that plays the **Forge-Game**. We denote by $\mathbf{Adv}[\mathcal{A}^{\mathrm{forge}}_{\mathsf{VDAA}}] = \Pr[\mathcal{A}\text{ wins}]$ the advantage with which the adversary $\mathcal{A}$ breaks the unforgeability game. We say that a VDAA scheme is unforgeable if for all efficient adversaries $\mathcal{A}$, $\mathbf{Adv}[\mathcal{A}^{\mathrm{forge}}_{\mathsf{VDAA}}]$ is negligible.

7.2. Privacy

Informally, one vehicle that signs V2X messages under two different pseudonyms during two non-overlapping epochs should be indistinguishable from two distinct vehicles that sign the same set of messages. We capture this requirement by defining the unlinkability game **Priv-Game**, which is played between an efficient adversary $\mathcal{A}$ and a challenger $\mathcal{C}$, as follows:

Priv-Game$_{\mathcal{C}}(1^{\eta}, \mathcal{A})$:

1) The challenger $\mathcal{C}$ simulates the $\mathsf{Setup}(1^{\eta})$ algorithm and provides the adversary $\mathcal{A}$ with the resulting $\mathsf{ik} = (\mathsf{ipk}, \mathsf{isk})$, $\mathsf{ak} = (P_{\mathrm{AA}}, x_{\mathrm{AA}})$ and the parameters par. $\mathcal{C}$ also simulates 2 vehicles with identities V_0 and V_1.
2) Let the number of vehicles $N_V = 2$, then this step is the same as in the **Forge-Game**.
3) Adversary $\mathcal{A}$ selects two distinct epochs $\mathsf{ep}0, \mathsf{ep}1$ and submits them to $\mathcal{C}$.
4) Challenger $\mathcal{C}$ flips a bit $b \leftarrow \{0, 1\}$. For each vehicle $V_i \in (V_b, V_{b-1})$, $\mathcal{C}$ simulates V_i and selects two distinct epoch keys $x_{\mathsf{ep}0,i}$ and $x_{\mathsf{ep}1,i}$. For each corresponding pseudonym public key $P_{\mathsf{ep}0,i}$ and $P_{\mathsf{ep}1,i}$, $\mathcal{C}$ simulates the Issue protocol with the adversary $\mathcal{A}$ who simulates the AA. $\mathcal{C}$ acquires the ECDSA signatures $\tau_{\mathsf{ep}0,i}, \tau_{\mathsf{ep}1,i}$ with respect to the AA public key P_{AA} on $P_{\mathsf{ep}0,i}$ and $P_{\mathsf{ep}1,i}$.
5) Challenge: Polynomially many times, the adversary $\mathcal{A}$ requests the challenger $\mathcal{C}$ to sign a message m during epoch ep.
   - If $\mathsf{ep} \notin \{\mathsf{ep}0, \mathsf{ep}1\}$ the challenger $\mathcal{C}$ outputs $\perp$ (invalid epochs).
   - If $b = 0$, $\mathcal{C}$ simulates vehicle V_0, simulates the $\mathsf{DSASign}$ algorithm and outputs the signed message (τ, m) with respect to pseudonym $P_{\mathsf{ep}0,0}$.
   - If $b = 1$ and $\mathsf{ep} = \mathsf{ep}0$ then $\mathcal{C}$ simulates vehicle V_0. If $\mathsf{ep} = \mathsf{ep}1$ then $\mathcal{C}$ simulates V_1. $\mathcal{C}$ outputs the signed message (τ, m) with respect to pseudonym public key $P_{\mathsf{ep}0,0}$ or $P_{\mathsf{ep}1,1}$, respectively.
6) Output: The adversary $\mathcal{A}$ outputs a bit $b' \in \{0, 1\}$ indicating its guess of b.

An adversary $\mathcal{A}$ wins the unlinkability game if $b = b'$.

Definition 2. Let $\mathcal{A}$ denote an adversary that plays the **Priv-Game**. We denote by $\mathbf{Adv}[\mathcal{A}^{\mathrm{link}}_{\mathsf{VDAA}}] = |\Pr[b' = b] - \frac{1}{2}|$ the advantage with which the adversary $\mathcal{A}$ breaks the unlinkability game. We say that a VDAA scheme is unlinkable if for all efficient adversaries $\mathcal{A}$, the advantage $\mathbf{Adv}[\mathcal{A}^{\mathrm{link}}_{\mathsf{VDAA}}]$ is negligible.

8. The Security and Privacy of VDAA

This section shows that our VDAA scheme is secure with respect to the definitions presented in Section 7.

