Future-proofing Secure V2V Communication against Clogging DoS Attacks

Hongyu Jin
Networked Systems Security Group
KTH Royal Institute of Technology
Stockholm, Sweden
hongyuj@kth.se

Zhichao Zhou
Networked Systems Security Group
KTH Royal Institute of Technology
Stockholm, Sweden
zhizho@kth.se

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

ABSTRACT

Clogging Denial of Service (DoS) attacks have disrupted or disabled various networks, in spite of security mechanisms. External adversaries can severely harm networks, especially when high-overhead security mechanisms are deployed in resource-constrained systems. This can be especially true in the emerging standardized secure Vehicular Communication (VC) systems: mandatory message signature verification can be exploited to exhaust resources and prevent validating incoming messages sent by neighboring vehicles, information that is critical, often, for transportation safety. Efficient message verification schemes and better provisioned devices could serve as potential remedies, but existing solutions have limitations. We point out those and identify, challenges to address for scalable and resilient secure VC systems, and, most notably, the need for integrating defense mechanisms against clogging DoS attacks. We take the position that existing secure VC protocols are vulnerable to clogging DoS attacks and recommend symmetric key chain based pre-validation with mandatory signature verification to thwart clogging DoS attacks, while maintaining all key security properties, including non-repudiation to enable accountability.

CCS CONCEPTS

• **Security and privacy → Denial-of-service attacks**; **Security protocols**; *Distributed systems security*; Privacy-preserving protocols.

KEYWORDS

Security, privacy, vehicular communication, pseudonymous authentication

1 INTRODUCTION

Standards and research on secure Vehicular Communication (VC) systems, or Vehicular Ad-hoc Networks (VANETs), propose pseudonymous authentication that protects privacy while satisfying security requirements, notably authentication, integrity, and non-repudiation (thus accountability) for two, basically, standardized types of Vehicle-to-Vehicle (V2V) messages: Cooperative Awareness Messages (CAMs) and Decentralized Environmental Notification Messages (DENMs) [6, 13, 24]. CAMs are periodical, with a typical frequency of 10 Hz, and they are safety messages informing neighboring vehicles about the sender mobility status. DENMs are event-driven, reporting events that affect nearby vehicles. CAMs are usually less critical, while DENMs could be usually immediately relevant to driver safety. Pseudonymous authentication of CAMs and DENMs relies on the Elliptic Curve Digital Signature Algorithm (ECDSA), with standard Elliptic Curves (ECs), such as nistp256, brainpoolP256r1 and brainpoolP384r1 [7]. Vehicles digitally sign CAMs and DENMs, and the receiving vehicles validate those signatures with the use of EC public keys obtained through 'bare bones' ephemeral Pseudonymous Certificates (PCs) attached to the messages [23].

Cryptographic operations, especially signature verification, essential for VC security, introduce significant computational overhead. This precisely can be exploited for clogging Denial of Service (DoS) attacks that exhaust computational resources of vehicular On-Board Units (OBUs); together with the inherent amplification opportunity VC offers, as practically every CAM or DENM transmission must be processed and validated by all (typically, tens) of neighboring vehicles. High-power radio jamming could trivially lead to DoS, through a brute-force attack disrupting communication. In contrast, we emphasize that even a modest increase of carefully crafted bogus network traffic that appears "legit" could already result in DoS (see Sec. 2). The challenge is especially interesting because clogging DoS defense mechanisms widely deployed in the Internet would not be immediately applicable for VC systems; e.g., reCAPTCHA [32] requires human interactions with seconds of delays.

Optimizations [11, 20, 26, 28, 35] relieve security overhead to a certain extent but they do not fundamentally solve the problem: they are only effective for message verification after an initial benign message signature verification per sender. Ample processing power (often in the form of hardware acceleration) is usually considered the end-solution to handle excessive security overhead. However, this is far from straightforward, because it assumes an unrealistic asymmetry in the evolution of computation power of benign system nodes and (brute-force) attackers, i.e., the computation power is an important factor for defining optimal security level to counter brute-force attacks on cryptographic keys that eventually affects computation overhead for benign entities. In fact, in this paper, we argue it is a fallacy to expect that secure and privacy-preserving VC will remain resilient to clogging DoS by simply provisioning the OBUs with more processing power. In contrast, we take the position that any such natural evolution in the future must hinge upon by-design protection against clogging DoS attacks.

2 BACKGROUND AND PROBLEM STATEMENT

2.1 Pseudonymous Authentication

Pseudonymous authentication with short-term credentials and public key cryptography ([1, 13, 14, 16, 24]) is the current standard. A curious observer, having eavesdropped messages digitally signed

with two different private keys corresponding to two different public keys on two corresponding PCs, cannot link those based on signatures and PCs. ECDSA signatures are used instead of the more popular, e.g., in Internet/Web, RSA ones, because EC key sizes and ECDSA signatures are much smaller than those for RSA, for the same security level [1]; thus ECDSA provides lower communication overhead, but at the expense of higher signature verification delay than RSA. Upon a message reception, a receiving vehicle (OBU) verifies the PC signature (if not verified previously and cached the result) with the certificate of the Certification Authority (CA) that issued the PC and verifies the message signature with the PC [14–16].

