# Evaluating On-demand Pseudonym Acquisition Policies in Vehicular Communication Systems

Mohammad Khodaei
Networked Systems Security Group
KTH Royal Institute of Technology
Stockholm, Sweden
khodaei@kth.se

Panos Papadimitratos
Networked Systems Security Group
KTH Royal Institute of Technology
Stockholm, Sweden
papadim@kth.se

## ABSTRACT

Standardization and harmonization efforts have reached a consensus towards using a special-purpose Vehicular Public-Key Infrastructure (VPKI) in upcoming Vehicular Communication (VC) systems. However, there are still several technical challenges with no conclusive answers; one such an important yet open challenge is the acquisition of short-term credentials, *pseudonym*: how should each vehicle interact with the VPKI, e.g., how frequently and for how long? Should each vehicle itself determine the pseudonym lifetime? Answering these questions is far from trivial. Each choice can affect both the user privacy and the system performance and possibly, as a result, its security. In this paper, we make a novel systematic effort to address this multifaceted question. We craft three generally applicable policies and experimentally evaluate the VPKI system performance, leveraging two large-scale mobility datasets. We consider the most promising, in terms of efficiency, pseudonym acquisition policies; we find that within this class of policies, the most promising policy in terms of privacy protection can be supported with moderate overhead. Moreover, in all cases, this work is the first to provide tangible evidence that the state-of-the-art VPKI can serve sizable areas or domain with modest computing resources.

## Keywords

Vehicular Communications, Security, Privacy, Access Control, Identity and Credential Management, Vehicular PKI

## 1. INTRODUCTION

Vehicular Communication (VC) systems aim at enhancing transportation safety and efficiency with a gamut of applications, ranging from collision avoidance to traffic condition updates and Location Based Services [9, 18]. By the same token, the need to enhance security and protect user privacy is well understood [19]. Standardization bodies (IEEE 1609.2 WG [13] and ETSI [9]) and harmonization efforts (C2C-CC [6]) have reached a consensus to use public key cryptography: a set of Certification Authorities (CAs) constitute the Vehicular Public-Key Infrastructure (VPKI), providing credentials to legitimate vehicles. Each vehicle, equipped with an On-Board Unit (OBU), is provided with a Long Term Certificate (LTC) (and has the corresponding private key) to ensure accountable identification of the vehicle. To achieve unlinkability of messages originating a vehicle, a set of short-term certificates, termed *pseudonyms*, are used with the corresponding short-term private keys. A vehicle digitally signs an outgoing message, e.g., a Cooperative Awareness Message (CAM) or a Decentralized Environmental Notification Message (DENM), time- and geo-stamped, using the private key that corresponds to its currently valid pseudonym. It then attaches the pseudonym to the signed messages to facilitate validation by any recipient. Upon reception, the pseudonym is verified (assuming a trust relationship with its issuer) before the message itself (signature validation). This process ensures communication authenticity, integrity and non-repudiation. By frequently changing the pseudonym (and the corresponding private key), the sender privacy is also protected as the pseudonyms per se are inherently unlinkable (if they are issued appropriately, as it will become clear later).

Table 1 classifies different approaches for issuing pseudonyms. A group of proposals suggests preloading the vehicles with the required pseudonyms for a long period, which we term as *preloading* schemes. Accordingly, systems relying on preloading (C2C-CC [6], CAMP VSC3 [23], PRESERVE [2]) issue pseudonyms with *overlapping* lifetimes (validity intervals) to facilitate the operation of the vehicles in safety critical situations. Recall that safety applications necessitate partial linkability to operate because inferring a collision hazard based on unlinkable CAMs can be hard and error prone. However, having multiple simultaneously valid pseudonyms sets the ground for Sybil-based [8] misbehavior, i.e., allowing internal adversaries to inject multiple bogus messages, thus controlling the outcome of specific protocols, e.g., those based on voting [20]. To thwart this, SeVeCom [17] and Safety Pilot [3] suggested preloading vehicles with pseudonyms that have *non-overlapping* lifetimes (no vehicle has more than one valid pseudonym at any given time). Another group of proposals suggests more frequent vehicles interactions with the Roadside Units (RSUs), i.e., with the VPKI servers, e.g., once or multiple times per day, which we term *on-demand* schemes. Among those, SRAAC [10], V-tokens [21], and CoPRA [5] propose issuing pseudonyms with overlapping lifetimes; while VeSPA [4], SEROSA [12], SR-VPKI [14], and PUCA [11] propose issuing pseudonyms with non-overlapping lifetimes.