8.1. Unforgeability

We show that if the underlying DAA and ECDSA signature schemes are unforgeable and EUF-CMA secure (See Appendix A), respectively, then our VDAA scheme is secure with respect to Definition 1. Informally, an adversary cannot forge a signature on a message because the underlying DAA scheme has the property of unforgeability; This means that no adversary can use a DAA credential from a vehicle with an honest TPM and consequently, no adversary can obtain an ECDSA pseudonym using the DAA credential of an honest vehicle. Provided DAA has unforgeability, the only way an adversary can forge a signature is if they can break

the underlying ECDSA signature scheme.

Theorem 1. Let DAA be a secure DAA scheme with respect to the ideal functionality $\mathcal{F}_{\text{pdaa+}}$ defined by Camenisch et al. [21] and let ECDSA be an EUF-CMA secure [39] signature scheme, then the VDAA scheme we present in Section 6 is secure with respect to unforgeability as defined in Section 7.1.

Proof. Assume for contradiction that our VDAA scheme is not unforegeable. This means that there is an adversary $\mathcal{A}$ who manages with a non-negligible probability to win the **Forge-Game** and therefore manages to output a signature τ on a message m with respect to a pseudonym P_{ep} and a signature τ_{ep} such that $\text{DSAVerify}(P_{\text{ep}}, m, \tau) =$ **true**, $\text{DSAVerify}(P_{\text{AA}}, P_{\text{ep}}, \tau_{\text{ep}}) =$ **true** and that m does not correspond to any query made by $\mathcal{A}$ to the challenger $\mathcal{C}$.

We construct an efficient adversary $\mathcal{B}$ which uses adversary $\mathcal{A}$ to either break the unforgeability of the ideal DAA functionality $\mathcal{F}_{\text{pdaa+}}$ or to win the EUF-CMA experiment.

$\mathcal{B}$ will execute $\mathcal{A}$ and simulate the challenger $\mathcal{C}$. Initially, $\mathcal{B}$ will randomly select a target vehicle $V^\star$ and epoch $\text{ep}^\star$. $\mathcal{B}$ will simulate the Setup algorithm and will provide $\mathcal{A}$ with the resulting DAA public key ipk, the parameters par and the ECDSA public key P_{AA}. $\mathcal{B}$ also simulates N_V vehicles with identities $\{V_1, \ldots, V_{N_V}\}$. For each vehicle $\mathcal{V}_i$, $\mathcal{B}$ selects the secret key vsk_i, the Sybil secret s_i and simulates the Join protocol and the EA so that each V_i has a DAA credential cre_i on vsk_i and s_i. Finally, $\mathcal{B}$ provides $\mathcal{A}$ with a reference to each vehicle.

The adversary $\mathcal{A}$ makes a polynomial number of signature requests to the adversary $\mathcal{B}$. Each request will specify a vehicle identity $V_i \in \{V_1, \ldots, V_{N_V}\}$, a message m and an epoch ep. If $V_i = V^\star$, $\text{ep} = \text{ep}^\star$ and V_i does not have the signing key x_{ep} then $\mathcal{B}$ will simulate the Issue protocol with the AA using the ideal-DAA-functionality $\mathcal{F}_{\text{pdaa+}}$ verify interface and the signature oracle $\mathcal{O}_S$ from the EUF-CMA experiment. Once V_i has the pseudonym signing key x_{ep}, then $\mathcal{B}$ will also use the signature oracle $\mathcal{O}_S$ from the EUF-CMA experiment to sign m.

For all other vehicles $V_i \in \{V_1, \ldots, V_{N_V}\} \setminus V^\star$, $\mathcal{B}$ will simulate V_i, the Issue protocol with the AA to generate x_{ep} if necessary and the DSASign algorithm to compute the vehicle signature on m with respect to P_{ep}. In all cases, adversary $\mathcal{B}$ will provide $\mathcal{A}$ with the resulting ECDSA signature τ on m, the pseudonym public key P_{ep} and the authorising AA signature τ_{ep}.

At some point $\mathcal{A}$ will terminate. By hypothesis and with a non-negligible probability $\mathcal{A}$ must output a signature τ on a message m, a pseudonym public key P_{ep} and a signature τ_{ep} such that:

- The tuple $(m^\star, \tau^\star)$ is a valid message-signature pair with respect to the pseudonym public key $P_{\text{ep}}{}^\star$. In other words $\text{DSAVerify}(P_{\text{ep}}, m, \tau) =$ **true**.
- The pseudonym public key and AA signature $(P_{\text{ep}}{}^\star, \tau^\star{}_{\text{ep}})$ is a valid message-signature pair with respect to the AA public key P_{AA}.
  i.e. $\text{DSAVerify}(P_{\text{AA}}, P_{\text{ep}}{}^\star, \tau^\star{}_{\text{ep}}) =$ **true**.
- The message m does not correspond to any query made by the adversary $\mathcal{A}$ to adversary $\mathcal{B}$.