2.2 V2V Communication

An important component of Vehicle-to-Everything (V2X) communication is V2V communication, which provides a direct communication channel for vehicles to exchange CAMs and DENMs. Dedicated Short Range Communication (DSRC) and Cellular-V2X are two technologies to enable V2V communication. DSRC operates on the 5.9 *GHz* band and is designed specifically for automotive applications. It provides low-latency communication with high reliability, making it suitable for safety-critical V2V communication. DSRC has been standardized as IEEE 802.11p and is adopted by ETSI as the V2X communication standard, with the default bandwidth of 6 *Mbps* and a maximum 27 *Mbps*. IEEE 802.11bd is proposed as an evolution of IEEE 802.11p with support for potentially significantly higher network bandwidth [31]. Cellular-V2X (C-V2X) is an extension of existing cellular networks (such as 4G LTE, 5G, and upcoming 6G) to support V2X communication [9]. It offers broader coverage, seamless integration with existing cellular infrastructure, and support for high-bandwidth applications. C-V2X can support both direct communication (similar to DSRC) and network-based communication, enabling V2V communication as well as Vehicle-to-Infrastructure (V2I) communication. Our problem in hand, described below, is orthogonal to any chosen communication channel (e.g., DSRC or C-V2X), but stems from cryptographic overhead from received messages and could be aggravated by higher network bandwidth introduced with advanced communication technologies in the future.

2.3 Clogging DoS Attacks

We are concerned with clogging DoS attacks that exploit the relatively heavy computational overhead of ECDSA signature verification. The challenge lies in the inherently high cryptographic overhead per message, exacerbated by the communication pattern. Each sent CAM or DENM must be digitally signed, and each received CAM or DENM must be cryptographically verified (at very least one signature verification). Given the broadcast and safety-critical nature of most VC network traffic, within a short period, each OBU must cryptographically verify CAMs (at all times) or DENMs (occasionally) from several nearby vehicles. A typical message verification requires two signature verifications on both the PC and the message, i.e., a verification of PC based on the certificate of the PC issuing CA and a verification on the message signature based on the PC. The increased ECDSA verification cost, compared to RSA, adds to this imbalance; even though each receiving vehicle could validate each PC only once and locally cache the result [3]. Still, the latter improvement would be significant yet limited by the PC lifetime and the volatility of the topology. Neighboring vehicles establish short-term trust based on periodically changing PCs for (often) ephemeral encounters without maintaining long-term relations with each other. Even two neighboring vehicles would not recognize each other after the PCs in use change, thus it would be impossible to establish trust in advance.

This opens up VC systems to clogging DoS attacks against vehicles in proximity. An adversary could flood the network with seemingly valid and well-formatted messages while attaching bogus signatures (for both PCs and messages). Receivers have to verify the signatures before revealing the invalidity of the bogus messages. Vehicles are allowed to change their MAC and IP addresses randomly together with their changing PCs [6]. Therefore, it is infeasible to impose rate control on the adversary based on its network addresses, because the adversary can also set different MAC and IP addresses for each bogus message. Internal adversaries can launch clogging DoS with valid signatures (at the expense of valid signature generation, typically computationally heavier than verification) but it would be trivial to verifiably link them to the offending sender, exceeding the expected rates by far.

More powerful processors could provide much lower delays to counter bogus messages and dense neighborhood, but they would significantly increase power consumption. Especially due to the popularization of electric cars, low-power processors (e.g., ARM processors) are preferable due to their relative power consumption efficiency. A recent hardware acceleration [22] for ECDSA claims that a brainpoolP256r1 signature verification takes around 1 *ms* with a standard double-and-add point multiplication [10]. The delay is close to the benchmarks on Raspberry Pi 4 with the OpenSSL library, which we use for an experimental evaluation of a prototype implementation of a CAM application. Table 1 shows the corresponding security level for the cryptographic benchmarks run on a Raspberry Pi 4, with an ARM Cortex-A72 processor. Comparing the two Brainpool curves from the ETSI standard for VC security [7], 384-bit verification delay (3.5 *ms*) is more than 2.5 times of that for 256-bit verification (1.3 *ms*). However, the verification delay with nistp256 (0.3 *ms*) is significantly lower. This is due to the optimization only applicable for nistp256: Brainpool curves use random primes that provide better security, while NIST curves use quasi-Mersenne primes [21, 27]. In our experiment, we choose the two Brainpool curves, due to improved security that could result in eventual migration to Brainpool for PCs [25].

Table 1: OpenSSL Benchmarks on Raspberry Pi 4

Security Level	*ECDSA*	*Sign (ms)*	*Verify (ms)*
128 bits	brainpoolP256r1	1.6	1.3
192 bits	brainpoolP384r1	4.5	3.5
256 bits	brainpoolP512r1	8.8	6.4
128 bits	nistp256	0.1	0.3
192 bits	nistp384	4.5	3.3
256 bits	nistp521	11.8	8.2

We perform an experimental study with two Raspberry Pis. One device emulates both legitimate senders and a clogging DoS attacker. Each instance of legitimate sender broadcasts CAMs at 10 Hz. The instance of the DoS attacker broadcasts bogus CAMs at a modest (for an attack) rate of 150 Hz. The bogus CAMs are crafted with bogus signatures that are simply random bits. Our second device receives the CAMs and verifies them in a First-Come First-Served (FCFS) manner. We use wireline communication not for convenience, but due to the lack of VC standard IEEE 802.11p/bd support by Raspberry Pi, to abstract away the wireless PHY layer and only focus on the cryptographic processing overhead. Each experiment is run for 2 min, and we use the results from the second minute. Results for each setting are averaged over three repeated runs.