Having different proposals with diverse views on pseudonym acquisition process emphasizes the need to standardize this process with clear objectives. Clearly, there are trade-offs: the longer the interval to obtain pseudonyms, the less frequent the vehicle-VPKI communications, but the higher the probability (and the longer the duration) the pseudonym provider can trivially link sets of pseudonyms issued for the same vehicle and thus all communication signed by

Table 1: Comparing Pseudonym Refilling Strategies

| Metrics \ Strategies | Preloading & Overlapping | Preloading & Nonoverlapping | On-demand & Overlapping | On-demand & Nonoverlapping |
|---|---|---|---|---|
| Storage size | large | large | small | small |
| Pseudonym quantity | fixed & low volume | fixed & high volume | varying | varying |
| Pseudonym lifetime | long | short | varying | varying |
| V-VPKI communication frequency | low | low | high | high |
| Communication overhead | low | low | high | high |
| Efficient pseudonym utilization | very low | very low | high | high |
| Pseudonym revocation | difficult & challenging | difficult & challenging | no need (lower risk) | no need (lower risk) |
| Pseudonym vulnerability window | wide | wide | narrow | narrow |
| Resilience to Sybil-based misbehavior | × | ✓ | × | ✓ |
| User privacy protection (probability of linking sets of pseudonyms based on timing information) | privacy protection: high (probability of linking: low) | privacy protection: low (probability of linking: high) | privacy protection: high (probability of linking: low) | privacy protection: low (probability of linking: high) |
| User privacy protection (duration for which a pseudonym provider can trivially link sets of pseudonyms for the same vehicle; the longer the duration, the higher the chance to link sets of pseudonyms) | privacy protection: low (long duration) | privacy protection: low (long duration) | privacy protection: high (short duration) | privacy protection: high (short duration) |
| Effect on safety application operations | low | low | high | high |
| Deployment cost (e.g. RSU) | low | low | high | high |
| Proposals & schemes | C2C-CC [6], PRESERVE [2], CAMP VSC3 [23] | SeVeCom [17], Safety Pilot [3] | SRAAC [10], V-tokens [21], CoPRA [5] | VeSPA [4], SEROSA [12], SR-VPKI [14], PUCA [11] |

that vehicle throughout that period. Furthermore, a privacy concern arises for any strategy that issues pseudonyms with non-overlapping lifetimes [14]: the use of timing information can enable an eavesdropper to link pseudonyms (the transcript of pseudonymously authenticated messages) by inspecting their successive pseudonym lifetimes. Additionally, efficient pseudonym utilization is a challenging issue: the average trip duration, according to available real mobility traces [7, 22], is around 10 minutes during week days. According to the US Census Bureau annual American Community Survey [1, 23], the average daily commute time is less than an hour. Thus, over-provisioning the vehicles with a large number of unused pseudonyms would be a waste of computation and resources.

Due to the (i) improved security, i.e., resilience to Sybil-based misbehavior, (ii) user privacy protection, i.e., unlinkable pseudonyms, and (iii) efficiency, i.e., no over-provisioning, on-demand pseudonym acquisition with non-overlapping pseudonym lifetimes is preferable. Within this class of proposals, we seek to address the fundamental questions: *how frequently, and for which period, each vehicle should interact with the VPKI to obtain pseudonyms?* Moreover, *should each vehicle have the freedom to determine the pseudonym acquisition periods and the lifetime of the pseudonyms?* We further need *to evaluate the effects of any approach on the overall VPKI performance.* The performance of the VPKI relates to security and safety: if a pseudonym acquisition approach and specific conditions result in excessive delays to provision a vehicle, then either the vehicle would have to sacrifice its privacy by using its LTC, or it would be excluded from the system, exactly reducing safety and transportation efficiency. This is why we need to investigate the overall effect on the VPKI system.

**Contributions:** We make a novel systematic effort to understand the pseudonym acquisition process. We propose three generally applicable policies for pseudonym provisioning that capture alternative approaches in the literature. To evaluate the overall VPKI performance, the fundamental metric is the *end-to-end latency* for the vehicle to obtain pseudonyms. This is essentially the only bottleneck any vehicle would encounter in an on-demand approach. We assess the effect of the three policies on the performance of an actual implementation of the most promising, in terms of efficiency, VPKI; we leverage two large-scale mobility traces (assuming all vehicles are equipped with VC enabling equipments) to emulate realistic conditions.

In the rest of the paper, we give an overview of the system and pseudonym acquisition policies (Sec. 2), followed by the description of the protocols (Sec. 3). We evaluate the effects of each policy on the overall VPKI performance (Sec. 4) before concluding (Sec. 5).

# 2. SYSTEM OVERVIEW

The common denominator among the majority of the proposals and schemes (details in Table 1) in the literature is essentially a VPKI architecture with two main entities: the Long Term CA (LTCA) and the Pseudonym CA (PCA). The LTCA is responsible for issuing LTCs for the registered vehicles and is the *policy decision and enforcement point*: it authenticates and authorizes the vehicles. The PCA is the responsible authority for issuing the pseudonyms. In case of misbehavior, detected locally [20] or for other reasons [16], the Resolution Authority (RA) initiates a process to resolve, and possibly revoke, a pseudonym based on a set of pseudonymously authenticated messages. Without loss of generality, we follow this common understanding without dwelling on the details of each scheme. We adhere to the assumed adversarial behavior [19]. We extend it by assuming that the VPKI servers are *honest-but-curious*: they correctly execute system protocols and follow the system policies, but they are tempted to harm user privacy.