If $\mathcal{V}_i = \mathcal{V}^\star$ and $\text{ep} = \text{ep}^\star$ then $\mathcal{A}$ will send the signature τ on the message m, the pseudonym public key P_{ep} and the signature τ_{ep} to $\mathcal{B}$, otherwise it will not. This means that adversary $\mathcal{A}$ has either broken the unforgeability of the ideal DAA functionality $\mathcal{F}_{\text{pdaa+}}$ or has broken the existential unforgeability of the ECDSA signature scheme.

The advantage of the adversary $\mathcal{A}$ winning the unforgeability game is therefore the probability that $\mathcal{A}$ attacks the target vehicle $V^\star$ during the epoch ep multiplied by the advantage of adversary $\mathcal{B}$ against the DAA and the ECDSA signature schemes. Since adversary $\mathcal{A}$ may attack either the signature τ on m or the signature τ_{ep} on P_{ep}, the advantage is further divided by two. Where N_{ep} is the number of different epochs that $\mathcal{A}$ requested signatures for and N_V is the number of vehicles simulated by $\mathcal{B}$, the advantage of $\mathcal{A}$ winning the unforgeability game is:

$$\mathbf{Adv}[\mathcal{A}^{\textbf{Forge-Game}}_{\text{VDAA}}] = \frac{\mathbf{max}\{\mathbf{Adv}[\mathcal{B}^{\textbf{EUF-CMA}}_{\text{ECDSA}}], \mathbf{Adv}[\mathcal{B}^{\mathcal{F}_{\text{pdaa+}}}_{\text{DAA}}]\}}{2 * N_V \cdot N_{\text{ep}}}$$

8.2. Unlinkability

We show that if the underlying DAA scheme provides unlinkability then our VDAA scheme satisfies unlinkability with respect to Definition 2. Informally, an adversary cannot distinguish between messages sent by a single vehicle during two different epochs and messages sent by two different vehicles during the same two epochs because the underlying DAA scheme has the property of strong privacy. Strong privacy guarantees, provided the vehicle host is honest, when given two DAA signatures σ_1 and σ_2 with respect to two different basenames $\text{bsn}_1 \neq \text{bsn}_2$, no adversary can distinguish whether both signatures were created by one vehicle or two. Strong privacy holds even when the TPM is malicious and the EA is corrupt. Because our VDAA scheme uses a DAA scheme with strong privacy, the ECDSA pseudonyms that are requested by honest vehicle hosts are as unlinkable as the DAA credentials which are used to request them.

Theorem 2. Let DAA be a secure DAA scheme with respect to the ideal functionality $\mathcal{F}_{\text{pdaa+}}$ defined by Camenisch et al. [21], then the VDAA scheme we present in Section 6 is secure with respect to unlinkability as defined in Section 7.2.

Proof. Assume for contradiction that our VDAA scheme is not unlinkable. This means that there is an adversary $\mathcal{A}$ who manages to win the **Priv-Game** and therefore manages to output $b' = b$ with a non-negligible advantage.

We construct an efficient adversary $\mathcal{B}$ that uses $\mathcal{A}$ to distinguish between interactions with the ideal functionality $\mathcal{F}_{\text{pdaa+}}$ and the underlying DAA scheme DAA. Specifically, every time a vehicle wants to sign a message m with respect to a unique basename bsn, $\mathcal{F}_{\text{pdaa+}}$ generates a fresh group secret key isk′ and then signs m using isk′. Using a fresh isk for every signature guarantees that signatures are anonymous. We use adversary $\mathcal{A}$ to distinguish from the ideal functionality $\mathcal{F}_{\text{pdaa+}}$ by breaking the anonymity of the DAA signatures used to request vehicle pseudonyms.

Initially, the adversary $\mathcal{B}$ will simulate the Setup algorithm and provides adversary $\mathcal{A}$ with the resulting

DAA group public key pair $\mathsf{ik} = (\mathsf{ipk}, \mathsf{isk})$, the parameters par and the AA ECDSA public key pair $\mathsf{ak} = (P_{\mathrm{AA}}, x_{\mathrm{AA}})$. $\mathcal{B}$ will also simulate two vehicles V_0 and V_1 and will simulate the Join protocol and provide $\mathcal{A}$ with the vehicle secret keys $\mathsf{vsk}_0, \mathsf{vsk}_1$ and the Sybil secrets s_0, s_1, respectively. The adversary $\mathcal{B}$ will select one target vehicle $V^\star \in \{V_0, V_1\}$.