Figure 1 shows a comparison between numerical analysis considering an M/D/1 queue [3] and experimental results. The average waiting time (or system time), T, total time in the queue until a CAM is verified, can be represented as:

$$T = S + \frac{\lambda S^2}{2(1-\lambda S)}, \tag{1}$$

where S is the service time (i.e., ECDSA signature verification delay) and λ is the aggregate CAM arrival rate. The service times of the queue are estimated with 1.3 ms and 3.5 ms for 256-bit and 384-bit curves respectively (Table 1).

In the benign scenario, we show the results with at most 60 senders and 25 senders for the two curves respectively, because the number of senders beyond those values would make the queue not stationary, i.e., queue size and waiting time would grow to infinity. The bogus CAM rate is equivalent to an aggregate rate of 15 benign senders in the DoS scenario, thus the queue can handle 15 less senders. More aggressive DoS rates would make the system performance even worse. Of course, much higher neighborhood density in reality than those considered in the experiments would aggravate the situation too. The aggregate 600 Hz CAMs from 60 senders corresponds to 1.44 $Mbps$, assuming 300 $byte$ CAMs, a fraction of the IEEE 802.11p default data rate (6 $Mbps$). Experimentally measured waiting times are generally higher than the optimistic numerical estimation, due to the message arrival not being exactly a Poisson process, and the actual service times in the experiments not being exactly deterministic (i.e., they fluctuate around the experimentally measured ECDSA verification delays, 1.5 ms and 3.7 ms, for the two curves respectively).

2.4 Distinction from Radio/MAC-layer Jamming

Jammers, at the radio or medium access control (MAC) layer, can constantly transmit random bits to cause collisions that force repeated backoffs by senders (kept in receive state) [34]. They can practically consume network bandwidth and degrade communication (prevent reception practically within range of the jammer), however, such jammers should nearly continuously transmit (possibly with rather higher power than legitimate transmissions), which result in high energy expenditure for the attacker. Moreover, jamming approaches could be totally different to target different MAC layer protocols (e.g., 5G).

In contrast, clogging DoS attacks are agnostic to the underlying MAC layers, leveraging bogus messages disseminated with regular

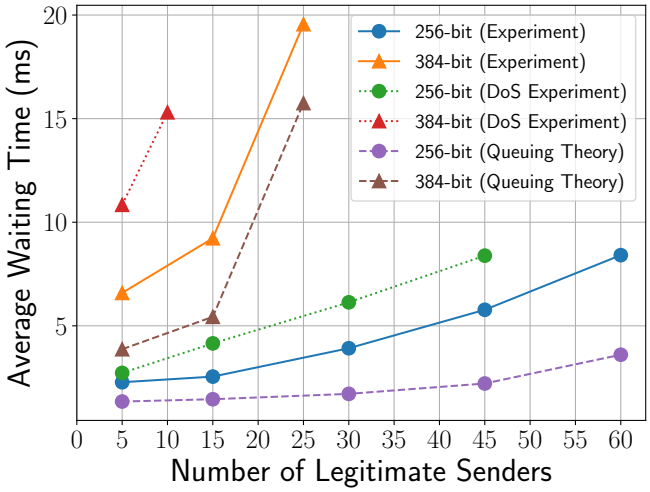

Figure 1: Evaluation results with the Brainpool curves.

frames. A clogging DoS attack can be mounted even with off-the-shelf devices, identical to those legitimate nodes use, only requiring adversarial messages that appear as "legit". It can effectively deny benign V2V message acceptance well ahead of exhausting network bandwidth, with only a fraction of the bandwidth (of a jamming attack) transmitting minimally more complex (i.e., specially crafted) traffic that majorly occupies computation resources with bogus message verification (and rejection). The amplification factor becomes apparent when one considers that it requires microseconds for the attacker to send a packet (CAM or DENM in this case), while it costs milliseconds to verify at each victim receiver.

3 EXISTING SOLUTIONS

We discuss research that seeks to reduce the security overhead of pseudonymous authentication. We are mostly interested in the optimization of the computational overhead, while communication overhead, although important, is out of the scope of our discussion.

3.1 Probabilistic Verification

Probabilistic verification preserves computational resources by performing signature verifications on a proportion of messages selected based on context or probabilistically [26, 35]. Adaptive Message Verification (AMA) proposes verifying each message probabilistically, and only switches to check all (i.e., 100% probability) signatures when any forged signatures are detected [26]. Another probabilistic approach [35] in the same spirit reports false signature detection to neighboring vehicles so that receivers of such reports check signatures of the corresponding messages. However, such approaches can be easily exploited locally by clogging DoS attackers that disseminate forged messages, fast pushing to the fallback, the baseline that verifies all message signatures - but protects the system by preventing the acceptance of the offending traffic.