## 2.1 Pseudonym Acquisition Policies

The choice of policy for obtaining pseudonyms has diverse ramifications: on the VPKI performance and operation as well as the user privacy. The policy determines the volume of the workload (basically, pseudonym requests and related computation and communication latencies) imposed to the VPKI. On the other hand, the user privacy is at stake: a transcript of pseudonymously authenticated messages could be linked simply based on the pseudonym lifetime and issuance times [14], and requests could act as user *"fingerprints"*. Simply put, individually determined pseudonym lifetimes allow an observer to link pseudonyms of the same vehicle only based on timing information of the credentials (without even examining the content of the message). To systematically investigate the effect of diverse on-demand non-overlapping pseudonym acquisition methods, we define three representatives, summarized in Table 2.

Table 2: Summary of the Pseudonym Acquisition Policies

| *Notation* | *Policy Name* | *Limitations* | *V-VPKI Interactions* | *Privacy Preserving* |
|---|---|---|---|---|
| $P1$ | User-controlled (user-defined) | Each user should know the exact trip duration. | once per trip | × |
| $P2$ | Oblivious | Requesting for the last $\Gamma_{P2}$, the user should obtain pseudonyms for the entire duration. | $\left\lceil \frac{TripDuration}{\Gamma_{P2}} \right\rceil$ per trip | × |
| $P3$ | Universally fixed | Each user cannot obtain all credentials with a single request. Pseudonym lifetime should be a divisor/factor of the $\Gamma_{P3}$. Requesting for $\Gamma^i_{P3}$, each user should obtain pseudonyms for the entire $\Gamma^i_{P3}$. | $\left\lceil \frac{TripDuration}{\Gamma_{P3}} \right\rceil$ per trip | ✓ |

**User-controlled (user-defined) policy (P1):** A vehicle requests pseudonyms for its residual (ideally entire) trip duration at the start of trip. We presume each vehicle *precisely estimates* the trip duration in advance, e.g., based on automotive navigation systems, previous trips, or user input. The PCA determines the pseudonym lifetime, either fixed for all vehicles or flexible for each requester. Additional pseudonyms should be requested if the actual trip duration exceeds the estimated one to ensure that the vehicle is always equipped with enough valid pseudonyms.

**Oblivious policy (P2):** The vehicle interacts with the VPKI every $\Gamma_{P2}$ seconds (determined by the PCA and fixed for all users) and it requests pseudonyms for the entire $\Gamma_{P2}$ time interval until the vehicle reaches its destination. This results in over-provisioning of pseudonyms only during the last iteration.[1] The difference, in comparison to P1, is that either the vehicle does not know the exact trip duration, or, it does not attempt to estimate, or possibly, overestimate it; thus, P2 is oblivious to the trip duration.

**Universally fixed policy (P3):** The PCA has predetermined *"universally"* fixed interval, $\Gamma_{P3}$, and pseudonym lifetime, $\tau_P$. At the start of its trip, a vehicle requests pseudonyms for the *"current"* $\Gamma_{P3}$, out of which useful (non-expired) ones are actually obtained for the residual trip duration. For the remainder of the trip, the vehicle requests pseudonyms for the entire $\Gamma_{P3}$ at each time. This policy issues time-aligned pseudonyms for all vehicles; thus, timing information does not harm user privacy.

Fig. 1 illustrates the three pseudonym acquisition policies with respect to trip duration, pseudonym lifetime ($\tau_P$), and the PCA-determined periods ($\Gamma_{P2}$ and $\Gamma_{P3}$). Using P1 and P2, the exact time of requests and all subsequent requests until the end of trip could be unique, or one of few, and thus linkable even by an external observer; it might be unlikely in a specific region to have multiple requests at a specific instance. While in contrast for P3, the requesting intervals fall within the *"universally"* fixed interval ($\Gamma_{P3}$) and the issued pseudonyms are aligned with the global system time (PCA clock); therefore, at any point in time all vehicles in a given area will be transmitting under pseudonyms which are indistinguishable, one from another, based on timing information alone; thus, protecting user privacy.