The adversary $\mathcal{A}$ selects two distinct non-overlapping epochs $\mathsf{ep0}$ and $\mathsf{ep1}$ and submits them to adversary $\mathcal{B}$. $\mathcal{B}$ selects a bit $b \leftarrow \{0,1\}$ and then for $V_i \in \{V_b, V_{b-1}\}$ will select the epoch keys $x_{\mathsf{ep0},i}, x_{\mathsf{ep1},i}$ and computes the public keys $P_{\mathsf{ep0},i}, P_{\mathsf{ep1},i}$. For the target vehicle $V^\star \in \{V_b, V_{b-1}\}$, $\mathcal{B}$ will interact with the ideal functionality $\mathcal{F}_{\mathsf{pdaa+}}$ sign interface to compute the DAA signatures σ_0, σ_1 on $P_{\mathsf{ep0},\star}$ and $P_{\mathsf{ep1},\star}$. For the non-target vehicle $\bar{V}^\star$ the adversary $\mathcal{B}$ will compute the DAA signatures σ_0, σ_1 by simulating the first part of the standard Issue protocol. For $V_i \in \{V_b, V_{b-1}\}$, $\mathcal{B}$ will simulate the remainder of the Issue protocol and will provide the adversary $\mathcal{A}$ with the pseudonym signatures $\tau_{\mathsf{ep0},i}, \tau_{\mathsf{ep1},i}$ on the public keys $P_{\mathsf{ep0},i}, P_{\mathsf{ep1},i}$.

The adversary $\mathcal{A}$ will make a polynomial number of signature requests to the adversary $\mathcal{B}$. Each request will comprise a message m and an epoch ep. If $\mathsf{ep} \notin \{\mathsf{ep0}, \mathsf{ep1}\}$ then the challenger outputs $\perp$ and then, as per the unlinkability game, $\mathcal{B}$ will act according to the bit b

- If $b = 0$, then $\mathcal{B}$ simulates vehicle V_0, simulates the DASign algorithm and outputs the signed message (τ, m) with respect to pseudonym $P_{\mathsf{ep0},0}$.
- If $b = 1$ and $\mathsf{ep} = \mathsf{ep0}$ then $\mathcal{B}$ simulates vehicle V_0. If $\mathsf{ep} = \mathsf{ep1}$ then $\mathcal{B}$ simulates vehicle V_1. $\mathcal{B}$ outputs the signed message (τ, m) with respect to pseudonym $P_{\mathsf{ep0},0}$ or $P_{\mathsf{ep1},1}$, respectively.

At some point adversary $\mathcal{A}$ will terminate and by hypothesis will output $b' = b$ with a non-negligible advantage. Since the pseudonym keys are random bitstrings generated by the trusted host, $\mathcal{A}$ must have attacked the ideal DAA sign functionality $\mathcal{F}_{\mathsf{pdaa+}}$ used to request the pseudonym signatures. The advantage of $\mathcal{A}$ in the unlinkability game is therefore the product of the probability that the target vehicle $V^\star$ is exposed by the bit value b, the probability that $\mathcal{A}$ attacks the target vehicle $V^\star$ and the advantage of adversary $\mathcal{B}$ against the $\mathcal{F}_{\mathsf{pdaa+}}$ sign interface.

$$\mathbf{Adv}[\mathcal{A}^{\textbf{Priv-Game}}_{\mathsf{VDAA}}] = \frac{1}{4} \cdot \mathbf{Adv}[\mathcal{B}^{\mathcal{F}_{\mathsf{pdaa+}}}_{\mathsf{DAA}}]$$

9. Evaluation

This section argues that the VDAA scheme we presented in Section 6 meets the standard security and privacy requirements for V2X from Section 2.

Authentication The primary security requirement for V2X is that there is a mechanism for determining the authenticity and integrity of broadcast messages. We show that our VDAA scheme is secure with respect to unforgeability in Section 8.1. The unforgeability of our scheme means that when the EA and all vehicle TPMs are honest, no adversary can forge a request for a pseudonym certificate from the AA. Correspondingly, if all TPMs are uncorrupted then no adversary can create a valid signature on any broadcast message, all messages then originate from a particular honest vehicle and the authentication and integrity of received messages is assured.

Unlinkability The main privacy mechanism in V2X is the use of multiple pseudonymous identities such that an adversary is unable to distinguish whether two uncorrelated identities originate from a single source or not. We show that our VDAA scheme is secure with respect to unlinkability in Section 8.2. The unlinkability of our VDAA scheme means that even if the AA is subverted, the signatures on broadcast messages sent by any particular vehicle are indistinguishable from those created by any other road user.

Corrupt CA Resistance Vehicles should be protected from dishonest or collaborating certificate authorities. In contrast to the leading V2X standards [12], [40] our scheme retains vehicle unlinkability despite dishonest certificate authorities.

Revocation It is critically important that vehicles which send false information can be prevented from continued participation. Our VDAA scheme allows both vehicle, private-key and signature based revocation which we describe in Section 6. Unlike other solutions that also provide enhanced vehicle privacy [25], [27], [28], we uniquely retain centralised control over revocation and are therefore able remove vehicles despite vehicle hosts that may refuse to forward messages to the TPM.

Sybil Resistance We optimally limit Sybil attacks by restricting each vehicle to a single pseudonym request per epoch. Requests for multiple pseudonyms in the same epoch are denied, forfeit vehicle unlinkability and are detected by the AA. At most, a vehicle can use just two pseudonyms concurrently and only during the small certificate overlap period that is necessary for harmonising vehicles without a synchronised clock source.