3.2 Integration of Symmetric Key Cryptography

Symmetric key cryptography usually provides significantly lower computational overhead (thus lower delays) than asymmetric key

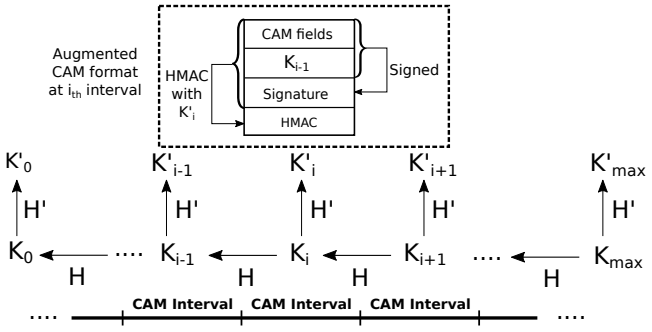

Figure 2: CAM augmentation with HMAC based on one-way key chain. H and H' are two hash functions.

cryptography. This property is usually exploited by a lot of research to provide efficient authentication on top of digital signatures (public key cryptography). The fundamental idea behind these approaches is to provide authentication based on Hash-based Message Authentication Codes (HMACs) [11, 12, 20, 28]. For example, solutions based on Timed Efficient Stream Loss-Tolerant Authentication (TESLA) use a pre-generated one-way hash/key chain as symmetric authentication keys to CAMs.

Figure 2 shows an example of one-way key chain generation and augmented CAM format that supports both efficient HMAC-based authentication and non-repudiable signature-based authentication. Each CAM is augmented with a symmetric key from the key chain and the signed CAM is further authenticated with a HMAC, that can be verified based on a late-disclosed key chain element with follow-up CAMs. Signature verification of any CAM from the chain is sufficient to trust the follow-up keys from the same chain. Validation of either signature or HMAC is sufficient to accept the CAM. However, an internal attacker could exploit the lack of non-repudiation (thus accountability) in a symmetric key based solution to inject bogus CAMs (attached valid HMACs but bogus signatures) to disrupt system functionalities (see Sec. 4).

3.3 Cooperative Verification

Due to the broadcast nature of V2V communication, neighboring vehicles would receive a majority of the same messages that they all have to verify. Cooperative verification takes advantage of frequent CAM exchanges to share message verification results in the form of message hashes [12, 18]. A message verification would potentially help verifying multiple extra messages the message sender had already verified. Therefore, the processing rate of queued messages would be higher than independently verifying them. However, finding and verifying benign messages, among high rate bogus messages, to make use of shared verification results is still an issue when under clogging DoS attacks.

3.4 Hardware Acceleration

Dedicated cryptographic hardware could expedite signature verification. For example, FPGAs can be configured to support specific algorithms [2, 22]. With a latest hardware implementation of ECDSA, signature verification latency for the brainpoolP256r1 curve is roughly 1 *ms* with the standard double-and-add approach, and 0.7 *ms* with optimizations (e.g., pre-computations) specific to the curve. However, hardware-based solutions have limitations (see Sec. 4).

3.5 Physical Layer Fingerprinting

Physical layer or radio frequency fingerprinting [5] can correlate the (high rate) bogus messages from a same adversarial transmitting device based on their similar physical layer characteristics, stemmed from hardware imperfection (e.g., carrier frequency offset), in order to efficiently filter them out. However, there is no significant evidence that the fingerprinting can be done efficiently (especially for moving targets), which usually requires high-end software-defined radio and computationally heavy fingerprint estimation. Last but not least, physical layer obfuscation [8] exists that could potentially invalidate the solutions relying on physical layer fingerprinting.

3.6 Solutions for Other Domains

Recent approaches for defending against DoS attacks exploit human effort or client computation resource to solve simple puzzles. Captcha-based solutions require user operation to complete some easy tasks hard to machine automate [32]. It is straightforward that requiring human effort in frequent V2V communication is impractical, especially when driving a car. Clients can be also presented with puzzles or challenges that require substantial computation resource to come up with solutions that need to be presented together with connection request, thus discouraging attackers from attempting DoS attacks [19, 29]. However, these solutions only work for pairwise connections, e.g., client-server connection or V2I communication; they are inappropriate for V2V communication.

4 POSITION: EXISTING SECURE (AND PRIVACY-ENHANCING) VC PROTOCOLS ARE VULNERABLE TO CLOGGING DOS ATTACKS

We argue that existing potential solutions against clogging DoS, introduced in Sec. 3, fall short in terms of addressing the problem at hand and maintaining the necessary security (and privacy) protection level, as per VC standards and the research state of the art. We explain why this is so from several perspectives.

4.1 Non-repudiation and Accountability

Non-repudiation is a corner-stone for accountability in VC systems. However, symmetric-key based authentication alone would outright forgo non-repudiation (symmetric keys known by multiple entities). Signed messages (in Figure 2) can be verified based on signatures if necessary, e.g., upon misbehavior detection. However, even if symmetric-key based authentication is efficient in a benign network, signature verification as fallback is equally vulnerable to clogging DoS *and* it disables misbehavior attribution.