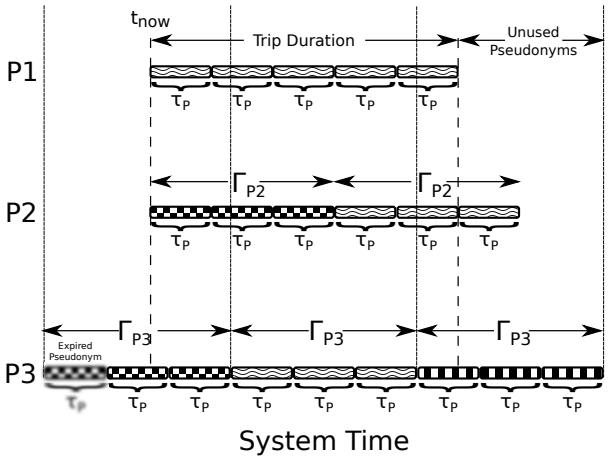



Figure 1: A Schematic Comparison of P1, P2, and P3

Table 3: Notation used in the protocols

| | |
|---|---|
| $P$, $P_v^i$, $(P_v^i)_{\sigma_{PCA}}$ | *current valid pseudonym signed by the PCA* |
| $(LK_v, Lk_v)$ | *long-term public and private key pairs* |
| $(K_v^i, k_v^i)$ | *current valid pseudonymous public and private key pairs* |
| $Id_{req}, Id_{res}$ | *request/response identifiers* |
| $Id_{CA}$ | *Certification Authority unique identifier* |
| $(msg)_{\sigma_v}$ | *a signed message with the vehicle's private key* |
| $N$ | *a nonce* |
| $t_{now}, t_s, t_e, t_{date}$ | *current, starting, ending, and a specific day timestamps* |
| $SN$ | *serial number* |
| $Exp_{tkt}$ | *ticket expiration time* |
| $H()$ | *hash function* |
| $Sign(Lk_{ca}, msg)$ | *signing a message with private key (Lk) of the CA* |
| $Verify(LTC_{ca}, msg)$ | *verifying a message with the CA's public key* |
| $\tau_P$ | *pseudonym lifetime* |
| $\Gamma_{Px}$ | *interacting period/interval with the VPKI for policy x* |
| $IK$ | *identifiable key* |
| $\zeta$ | *a temporary variable* |

## 3. PSEUDONYM ACQUISITION

For completeness and broader applicability, we extend and refine the common system model (Sec. 2) to enable multi-domain VPKI operation [14]. Our system works as follows: each registered vehicle interacts first with its LTCA to obtain a ticket. The LTCA authenticates and authorizes the requester, issuing a *service-granting ticket* for the requester, which enables the vehicle to obtain pseudonyms from any PCA. The trust establishment between the LTCA and the PCA is with the help of a higher-level authority or by using cross certification [15]. The vehicle, i.e., the OBU[2], decides when to trigger the pseudonym acquisition process based on different parameters, e.g., the number of remaining valid pseudonyms, the residual trip duration, and the networking connectivity. We presume connectivity to the VPKI (via RSUs, cellular networks, or other connectivities) to execute all of these protocols. Should the connectivity be intermittent, the OBU could initiate the protocols proactively when there is connectivity. Further discussion on a reliable connectivity to the VPKI is beyond the scope of this paper. The notation used in the protocols is given in Table 3.

**Ticket Acquisition Process (Protocols 1 and 2):** Assume the OBU decides to obtain pseudonyms from a spe-

[1]As an optimization, a vehicle could *roughly estimate* the residual trip duration, $D_v$; if $D_v \ll \Gamma_{P2}$, then it will request only for the $D_v$ period rather than the entire $\Gamma_{P2}$ interval; otherwise, it will request for the $\Gamma_{P2}$ interval. Here for simplicity, we do not consider this optimization.

[2]The terms vehicle and OBU are used interchangeably.

**Protocol 1** Ticket Request (from the LTCA)

1: **procedure** REQTICKET($P_x, \Gamma_{Px}, t_s, t_e, t_{date}$)
2: **if** $P_x = P1$ **then**
3: $(t_s, t_e) \leftarrow (t_s, t_e)$
4: **else if** $P_x = P2$ **then**
5: $(t_s, t_e) \leftarrow (t_s, t_s + \Gamma_{P2})$
6: **else if** $P_x = P3$ **then**
7: $(t_s, t_e) \leftarrow (t_{date} + \Gamma^i_{P3}), t_{date} + \Gamma^{i+1}_{P3})$
8: **end if**
9: $\zeta \leftarrow (Id_{tkt\text{-}req}, H(Id_{PCA} \| Rnd_{tkt}), t_s, t_e)$
10: $(\zeta)_{\sigma_v} \leftarrow Sign(Lk_v, \zeta)$
11: **return** $((\zeta)_{\sigma_v}, \text{LTC}_v, N, t_{now})$
12: **end procedure**

**Protocol 2** Issuing a Ticket (by the LTCA)

1: **procedure** ISSUETICKET($(msg)_{\sigma_v}, \text{LTC}_v, N, t_{now}$)
2: Verify($\text{LTC}_v, (msg)_{\sigma_v}$)
3: $IK_{tkt} \leftarrow H(\text{LTC}_v || t_s || t_e || Rnd_{IK_{tkt}})$
4: $\zeta \leftarrow (SN, H(Id_{PCA} \| Rnd_{tkt}), IK_{tkt}, t_s, t_e, Exp_{tkt})$
5: $(tkt)_{\sigma_{ltca}} \leftarrow Sign(Lk_{ltca}, \zeta)$
6: **return** $((tkt)_{\sigma_{ltca}}, Rnd_{IK_{tkt}}, N+1, t_{now})$
7: **end procedure**