Performance Analysis. The most performance critical operation in V2X is broadcast-message signature verification. Correspondingly and based on early field studies [32], the two major V2X standards both use the ECDSA signature scheme. Since our VDAA scheme also uses regular ECDSA signatures on broadcast messages we occur no additional overhead with regards to either signing or verification of CAM. In line with the standards we use either NIST curve P-256 [41] or BrainpoolP256r1 [42] which result in a signature size of 64 bytes.

Where our scheme introduces an overhead compared to the standards is when vehicles are enrolled for the first time and, more importantly, each time they request a pseudonym certificate. The DAA credential and signature sizes used in our Join and Issue protocols depend on the underlying DAA scheme. For the q-SDH-based instantiation of Camenisch et al. [43] the DAA credential size is 96 bytes, composed of 2 elements in $\mathbb{Z}_p$ and one in $\mathbb{G}_1$. The corresponding signature size is 356 bytes, composed from 6 elements in $\mathbb{Z}_p$, 4 elements in $\mathbb{G}_1$ and one 32 byte hash digest. The total bandwidth requirements of our Issue protocol, run each time a vehicle requests

a new pseudonym, is one ECDSA pseudonym public key P_{ep} and signature τ_{ep}, one 356 byte DAA signature σ and a 4 byte epoch identifier ep. In other words, compared to the ETSI standard our scheme requires an additional 360 bytes of bandwidth per pseudonym that is requested by each vehicle.

Centralised revocation depends on the attestation list Auth-L, maintained by the AA, which retains all of the randomised DAA signature and ECDSA pseudonym tuples received during all runs of the Join protocol. Each tuple in Auth-L, comprising one 356 byte DAA signature and one 64 byte ECDSA pseudonym public key, requires 420 bytes of storage. Taking a 5 minute pseudonym validity period, the upper bound on the AA storage required is 118.125 KB per vehicle, per day. This number scales linearly in the proportion of time that a vehicle is driven for, for example reducing to less than 5 KB for vehicles used for one hour per day.

Computationally, each pseudonym request requires one DAA sign operation which takes approximately 20 ms [43] for q-SDH DAA. The DAA verification algorithm run by the AA is also efficient and takes around 60 ms.

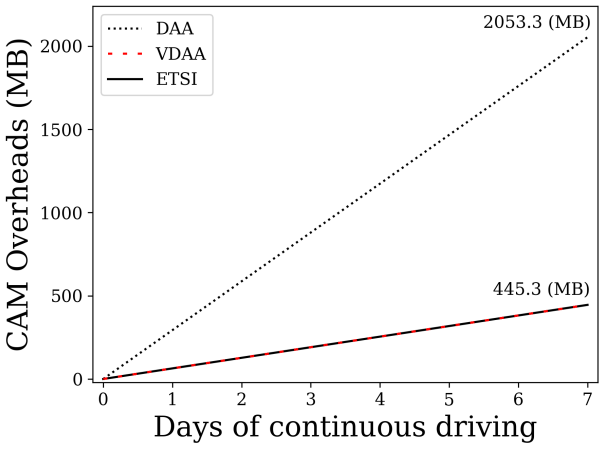

Figure 4: With an epoch duration of 5 minutes, a comparison of the CAM signature and certificate bandwidth overheads between the ETSI standard, our VDAA scheme and the direct application of DAA.

Since our VDAA scheme authenticates each broadcast message with a standard ECDSA signature, our scheme has the same signature and certificate bandwidth overheads as the ETSI approach. Assuming the standard epoch duration of 5 minutes, and that authorising certificates are included in one out of every 10 messages sent by a vehicle, Figure 4 shows the bandwidth required by both our solution and ETSI's in contrast to the direct application of DAA [**?**]. Including the certificate in one out of every 10 CAM is a pessimistic estimate when considering that this captures the scenario in which all interactions lasting more than one second result in sending the necessary certificate.

Whilst VDAA requires no additional CAM bandwidth over the ETSI standard, a small certificate issuance overhead is required for the DAA signatures used to request each certificate. In particular, VDAA requires the additional transmission of one 356 byte DAA signature and one 4 byte epoch identifier per epoch. The 5 minute value

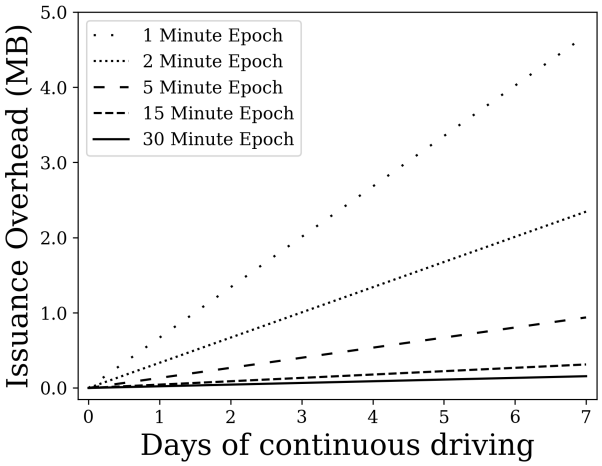

Figure 5: A comparison of the additional certificate issuance bandwidth required by VDAA, in contrast to the ETSI standard, at epoch durations ranging from 1 to 30 minutes.