Figure 3 shows an example of bogus CAM dissemination that exploits the lack of non-repudiation. An internal attacker can forge a CAM with a bogus signature and a valid HMAC. A resource-constrained receiver would only validate HMAC and accept the

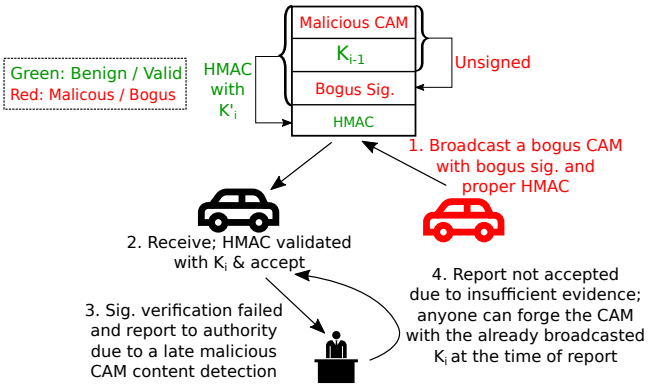

Figure 3: An attacker exploits the lack of non-repudiation in the symmetric key chain based solution.

CAM. A late signature verification, assuming the successful detection of the malicious CAM content based on data validation approaches, would reveal the CAM invalidity. Although the receiver is certain the CAM was generated by the attacker that previously authenticated the key chain (that involves K_i), the authority does not have sufficient evidence to deem the key chain owner malicious - K_i is already public at the time of malicious CAM reporting and anyone could have forged such a CAM.

Pairwise symmetric key establishment protocols would reduce the risk of revealing symmetric keys to attackers. However, they would require asymmetric cryptography (e.g., TLS handshakes). This would introduce significant overhead upon neighbor discovery, especially because of dense network neighborhoods and, more so, the dynamicity of vehicle connectivity (topology). Worse even, the key establishment protocols could be exactly targeted by clogging DoS attackers.

In cooperative verification [12, 18], an adversary could disseminate a properly signed message in an attempt to validate other previously transmitted (by itself or an accomplice) bogus messages, claiming it verified them. Although the adversary could be held accountable when detected (recall: message acceptance always hinges upon signature verification), an effective misbehavior detection approach is required to mitigate the vulnerability.

4.2 Neighboring Vehicle Discovery

Efficient verification takes advantage of periodicity and correlation across successive CAMs. This assumes neighboring vehicles already discovered based on pseudonymous authentication - a prerequisite for efficient CAM verification. For V2V communication, this has to be achieved based on message signature verification. Once a message signature is verified, subsequent messages by the same sender can be verified based on efficient symmetric key approaches. However, the challenge is how to discover and verify that first digitally signed message, when there could be plenty of bogus messages in the queue when under DoS attack. The same challenge arises for cooperative verification, because benign messages have to be verified to make use of the shared verification results. Existing works (in Sec. 3) only address efficient verification of messages from already known and trusted senders. Resilient neighboring vehicle discovery when under a DoS attack, a basis for efficient message verification, remains an open challenge.

4.3 Decentralized Environmental Notification Messages

Compared to high-frequency CAMs, DENMs (or generally event-driven messages) are relatively less regular, and triggered only by certain events. Similar to neighbor vehicle discovery based on CAMs, DENMs cannot be verified leveraging periodicity (exploited for efficient CAM verification), because the timing and content could be very diverse. Moreover, due to usually higher (transportation safety) criticality, DENMs have to be strictly verified based on signatures, instead of alternative efficient approaches. An adversary flooding with 'highly critical' bogus DENMs could easily clog victim vehicles.

4.4 Hardware Acceleration

Dedicated hardware supports faster cryptographic operations. But it could be costly and usually can be configured/programmed to support simultaneously only one or a very limited number of algorithms. They need to be manually updated to support new algorithms, at the expense of removal of earlier algorithms. Hardware support is not easily applicable to any cryptographic primitive. For example, to the best of our knowledge, no hardware acceleration is available for 384-bit or higher Brainpool curves so far. Therefore, solutions merely relying on hardware acceleration could not scale as time progresses, and could also possibly be invalidated by the security level increase.

Powerful hardware can be considered a solution only when cryptographic hardware is universally available in the VC system; because vehicle safety is equally important for all vehicles, with powerful or budget devices alike. However, reaching such a point implies the same (or even more powerful) hardware would be readily available to attackers. Consequently, this would lower the difficulty of brute force attacks. Thus, the evolution of hardware capabilities (e.g., better hardware acceleration and more powerful processors) would also necessitate the increase of cryptographic security.

4.5 Cryptographic Security Level

The security level reflects the difficulty of brute-force attacks; for example, an algorithm with 128-bit security requires maximum 2^{128} trials to break the key/system. The rationale behind mandated increasing security levels is to keep the cost/speed of brute force attacks relatively constant while the available processing power continuously increases [17]. For example, the ETSI standard has recently updated supported ECs from 224/256 bits to 256/384 bits [7]. With the security level increase, computational delays for benign system nodes will increase too, i.e., the delays will revert to a level prior to the processing power upgrade and the security level increase. This implies that solutions relying on provisioning OBUs with more powerful hardware do not fundamentally solve the problem.