cific PCA. If the relevant policy is P1, each vehicle *estimates* the actual trip duration $[t_s, t_e]$ (steps 1.2–1.3, i.e., steps 2–3 in protocol 1) while with P2, each vehicle requests pseudonyms for $[t_s, t_s+\Gamma_{P2}]$ (steps 1.4–1.5). If the relevant policy is P3, the vehicle calculates the trip duration based on the date of travel, $t_{date}$, and the exact time of travel corresponding to the universally fixed interval, $\Gamma_{P3}$, of that specific PCA (steps 1.6–1.7). It then calculates the hash value of the concatenation of the specific PCA identity with a random number; this conceals the identity of the PCA from the LTCA. The vehicle prepares the request (step 1.9) before signing it using the private key corresponding to its LTC (step 1.10), and returning the ticket request (step 1.11). It will then interact with the LTCA over a bidirectional authenticated Transport Layer Security (TLS).

Upon reception of the ticket request, the LTCA verifies the LTC (thus authenticating and authorizing the requester) and the signed message (step 2.2). The LTCA calculates the *"ticket identifiable key"* to bind the ticket to the LTC as: $H(\text{LTC}_v || t_s || t_e || Rnd_{IK_{tkt}})$ (step 2.3); this prevents a compromised LTCA from mapping a different LTC during the resolution process. The LTCA then encapsulates (step 2.4), signs (step 2.5), and delivers the response (step 2.6).

**Pseudonym Acquisition Process (Protocols 3 and 4):** Upon reception of a valid ticket, the vehicle generates required Elliptic Curve Digital Signature Algorithm (ECDSA) public/private key pairs (steps 3.2–3.6). It then prepares the request (step 3.7) and sends the pseudonym request to the PCA over a unidirectional authenticated TLS.

Having received a request, the PCA verifies the ticket, signed by the LTCA (assuming a trust is established) (steps 4.2–4.3). The PCA then verifies the pseudonym provider identity (step 4.4) and the requesting intervals for obtaining pseudonyms (step 4.5). Afterward, the PCA initiates a proof-of-possession protocol to verify the ownership of the corresponding private keys. Then, it calculates the *"pseudonym identifiable key"* to bind the pseudonyms to the ticket

**Protocol 3** Pseudonym Request (from the PCA)

1: **procedure** REQPSNYMS($t_s, t_e, (tkt)_{\sigma_{ltca}}$)
2: **for** i:=1 to **n do**
3: **Begin**
4: Generate($K^i_v, k^i_v$)
5: $(K^i_v)_{\sigma_{k^i_v}} \leftarrow \text{Sign}(k^i_v, K^i_v)$
6: **End**
7: $psnymReq \leftarrow (Id_{req}, Rnd_{tkt}, t_s, t_e, (tkt)_{\sigma_{ltca}}, \{(K^1_v)_{\sigma_{k^1_v}}, ..., (K^n_v)_{\sigma_{k^n_v}}\}, N, t_{now})$
8: **return** $psnymReq$
9: **end procedure**

**Protocol 4** Issuing Pseudonyms (by the PCA)

1: **procedure** ISSUEPSNYMS($psnymReq$)
2: $psnymReq \rightarrow (Id_{req}, Rnd_{tkt}, t_s, t_e, (tkt)_{\sigma_{ltca}}, \{(K^1_v)_{\sigma_{k^1_v}}, ..., (K^n_v)_{\sigma_{k^n_v}}\}, N, t_{now})$
3: Verify($LTC_{ltca}, (tkt)_{\sigma_{ltca}}$)
4: $H(Id_{this\text{-}PCA} \| Rnd_{tkt}) \stackrel{?}{=} H(Id_{PCA} \| Rnd_{tkt})$
5: $[t_s,\ t_e] \stackrel{?}{=} ([t_s, t_e])_{tkt}$
6: **for** i:=1 to **n do**
7: **Begin**
8: Verify($K^i_v, (K^i_v)_{\sigma_{k^i_v}}$)
9: $IK_{P^i} \leftarrow H(IK_{tkt} || K^i_v || t^i_s || t^i_e || Rnd_{IK^i_v})$
10: $\zeta \leftarrow (SN^i, K^i_v, IK_{P^i}, t^i_s, t^i_e)$
11: $(P^i_v)_{\sigma_{pca}} \leftarrow Sign(Lk_{pca}, \zeta)$
12: **End**
13: **return** $(\{(P^1_v)_{\sigma_{pca}}, \ldots, (P^n_v)_{\sigma_{pca}}\}, \{Rnd_{IK^1_v}, \ldots, Rnd_{IK^n_v}\}, N{+}1, t_{now})$
14: **end procedure**

as: $H(IK_{tkt} || K^i_v || t^i_s || t^i_e || Rnd_{IK^i_v})$. This essentially prevents a compromised PCA from mapping a different ticket during resolution process. It issues the pseudonyms (steps 4.6–4.12) and delivers the response (step 4.13).

