Targeted Discreditation Attack against Trust Management in Connected Vehicles

Geoff Twardokus, Jaime Ponicki, Samantha Baker, Peter Carenzo, Hanif Rahbari, and Sumita Mishra
Department of Computing Security, Rochester Institute of Technology, Rochester, NY
{gdt5762,jxp3224,sxb2134,pdc5230,hanif.rahbari,sumita.mishra}@rit.edu

***Abstract*—Vehicle-to-vehicle (V2V) communication systems in the U.S. rely on IEEE 1609.2 security protocols for message authentication using digital signatures. A key requirement for trust management in such systems is the ability to detect misbehaving vehicles, e.g., when vehicles are repeatedly forging signatures. However, this creates a new attack surface where receivers cannot determine whether the causes of signature verification failures are indeed malicious attacks. In this paper, we present our novel, open-source, USRP-based testbed and utilize it to demonstrate how a stealthy reactive jammer can exploit this vulnerability. Our novel, targeted attack is highly efficient (even given the short validity period for vehicle pseudonyms) and difficult to detect. Our experimental results show that our attack can successfully discredit a victim in prominent misbehavior detection schemes with just two minutes of jamming. Finally, we discuss the capabilities and extensibility of our testbed as well as the challenges of potential attack mitigation techniques.**

***Index Terms*—Connected vehicle security, trust management, IEEE 1609.2, reactive jamming, USRP testbed**

I. Introduction

Vehicle-to-Vehicle (V2V) communication, in which vehicles talk directly with one another to coordinate their movements and prevent collisions, is projected to be an integral component of smart, connected transportation infrastructure in the near future. Widespread use of V2V promises prevention of up to $600,000$ collisions, $270,000$ injuries, and thousands of deaths annually in the U.S. [1]. V2V communications do not require a line of sight, making V2V complementary to sensor technologies such as LiDAR or cameras. However, V2V technology cannot be considered safe or reliable enough to unleash its benefits on roadways until significant safety concerns have been addressed. For example, drivers will need to react swiftly and decisively if an imminent collision is projected based on information (or warnings) received from another vehicle. It is therefore critical to authenticate incoming messages; otherwise, a decisive action (e.g., swerving) might be taken based on a forged message, leading to a collision, lane departure, or another unsafe outcome.

Security service requirements for V2V have been standardized in IEEE 1609.2 [2]. However, the lack of accessible and affordable testbeds for evaluating 1609.2 protocols in realistic wireless environments (not to mention on actual roads) remains an obstacle to rigorous security assessment and large-scale V2V deployment. This has limited many automakers' willingness to move forward with V2V, exemplified by Toyota's 2019 decision to halt plans for deploying V2V equipment on all U.S. models by 2021 [3]. In fact, even *commercial* evaluation kits (e.g., Cohda MK6c [4]) have only recently begun to implement V2V security protocols, and no open-source implementations are known to be available.

Among the insufficiently validated security requirements defined in the 1609.2 standard, message authentication requires that all messages must be digitally signed with keys supported by public-key certificates. Unfortunately, this mandate creates a significant challenge: deploying a vehicular public-key infrastructure (VPKI) for issuing, managing, and *revoking* vehicles' signing certificates as needed. In turn, each VPKI needs to include a "misbehavior detection system" – required, but not specified, by the current U.S. VPKI standard [5] – to identify vehicles that are not complying with security (or other) requirements and revoke their certificates. Many proposals for misbehavior detection use reputation-based schemes (e.g., [6]) where a vehicle's trustworthiness erodes as its messages are reported for problems such as *unverifiable signatures*. Such a vehicle may eventually be expelled from the VPKI (via certificate revocation) because of its low reputation. However, identifying a vehicle as misbehaving in this naive way is problematic because a valid signature may fail verification as a result of either environmental or malicious causes. In fact, under 1609.2 standards a receiver can only determine *whether* a given signature is not valid, not *why* that signature is invalid.

In this paper, we show that an intelligent attacker can induce repeated signature failures through strategic, stealthy, reactive jamming – a *targeted discreditation attack* – so that a legitimate vehicle whose messages are jammed may be improperly labeled as misbehaving by the VPKI. This, in turn, will result in that vehicle's certificate being improperly revoked, and its future messages being ignored by all other vehicles, potentially causing catastrophic consequences as exemplified in Fig. 1. We contend that this vulnerability has not previously been exposed because 1609.2 has not been sufficiently validated in realistic wireless environments.