4.6 Evolution of Wireless Communication

The IEEE 802.11p standard, published in 2010, supports a maximum data rate of 27 *Mbps* [30]. Recently, IEEE 802.11bd was proposed as an evolution of IEEE 802.11p, with more efficient physical layer implementation, providing a potential maximum rate of 87.75 *Mbps* [31]. Such data rates for IEEE 802.11p/bd result in packet (CAM/DENM) rates that exceed 10000 packets/sec and 30000 packets/sec respectively, assuming an average packet length of 300 *bytes*; although the actual packet rates could be lower depending on the network load. This shows the huge room for an adversary to flood its neighborhood with bogus messages. Continuously developing communication standards, e.g., the current 5G and upcoming 6G, that could be used for V2V communication [33], together with relatively constant cryptographic delays (for sustainable security, as discussed earlier) would aggravate clogging DoS attacks.

5 POTENTIAL REMEDIES AND RESIDUAL CHALLENGES AHEAD

Our position is that the combination of signature verification and symmetric key chain *pre-validation* can minimize the impact by DoS attacks; with the key chain (anchor) verification still a challenge ahead. Our proposed solution aims at minimizing the clogging DoS attack impact without attempting a guaranteed timely validation of all messages.

5.1 Key Chain Pre-validation & Signature Verification

We explained in Sec. 4 that lack of non-repudiation implies potential symmetric key chain abuse. However, symmetric key chain based CAM pre-validation can be effective against high-rate bogus message floods; inspired by the key chain-based DoS defense approach for sensor networks [4]. Given that the key chain (its anchor) has been verified based on an earlier signature verification, the follow-up key chain elements can be efficiently validated. As shown in Figure 4, a vehicle would disseminate one CAM carrying the correct key chain element, K_i, in each CAM dissemination interval. This K_i is not used to verify HMACs, but rather pre-validate the CAM before its signature verification. The first CAM carrying the correct K_i at the specific CAM interval is either from the legitimate sender or from the attacker that forged the CAM with the overheard K_i, in case the legitimate one were not received (e.g, lost or sender out of communication range). The legitimate CAM that initially disclosed K_i is impossible to arrive after forged ones due to single-hop broadcast communication (signal propagation at speed of light and no re-transmission of CAMs). Any late CAMs carrying the disclosed K_i are dropped.

To preserve non-repudiation, any CAM that passed the efficient pre-validation is verified based on mandatory signature verification. Even if an attacker floods with high-rate bogus CAMs, this approach guarantees at most one signature verification at each receiver, per CAM interval per legitimate sender. However, this approach assumes an already verified and trusted key chain from each legitimate sender vehicle, thus cannot be used to defend against bogus CAMs carrying bogus PCs. Carefully crafted bogus beacons would look similar to any legitimate CAMs carrying newly received

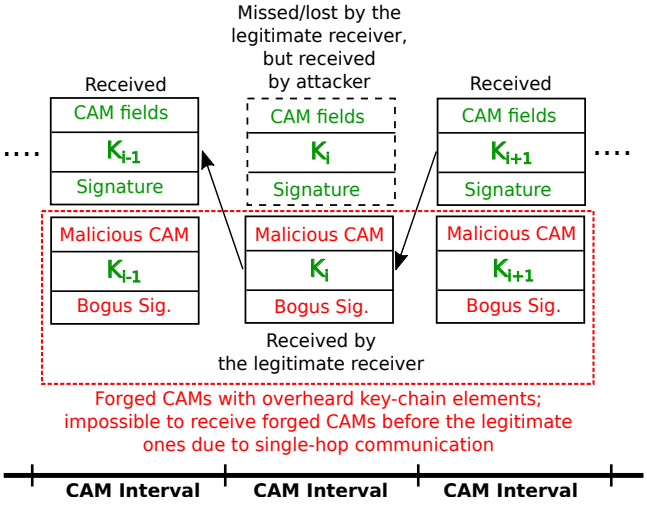

Figure 4: Key chain pre-validation before signature verification.

PCs. Signature verifications of such CAMs implies discovering legitimate CAMs among a flood of bogus ones; necessary for follow-up key chain pre-validation. We discuss potential solution for initial key chain verification at the end of this section.

5.2 Decentralized Environmental Notification Messages

The key chain based approach, explained above, leverages the periodicity of CAMs but cannot be readily used for DENMs. An adaptation is to piggyback DENMs to their immediate next CAMs at the expense of larger packets and slightly increased delay. Alternatively, the key chain interval can be shortened to disseminate DENMs independently. For example, instead of the default CAM interval, 100*ms*, the key chain interval can be shortened to 25 (or 50) *ms* for maximum three (or one) DENMs between two CAMs. The shorter the interval is, the more DENMs can be disseminated (in urgent situations) at the expense of slightly higher hash computation overhead for key chain pre-validation. However, delaying DENM dissemination until the next interval could be an issue for highly critical DENM (that require immediate reaction). Moreover, the effectiveness of key chain pre-validation among loosely synchronized OBUs requires further investigation, especially with shorter intervals to minimize the DENM latency.