# 4. PERFORMANCE EVALUATION

Due to the lack of a large-scale deployment of VC systems, we resort to realistic large-scale mobility traces, which determine the period the vehicles need pseudonyms. We extract two features of interest from the mobility traces, i.e., departure time and trip duration, and we apply policies described in Sec. 2.1 to assess the efficiency of the full-blown implementation of our VPKI for a large-scale deployment. The main metric is the *end-to-end pseudonym acquisition latency*, i.e., the delay from the initialization of protocol 1 till the successful completion of protocol 4 (steps 1.1–4.14), measured at the vehicle. The processing time to generate the key pairs (steps 3.2–3.6) is not considered here as the OBU can generate them off-line.

## 4.1 Experimental Setup

**VPKI Testbed:** We create a testbed comprising different Virtual Machines (VMs) allocated to distinct VPKI servers; Table 4 details the specifications for the servers and the emulated client[3]. Our implementation is in C++ and we

[3]The processing power of the client is comparable to the Nexcom boxes (dual-core 1.66GHz, 2GB memory) in PRESERVE project [2] as we execute all clients in one VM.

Table 4: Servers and Clients Specifications

| | LTCA | PCA | RA | Client |
|---|---|---|---|---|
| Number of entities | 1 | 1 | 1 | 1 |
| Dual-core CPU (Ghz) | 2.0 | 2.0 | 2.0 | 2.0 |
| BogoMips | 4000 | 4000 | 4000 | 4000 |
| Memory | 2GB | 2GB | 1GB | 1GB |
| Database | MySQL | MySQL | MySQL | MySQL |

Table 5: Mobility Traces Information

| | TAPASCologne | LuST |
|---|---|---|
| Number of vehicles | 75,576 | 138,259 |
| Number of trips | 75,576 | 287,939 |
| Duration of snapshot (hour) | 24 | 24 |
| Available duration of snapshot (hour) | 2 (6-8 AM) | 24 |
| Average trip duration (sec.) | 590.49 | 692.81 |

use OpenSSL for the cryptographic protocols and primitives, i.e., TLS and ECDSA-256 (according to the standards [9, 13]). We run the experiments in our testbed with VPKI servers and clients (emulating OBUs) running on the VMs. This set up eliminates the network propagation delays of OBU-VPKI connectivity. Depending on the actual OBU-VPKI connectivity, the network propagation delays would vary; for simplicity, we do not consider it here.

**Mobility Traces:** To have realistic arriving requests to the VPKI, we used two microscopic mobility vehicle datasets: TAPASCologne [22] and Luxembourg SUMO Traffic (LuST) [7], detailed in Table 5. The former one represents the traffic demand information across the Köln urban area (available for 2 hours, 6-8 AM) while the latter presents a full-day realistic mobility pattern in the city of Luxembourg.

## 4.2 VPKI Servers Performance

Figs. 2-4 show the interplay between the end-to-end latency, averaged over all completed protocol executions within each minute period, and different pseudonym acquisition policies (with different configurations) for the two datasets. Table 6 details end-to-end latency statistics for each policy. With P1 (Fig. 2), each vehicle requests all required pseudonyms at once; with $\tau_P = 0.5$ min., 99% of the requesters for TAPASCologne and LuST datasets are served within less than 153 ms and 167 ms respectively. As it is shown in Fig. 2.b, the end-to-end latency with P1 follows the arrival distribution and it is fluctuating over time; the reason is that with P1, vehicles can request for any trip duration, thus requesting more pseudonyms at once.

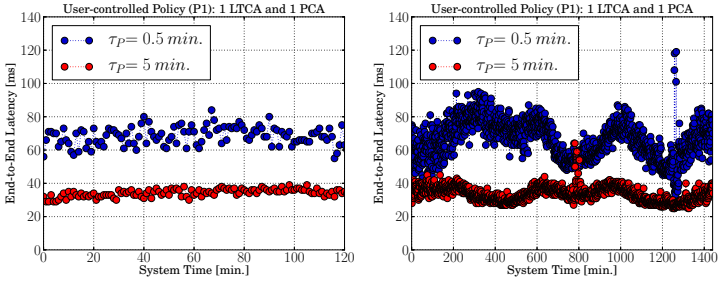
With P2 (Fig. 3), the vehicles request a fixed amount of pseudonyms every time (for a duration of $\Gamma_{P2}$=5 min.), thus never overloading the PCA server with large amount of pseudonyms acquisition in a single request; this results in a low standard deviation and variance, and a smooth end-to-end latency in compare with other policies. The end-to-end latency for the TAPAS and LuST datasets ($\tau_P$=0.5 min.) is 50 ms and 45 ms respectively; accordingly, 99% of vehicles are served within less than 109 ms and 80 ms respectively.