used in our analysis of is chosen for conformity with the value used by ETSI when evaluating different pseudonym change strategies [14] and is also the value recommended by SAE [45]. Figure 5 shows how the VDAA certificate issuance overhead scales to epoch durations ranging from 1 to 30 minutes. We note that even with an epoch duration of only 1 minute, VDAA requires less than 4.8 MB of vehicle-to-AA certificate issuance bandwidth per week of continuous driving. Using the recommended 5 minute epoch period, no more than 701 KB is needed per week. In practice, as most vehicles are only operated for a small proportion of each day, the overheads will be much less than the upper bounds shown here. In addition this bandwidth is only required periodically and can be scheduled according to the connectivity available to each vehicle. When considering all overheads including CAM authentication and certificate issuance in this model, VDAA requires less than 22% of the total bandwidth needed for the direct application of DAA (e.g. Chen at al. [30]).

Revocation in DAA is an inefficient process which is linear in the size of the revocation list [22]. Specifically, the signature-based revocation that allows for the credentials of misbehaving vehicles to be revoked introduces a significant computational overhead. For example, using the ECC-DAA [20] scheme on a 192 bit curve and a blacklist with 200 revoked signatures, attestation takes 24.4 seconds on an ARM11 host [46]. The corresponding proof verification takes 1.4 seconds to verify on a standard desktop PC. Xi et al. [46] observe that the proof burden can be reduced to signatures revoked since the last proof was created and a corresponding approach is to compute the attestation proofs periodically, such as when charging the vehicle overnight. Another way of minimising the performance impact of revocation is to implement DAA groups associated with short periods of time. For example, the EA could create a DAA group for each week; Every week, each vehicle would prove that it is a non-revoked member of the current group and would be issued a new credential. The AA may even forego revocation altogether

and simply wait for revoked vehicles to be removed during weekly re-keying.

10. Conclusion

We have presented a novel V2X scheme based on DAA and a unique serial number construction that prevents the abuse of anonymous credentials. Our VDAA scheme, which we have proven secure under standard assumptions for DAA, is compatible with the PKI architectures of the proposed ITS standards [12], [40] and addresses the currently unmet need [10], [47] for measures which limit long-term vehicle tracking and that minimise the impact of certificate authority collusion. Critically, and relative to both the ITS standards and many of the proposals in the literature [12], [28], [30], [48], our scheme provides a stronger adversary model and a higher degree of privacy. Rather than forfeiting their canonical identity, vehicles that send malicious messages or which request multiple pseudonyms for the same epoch only forfeit their unlinkability and ongoing participation in the scheme. Vehicles are safe from certificate authorities that conspire to revoke their privacy. In contrast to other V2X proposals [25], [28] that utilise anonymous attestation, we do not require cooperation from the vehicle host or TPM to enforce revocation.

References

[1] J. Harding, G. Powell, R. R., Yoon, J. Fikentscher, C. Doyle, D. Sade, M. Lukuc, J. Simons, and J. Wang, "Vehicle-to-Vehicle Communications: Readiness of V2V Technology for Application (Report No. DOT HS 812 014)," 2014.

[2] B. Wiedersheim, Z. Ma, F. Kargl, and P. Papadimitratos, "Privacy in inter-vehicular networks: Why simple pseudonym change is not enough," in *Seventh International Conference on Wireless On-demand Network Systems and Services*, 2010.

[3] I. Symeonidis, A. Aly, M. A. Mustafa, B. Mennink, S. Dhooghe, and B. Preneel, "SePCAR: A Secure and Privacy-Enhancing Protocol for Car Access Provision," in *Computer Security – ESORICS 2017*, 2017.

[4] A. Pham, I. Dacosta, G. Endignoux, J. R. T. Pastoriza, K. Huguenin, and J.-P. Hubaux, "Oride: A privacy-preserving yet accountable ride-hailing service," in *26th USENIX Security Symposium (USENIX Security 17)*, 2017.

[5] "Intelligent Transport Systems in action," Directorate-General for Mobility and Transport, European Commission, Tech. Rep., August 2010.

[6] T. Pornin, "Deterministic Usage of the Digital Signature Algorithm (DSA) and Elliptic Curve Digital Signature Algorithm (ECDSA)," Tech. Rep., 2013.

[7] Y. Lindell, "Fast Secure Two-Party ECDSA Signing," in *Advances in Cryptology – CRYPTO 2017*, 2017.