5.3 Key Chain Anchor Verification & Neighboring Vehicle Discovery

To make the above key chain pre-validation effective, at least one key from any key chain has to be verified based on a signature verification; to obtain the pre-validation basis. This faces the same challenge as the discovery of neighboring vehicles. Even though benign vehicles generate CAMs and DENMs with the key chain pre-validation approach, attackers can flood their neighborhood with bogus messages making it still hard to find and verify at least

one benign message from each benign sender, needed to make use of the key chain pre-validation.

A potential solution is to leverage V2I communication to obtain pairs of a benign PC and its latest key chain element for neighboring vehicles (to the Roadside Unit (RSU) and thus, potentially to each other). Vehicles can register their PCs and the corresponding key chain elements to a server in the VC system. Local/regional RSUs, synchronized with the server, receive the broadcast V2V messages and verify if any latest valid key chain element exists based on the registered key chain elements at the server, and distribute those pairs of PC and the latest key chain element to the vehicles in range. This has to be carried out via a secure channel different than the broadcast V2X channel(s) (e.g., over a cellular network). Otherwise, direct distribution by RSUs through broadcast (similar to V2V) communication would still be vulnerable to clogging DoS attacks: signature verification is needed there too. V2I communication overhead, storage and computation overhead by the server and RSUs, and the impact on vehicle privacy need to be further investigated for such an approach.

6 CONCLUSIONS

Clogging DoS attacks target timely validation of received V2V messages, a key to secure VC and safe transportation. Despite the common belief that better provisioned/powered OBUs are the answer, we establish that this is unrealistic and that DoS-resilience-by-design is necessary. Efficient solutions could mitigate clogging DoS attacks to some extent, but the evolving wireless communication landscape (with higher packet rates) could aggravate the situation. We position that existing solutions do not fundamentally solve the problem and discuss necessary remedies addressing important aspects including accountability, neighboring vehicle discovery, and DENMs.

ACKNOWLEDGMENTS

The work was supported in parts by the Swedish Science Foundation (VR), the Knut & Alice Wallenberg Foundation, and in parts for Z. Zhou by the China Scholarship Council.