With P3 (Fig. 4), the system enforces synchronized batch arrivals to obtain pseudonyms: each vehicle requests pseudonyms for the entire $\Gamma_{P3}$, timely aligned with the rest. The end-to-end latency for the two datasets ($\tau_P$=0.5 min.) is 45 ms and 47 ms respectively; moreover, 99% of the requesters are served within less than 70 ms and 74 ms respectively.

The results confirm that our secure and privacy preserving scheme efficiently issues pseudonyms for the requesters; thus, an OBU can initiate a request for pseudonyms within the lifetime of the last single valid pseudonym. We can conclude that modest VMs can serve very large number of vehicles even during the most harsh traffic conditions with very low delays, and the most promising policy in terms of privacy protection incurs moderate overhead. The choice of parameters for $\Gamma_{P2/P3}$ and $\tau_P$ mainly determines the frequency of interaction with the VPKI and the volume of workload imposed to the PCA: the shorter the pseudonym lifetimes are, the greater number of pseudonyms will be requested, thus a higher workload is imposed on the PCA.

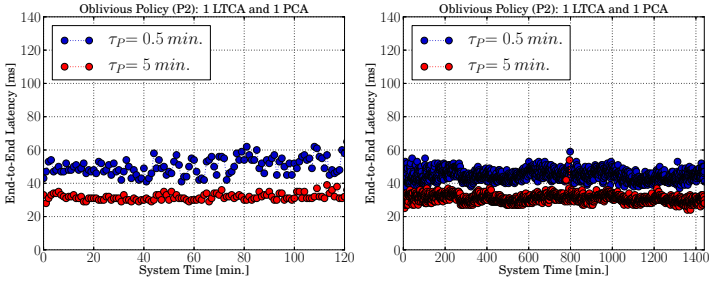
Table 6: Latency Statistics for each Policy ($\Gamma$ = 5 min., $\tau_P$ = 0.5 min.)

| | TAPAS-P1 | TAPAS-P2 | TAPAS-P3 | LuST-P1 | LuST-P2 | LuST-P3 |
|---|---|---|---|---|---|---|
| **Maximum (ms)** | 426 | 268 | 4254 | 504 | 248 | 3408 |
| **Minimum (ms)** | 17 | 26 | 18 | 15 | 25 | 20 |
| **Average (ms)** | 69 | 50 | 45 | 69 | 45 | 47 |
| **Std. Deviation** | 26 | 17 | 23 | 30 | 12 | 21 |
| **Variance** | 708 | 295 | 535 | 895 | 138 | 449 |
| **Pr{t ≤ x} = 0.99 (ms)** | 153 | 109 | 70 | 167 | 80 | 74 |



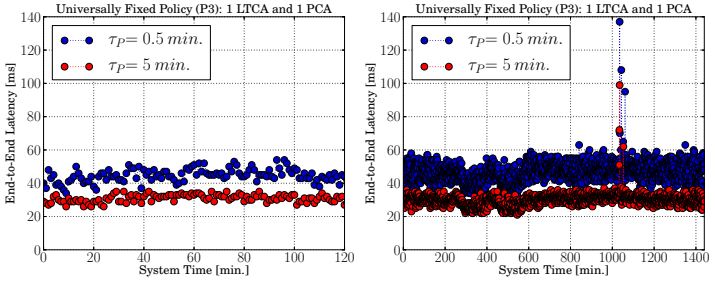

(a) TAPASCologne dataset (b) LuST dataset

Figure 2: End-to-end latency for P1




(a) TAPASCologne dataset (b) LuST dataset

Figure 3: End-to-end latency for P2, $\Gamma_{P2}$=5 minutes




(a) TAPASCologne dataset (b) LuST dataset

Figure 4: End-to-end latency for P3, $\Gamma_{P3}$=5 minutes

As the results show, issuing pseudonyms with very short lifetimes (30 sec.) does not have a high impact on the overall performance of the servers. The results presented here are obviously dataset-dependent; however, by understanding the characteristics of the mobility, i.e., the road-constrained movements, the appearance of the RSUs, vehicle movement direction, and sudden bursts of traffic, system designers can evaluate the impact of a given mobility trace on the deployment and dimensioning of the VPKI resources.

# 5. DISCUSSION AND CONCLUSION

**Remark on the suitability of non-overlapping pseudonym lifetimes for safety applications:** As mentioned earlier, safety applications can operate more easily if there is linkability (with the vehicle keeping the same pseudonym) during a critical situation. In such a case, e.g., emergency braking or collision avoidance, the vehicle can simply include a link to its previous pseudonym. In fact, the vehicle could even sign with two $k_v^{i-1}$ and $k_v^i$ private keys, corresponding to pseudonyms $P_v^{i-1}$ and $P_v^i$. This would ensure the operation of the safety application (partial linkability).