[8] C. P. Schnorr, "Efficient identification and signatures for smart cards," in *Advances in Cryptology — CRYPTO' 89 Proceedings*, 1990.

[9] O. Goldreich, *Foundations of Cryptography*. Cambridge University Press, 2004, vol. 2.

[10] A. . W. Party, "Opinion 03/2017 on Processing personal data in the context of Cooperative Intelligent Transport Systems (C-ITS)," Tech. Rep., 2017.

[11] "ETSI TS 102 731. Intelligent Transport Systems (ITS); Security; Security Services and Architecture," V1.1.1, European Telecommunications Standards Institute, Tech. Rep., September 2010.

[12] B. Brecht, D. Therriault, A. Weimerskirch, W. Whyte, V. Kumar, T. Hehn, and R. Goudy, "A Security Credential Management System for V2X Communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, 2018.

[13] J. Petit, F. Schaub, M. Feiri, and F. Kargl, "Pseudonym Schemes in Vehicular Networks: A Survey," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, 2015.

[14] "ETSI TR 103 415. Intelligent Transport Systems (ITS); Security; Pre-standardization study on pseudonym change management," European Telecommunications Standards Institute, Tech. Rep., 2018.

[15] "IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture," Tech. Rep., 2014.

[16] N. Bißmeyer, H. Stübing, E. Schoch, S. Götz, J. P. Stotz, and B. Lonc, "A generic public key infrastructure for securing Car-to-X communication," in *18th World Congress on Intelligent Transport Systems*, 2011.

[17] W. Whyte, A. Weimerskirch, V. Kumar, and T. Hehn, "A security credential management system for V2V communications," in *IEEE Vehicular Networking Conference*, 2013.

[18] E. Brickell, J. Camenisch, and L. Chen, "Direct Anonymous Attestation," in *Proceedings of the 11th ACM Conference on Computer and Communications Security*, ser. CCS '04, 2004.

[19] ISO, *ISO/IEC standard 11889-1:2015: Trusted platform module library – Part 1: Architecture*. International Organization for Standardization, 2015.

[20] L. Chen, D. Page, and N. P. Smart, "On the Design and Implementation of an Efficient DAA Scheme," in *Smart Card Research and Advanced Application*, 2010.

[21] J. Camenisch, L. Chen, M. Drijvers, A. Lehmann, D. Novick, and R. Urian, "One TPM to Bind Them All: Fixing TPM 2.0 for Provably Secure Anonymous Attestation," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017.

[22] V. Kumar, H. Li, N. Luther, P. Asokan, J.-M. J. Park, K. Bian, M. B. H. Weiss, and T. Znati, "Direct Anonymous Attestation with Efficient Verifier-Local Revocation for Subscription System," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, ser. ASIACCS '18, 2018.

[23] E. Brickell and J. Li, "Enhanced Privacy ID: A Direct Anonymous Attestation Scheme with Enhanced Revocation Capabilities," in *Proceedings of the 2007 ACM Workshop on Privacy in Electronic Society*, ser. WPES '07, 2007.

[24] ——, "Enhanced Privacy ID from Bilinear Pairing for Hardware Authentication and Attestation," in *2010 IEEE Second International Conference on Social Computing*, 2010.

[25] J. Whitefield, L. Chen, T. Giannetsos, S. Schneider, and H. Treharne, "Privacy-enhanced capabilities for VANETs using direct anonymous attestation," in *2017 IEEE Vehicular Networking Conference (VNC)*, 2017.

[26] D. Förster, H. Löhr, J. Zibuschka, and F. Kargl, "REWIRE – Revocation Without Resolution: A Privacy-Friendly Revocation Mechanism for Vehicular Ad-Hoc Networks," in *Trust and Trustworthy Computing*, 2015.

[27] J. Whitefield, L. Chen, F. Kargl, A. Paverd, S. Schneider, H. Treharne, and S. Wesemeyer, "Formal Analysis of V2X Revocation Protocols," 2017.

[28] D. Förster, F. Kargl, and H. Löhr, "PUCA: A pseudonym scheme with strong privacy guarantees for vehicular ad-hoc networks," *Ad Hoc Networks*, vol. 37, 2016.

[29] J. Camenisch, S. Hohenberger, M. Kohlweiss, A. Lysyanskaya, and M. Meyerovich, "How to Win the Clone Wars: Efficient Periodic N-times Anonymous Authentication," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, ser. CCS '06, 2006, pp. 201–210.

[30] L. Chen, S. Ng, and G. Wang, "Threshold Anonymous Announcement in VANETs," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, 2011.

[31] "ETSI TS 102 941. Intelligent Transport Systems (ITS); Security; Trust and Privacy Management," European Telecommunications Standards Institute, Tech. Rep., 2018.

[32] "Security Requirements of Vehicle Security Architecture," PREparing SEcuRe VEhicle-to-X Communication Systems (PRESERVE), Tech. Rep., 2011.