REFERENCES

[1] IEEE Std 1609.2. 2016. IEEE Standard for Wireless Access in Vehicular Environments–Security Services for Applications and Management Messages. *IEEE Std 1609.2-2016* (2016).
[2] Hamad Alrimeih and Daler Rakhmatov. 2014. Fast and flexible hardware support for ECC over multiple standard prime fields. *IEEE Transactions on VLSI Systems* 22, 12 (2014), 2661–2674.
[3] Giorgio Calandriello, Panos Papadimitratos, Jean-Pierre Hubaux, and Antonio Lioy. 2011. On the performance of secure vehicular communication systems. *IEEE Transactions on Dependable and Secure Computing* 8, 6 (2011), 898–912.
[4] Qi Dong, Donggang Liu, and Peng Ning. 2013. Providing DoS resistance for signature-based broadcast authentication in sensor networks. *ACM Transactions on Embedded Computing Systems* 12, 3 (2013), 73.
[5] Siddharth Dongre and Hanif Rahbari. 2021. Message sieving to mitigate smart gridlock attacks in V2V. In *ACM WiSec*.
[6] ETSI TR 102 941. 2021. Intelligent Transport Systems (ITS); Security; Trust and Privacy Management.
[7] ETSI TS 103 096-2. 2022. Intelligent Transport Systems (ITS); Testing; Conformance test specifications for ITS Security; Part 2: Test Suite Structure and Test Purposes.
[8] Hadi Givehchian, Nishant Bhaskar, Alexander Redding, Han Zhao, Aaron Schulman, and Dinesh Bharadia. 2024. Practical Obfuscation of BLE Physical-Layer Fingerprints on Mobile Devices. In *IEEE Symposium on Security and Privacy*. San Francisco, CA.
[9] Sohan Gyawali, Shengjie Xu, Yi Qian, and Rose Qingyang Hu. 2020. Challenges and solutions for cellular based V2X communications. *IEEE Communications Surveys & Tutorials* 23, 1 (2020), 222–255.
[10] Darrel Hankerson, Alfred J Menezes, and Scott Vanstone. 2006. *Guide to elliptic curve cryptography*. Springer Science & Business Media.
[11] Hsu-Chun Hsiao, Ahren Studer, Chen Chen, Adrian Perrig, Fan Bai, Bhargav Bellur, and Aravind Iyer. 2011. Flooding-resilient broadcast authentication for vanets. In *ACM MobiCom*. Las Vegas, NV.
[12] Hongyu Jin and Panos Papadimitratos. 2019. DoS-resilient cooperative beacon verification for vehicular communication systems. *Ad Hoc Networks* 90 (2019), 101775.
[13] Frank Kargl, Panos Papadimitratos, Levente Buttyan, M Muter, Elmar Schoch, Bjoern Wiedersheim, Ta-Vinh Thong, Giorgio Calandriello, Albert Held, Antonio Kung, and Jean-Pierre Hubaux. 2008. Secure Vehicular Communication Systems: Implementation, Performance, and Research Challenges. *IEEE Communications Magazine* 46, 11 (2008), 110–118.
[14] Mohammad Khodaei, Hongyu Jin, and Panos Papadimitratos. 2018. SECMACE: Scalable and Robust Identity and Credential Management Infrastructure in Vehicular Communication Systems. *IEEE Transactions on Intelligent Transportation Systems* 19, 5 (May 2018), 1430–1444.
[15] Mohammad Khodaei, Hamid Noroozi, and Panos Papadimitratos. 2023. SECMACE+: Upscaling Pseudonymous Authentication for Large Mobile Systems. *IEEE Transactions on Cloud Computing* 11, 3 (2023), 3009–3026.
[16] Mohammad Khodaei and Panos Papadimitratos. 2015. The Key to Intelligent Transportation: Identity and Credential Management in Vehicular Communication Systems. *IEEE Vehicular Technology Magazine* 10, 4 (2015).
[17] Arjen K Lenstra and Eric R Verheul. 2001. Selecting cryptographic key sizes. *Journal of cryptology* 14 (2001), 255–293.
[18] Xiaodong Lin and Xu Li. 2013. Achieving efficient cooperative message authentication in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology* 62, 7 (2013).
[19] P. Liu, B. Liu, Y. Sun, B. Zhao, and I. You. 2018. Mitigating DoS Attacks Against Pseudonymous Authentication Through Puzzle-Based Co-Authentication in 5G-VANET. *IEEE Access* 6 (2018).
[20] Chen Lyu, Dawu Gu, Yunze Zeng, and Prasant Mohapatra. 2016. PBA: Prediction-Based Authentication for Vehicle-to-Vehicle Communications. *IEEE Transactions on Dependable and Secure Computing* 13, 1 (2016), 71–83.
[21] Mbed TLS. 2022. https://mbed-tls.readthedocs.io/en/latest/kb/cryptography/elliptic-curve-performance-nist-vs-brainpool/
[22] Mohamad Ali Mehrabi and Alireza Jolfaei. 2022. Efficient Cryptographic Hardware for Safety Message Verification in Internet of Connected Vehicles. *ACM Transactions on Internet Technology* 22, 4 (2022), 1–16.
[23] Panos Papadimitratos. 2024. *Secure Vehicular Communication Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1–6.
[24] Panagiotis Papadimitratos, Levente Buttyan, Tamas Holczer, Elmar Schoch, Julien Freudiger, Maxim Raya, Zhendong Ma, Frank Kargl, Antonio Kung, and J-P Hubaux. 2008. Secure Vehicular Communication Systems: Design and Architecture. *IEEE Communications Magazine* 46, 11 (2008), 100–109.
[25] Jan-Felix Posielek, Norbert Bißmeyer, and Annika Strobel. 2017. A security migration concept for vehicle-to-X communication to allow long-term PKI operation. In *Nets4Cars/Nets4Trains/Nets4Aircraft*. Toulouse, France.
[26] Nikodin Ristanovic, Panos Papadimitratos, George Theodorakopoulos, Jean-Pierre Hubaux, and Jean-Yves Le Boudec. 2011. Adaptive message authentication for multi-hop networks. In *IEEE/IFIP WONS*. Bardonecchia, Italy.
[27] Torsten Schütze. 2011. Automotive security: Cryptography for car2x communication. In *Embedded World Conference*, Vol. 3. 4–24.
[28] Ahren Studer, Fan Bai, Bhargav Bellur, and Adrian Perrig. 2009. Flexible, extensible, and efficient VANET authentication. *Journal of Communications and Networks* 11, 6 (2009), 574–588.
[29] Cong Sun, Jiao Liu, Xinpeng Xu, and Jianfeng Ma. 2017. A privacy-preserving mutual authentication resisting dos attacks in vanets. *IEEE Access* 5 (2017), 24012–24022.
[30] Fernando A Teixeira, Vinicius F e Silva, Jesse L Leoni, Daniel F Macedo, and José MS Nogueira. 2014. Vehicular networks using the IEEE 802.11 p standard: An experimental analysis. *Vehicular Communications* 1, 2 (2014).
[31] Andy Triwinarko, Iyad Dayoub, and Soumaya Cherkaoui. 2021. PHY layer enhancements for next generation V2X communication. *Vehicular Communications* 32 (2021), 100385.
[32] Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. 2008. recaptcha: Human-based character recognition via web security measures. *Science* 321, 5895 (2008), 1465–1468.
[33] Cheng-Xiang Wang, Jie Huang, Haiming Wang, Xiqi Gao, Xiaohu You, and Yang Hao. 2020. 6G wireless channel measurements and models: Trends and challenges. *IEEE Vehicular Technology Magazine* 15, 4 (2020), 22–32.
[34] Wenyuan Xu, Wade Trappe, Yanyong Zhang, and Timothy Wood. 2005. The feasibility of launching and detecting jamming attacks in wireless networks. In *ACM MohiHoc*. Urbana-Champaign, Illinois.

[35] Chenxi Zhang, Xiaodong Lin, Rongxing Lu, Pin-Han Ho, and Xuemin Shen. 2008. An efficient message authentication scheme for vehicular communications. *IEEE Transactions on Vehicular Technology* 57, 6 (2008), 3357–3368.