**Summary and future work:** In this paper, we specified three policies for pseudonym acquisition, drawing from the literature. We integrated those into the pseudonym acquisition process of the state-of-the-art VPKI system [14]. To the best of our knowledge, our system [14] is the latest and fastest VPKI. Nonetheless, our investigation is relevant to any VPKI that relies on non-overlapping on-demand pseudonyms acquisition. We presented a secure and efficient solution for pseudonyms acquisition while the timing information cannot harm user privacy. Through experimental evaluation, we demonstrated that modest VMs dedicated as servers can serve on-demand requests with very low delay, and the most promising policy in terms of privacy protection incurs moderate overhead.

Using P1, a vehicle interacts with the VPKI servers once to obtain the necessary pseudonyms for the entire trip duration (ideally without over-provisioning). However, according to P2 and P3, vehicles could be potentially equipped with more pseudonyms than needed, i.e., the PCA might issue pseudonyms for a period during which the vehicle will not use them. In general, the longer the pseudonym refill interval ($\Gamma_{P2}$ or $\Gamma_{P3}$) is, the less frequent vehicles-VPKI interactions, but the higher the chance to overprovision a vehicle. As future work, we will investigate the pseudonym utilization with various configurations ($\Gamma_{P2/P3}$ and $\tau_P$) to investigate the interplay with the server workload and privacy protection (the shorter $\tau_P$, the less linkable are messages by a vehicle). We further intend to rigorously analyze the security and privacy protocols and evaluate the level of privacy, i.e., unlinkability, based on the timing information of the pseudonyms for each policy.

# 6. REFERENCES


[1] American Community Survey. https://www.census.gov/programs-surveys/acs/.

[2] Preparing Secure Vehicle-to-X Communication Systems - PRESERVE. http://www.preserve-project.eu/.

[3] U.S. Department of Transportation (DoT). Safety Pilot Model Deployment. http://safetypilot.umtri.umich.edu/.

[4] N. Alexiou et al. VeSPA: Vehicular Security and Privacy-preserving Architecture. In *ACM HotWiSec*, Budapest, Hungary, Apr. 2013.

[5] N. Bißmeyer et al. CoPRA: Conditional Pseudonym Resolution Algorithm in VANETs. In *IEEE WONS*, Banff, Canada, Mar. 2013.

[6] Car-to-Car Communication Consortium (C2C-CC). http://www.car-2-car.org/.

[7] L. Codeca et al. Luxembourg Sumo Traffic (LuST) Scenario: 24 Hours of Mobility for Vehicular Networking Research. In *IEEE VNC*, Kyoto, Japan, Dec. 2015.

[8] J. R. Douceur. The Sybil Attack. In *ACM Peer-to-peer Systems*. London, UK, Mar. 2002.

[9] ETSI. Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions, Jun. 2009.

[10] L. Fischer et al. Secure Revocable Anonymous Authenticated Inter-vehicle Communication (SRAAC). In *ESCAR*, Berlin, Germany, Nov. 2006.

[11] D. Förster et al. PUCA: A Pseudonym Scheme with User-Controlled Anonymity for Vehicular Ad-Hoc Networks (VANET). In *IEEE VNC*, Paderborn, Germany, Dec. 2014.

[12] S. Gisdakis et al. SEROSA: SERvice Oriented Security Architecture for Vehicular Communications. In *IEEE VNC*, Boston, MA, USA, Dec. 2013.

[13] IEEE P1609.2/D12. Draft Standard for Wireless Access in Vehicular Environments, Jan. 2012.

[14] M. Khodaei et al. Towards Deploying a Scalable & Robust Vehicular Identity and Credential Management Infrastructure. In *IEEE VNC*, Paderborn, Germany, Dec. 2014.

[15] M. Khodaei et al. The Key to Intelligent Transportation: Identity and Credential Management in Vehicular Communication Systems. *IEEE VT Magazine*, 10(4), Dec. 2015.

[16] P. Papadimitratos. "On the road" - Reflections on the Security of Vehicular Communication Systems. In *IEEE ICVES*, Columbus, OH, USA, Sep. 2008.

[17] P. Papadimitratos et al. Secure Vehicular Communication Systems: Design and Architecture. *IEEE CommMag*, 46(11):100–109, Nov. 2008.

[18] P. Papadimitratos et al. Vehicular Communication Systems: Enabling Technologies, Applications, and Future Outlook on Intelligent Transportation. *IEEE CommMag*, 47(11):84–95, Nov. 2009.

[19] P. Papadimitratos et al. Securing Vehicular Communications-Assumptions, Requirements, and Principles. In *ESCAR*, Berlin, Germany, Nov. 2006.

[20] M. Raya et al. Eviction of Misbehaving and Faulty Nodes in Vehicular Networks. *IEEE JSAC*, Oct. 2007.

[21] F. Schaub et al. V-tokens for Conditional Pseudonymity in VANETs. In *IEEE WCNC*, NJ, USA, Apr. 2010.

[22] S. Uppoor et al. Generation and Analysis of a Large-scale Urban Vehicular Mobility Dataset. *IEEE Transactions on Mobile Computing*, May 2014.

[23] W. Whyte et al. A Security Credential Management System for V2V Communications. In *IEEE VNC*, Boston, Dec. 2013.