[33] J. Camenisch and M. Stadler, "Efficient Group Signature Schemes for Large Groups (Extended Abstract)," in *Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '97, 1997.

[34] S. Vaudenay, "The Security of DSA and ECDSA," in *Public Key Cryptography — PKC 2003*, 2002.

[35] A. Lysyanskaya, R. L. Rivest, A. Sahai, and S. Wolf, "Pseudonym Systems," in *Selected Areas in Cryptography*, H. Heys and C. Adams, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 184–199.

[36] D. Boneh and X. Boyen, "Short Signatures Without Random Oracles," in *Advances in Cryptology - EUROCRYPT 2004*, C. Cachin and J. L. Camenisch, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 56–73.

[37] M. H. Au, W. Susilo, and Y. Mu, "Constant-size dynamic k-taa," in *Security and Cryptography for Networks*, R. De Prisco and M. Yung, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 111–125.

[38] E. Verheul, C. Hicks, and F. D. Garcia, "IFAL: Issue First Activate Later Certificates for V2X," in *IEEE European Symposium on Security and Privacy, EuroS&P*, 2019.

[39] S. Goldwasser, S. Micali, and R. L. Rivest, "A Digital Signature Scheme Secure Against Adaptive Chosen-message Attacks," *SIAM J. Comput.*, vol. 17, no. 2, 1988.

[40] "ETSI TS 103 097. Intelligent Transport Systems (ITS); Security; Security header and certificate formats," European Telecommunications Standards Institute, Tech. Rep., 2017.

[41] National Institute of Standards and Technology, "Digital Signature Standard (DSS) (FIPS 186-4)." Tech. Rep., 2013.

[42] M. Lochter and J. Merkle, "Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation," Tech. Rep., March 2010.

[43] J. Camenisch, M. Drijvers, and A. Lehmann, "Anonymous Attestation Using the Strong Diffie Hellman Assumption Revisited," in *Trust and Trustworthy Computing*, 2016.

[44] L. Chen and J. Li, "Flexible and Scalable Digital Signatures in TPM 2.0," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, ser. CCS '13, 2013.

[45] "On-Board System Requirements for V2V Safety Communications," *SAE J2945/1*, 2016.

[46] L. Xi, D. Feng, Y. Qin, F. Wei, J. Shao, and B. Yang, "Direct Anonymous Attestation in practice: Implementation and efficient revocation," in *2014 Twelfth Annual International Conference on Privacy, Security and Trust*, 2014.

[47] *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 (General Data Protection Regulation)*. Official Journal of the European Union, 2016.

[48] F. Kargl, P. Papadimitratos, L. Buttyan, M. Müter, E. Schoch, B. Wiedersheim, T. V. Thong, G. Calandriello, A. Held, A. Kung, and J. P. Hubaux, "Secure vehicular communication systems: implementation, performance, and research challenges," *IEEE Communications Magazine*, vol. 46, no. 11, 2008.

Appendix

A signature scheme is a triple (G, S, V) of efficient algorithms which satisfy the following two conditions

- On input the security parameter 1^η the key-generation algorithm G outputs a pair of bit strings (s, v).
- For every pair (s, v) in the range of $G(1^\eta)$ and $\forall m \in \{0,1\}^*$, the signing algorithm S and the verification algorithm V satisfy the following consistency

$$\Pr[V(v, m, S(s, m)) = 1] = 1$$

The standard security definition for public key signature schemes is the notion of *existential forgery on adaptively chosen message attacks* (EUF-CMA) [39]. The EUF-CMA definition involves a game which takes as input the security parameter 1^η and an adversary $\mathcal{A}$ who interacts with a challenger $\mathcal{C}$.

EUF-CMA$_{\mathcal{C}}(1^\eta, \mathcal{A})$:

1) The challenger $\mathcal{C}$ simulates the key-generation algorithm G and provides the adversary $\mathcal{A}$ with the target verification key $v^\star$.
2) Challenge: Polynomially many times, the adversary $\mathcal{A}$ submits a message m to the challenger to simulates the signing algorithm S and provides $\mathcal{A}$ with the signature σ on the message m.
3) Output: The adversary $\mathcal{A}$ outputs a signature $\sigma^\star$ on the message $m^\star$.

An adversary $\mathcal{A}$ wins the EUF-CMA game if

1) $V(v^\star, m^\star, \sigma^\star) = 1$
2) The message $m^\star$ does not correspond to any query made by the adversary $\mathcal{A}$ to the challenger $\mathcal{C}$.

Definition 3 (EUF-CMA). A digital signature scheme $\sum = (G, S, V)$ is said to be secure against EUF-CMA if for all efficient adversaries $\mathcal{A}$, the probability of the experiment **EUF-CMA**$_{\sum}(\mathcal{A}) =$ true is a negligible function of η.