We expose this vulnerability through our efforts to address the V2V testing and validation gap by developing a new software-defined radio (SDR) testbed, *V2Verifier*, for V2V security. *V2Verifier* features an entirely open-source imple-

This research was supported by the National Security Agency under Grant Number H98230-19-1-0318. The United States Government is authorized to reproduce and distribute reprints notwithstanding any copyright notation herein. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Security Agency.

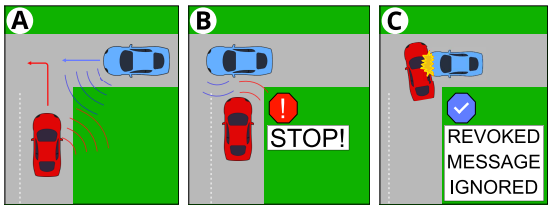

Fig. 1. An example scenario showing (A) two vehicles approaching an intersection, (B) proper behavior when a stop warning is accepted, and (C) a possible collision if the vehicle with the right of way (the blue car) is wrongly excluded from the VPKI, making the other (red) vehicle ignores its warnings.

mentation[1] of the Wireless Access in Vehicular Environments (WAVE) protocol stack [7] together with the *IEEE 1609.2 security protocol*. Our open-source testbed is the first of its kind and we introduce it here as an integral part of developing and demonstrating our targeted discreditation attack. This attack has an extremely low duty cycle ($< 1\%$), is extremely difficult to distinguish from interference or noise, and can succeed by jamming as few as 20% of messages from its target in any 5-minute period. In addition, because the vulnerability we exploit is integral to the core authentication mechanism of 1609.2, our attack applies to virtually all current forms of misbehavior detection schemes that blindly rely solely on reports of signature failure by one or multiple vehicles.

Specifically, we make the following contributions:

- We introduce what is, to the best of our knowledge, the first open-source, SDR-based wireless testbed for V2V security based on the IEEE 1609.2 standard.
- We expose, analyze, and experimentally demonstrate exploitation of a critical vulnerability in existing, low-false-positive misbehavior detection schemes for V2V systems protected by 1609.2. We accomplish this by developing a reactive, low-duty-cycle and hard-to-detect jamming attack that, despite error correction codes, causes a legitimate vehicle to be classified as misbehaving.

The rest of this paper is organized as follows. In Section II, we provide an overview of V2V and its security as well as the requirements and existing schemes for misbehavior detection in V2V. We introduce our *V2Verifier* testbed in Section III. Our novel reactive-jamming discreditation attack and its experimental validation are presented in Sections IV and V, respectively. We conclude with an overview of related work in Section VI and final remarks in Section VII.

II. Preliminaries

All V2V systems use one of two competing technologies: Dedicated Short Range Communications (DSRC), based on IEEE 802.11p [7] protocol, or Cellular Vehicle-to-Everything (C-V2X) [8], based on 4G/LTE (or 5G) cellular communication. In both cases, conventional protocols (802.11 or LTE) are adapted to the unique V2V requirements for direct communication between vehicles with low latency and high reliability. Without loss of generality, we focus on DSRC/WAVE, the technology stack currently implemented in *V2Verifier*, while noting that *V2Verifier* can easily replace DSRC with C-V2X at the physical (PHY) and MAC layer. DSRC uses 802.11p protocol to communicate in the $5.9\,\text{GHz}$ band over a $10\,\text{MHz}$ channel. IEEE 1609.3 [9] then defines WAVE Short Message Protocol (WSMP) as a low-latency network and transport layer protocol on top of 802.11p (or C-V2X). Finally, security services and protocols are defined in the IEEE 1609.2 standard.

Basic Safety Messages—The core component of V2V safety applications in the U.S. is the Basic Safety Message (BSM). A V2V-equipped vehicle broadcasts a BSM at least once every $100\,ms$ to inform nearby vehicles about its GPS coordinates, speed, direction of travel, and more. This information allows nearby vehicles to avoid colliding with the sender, even if the sender's movements cannot be seen by other drivers or detected by vehicle sensors.

A. Message Authentication in IEEE 1609.2

The IEEE 1609.2-2016 standard [2] and its amendments, together with the recent 1609.2.1-2020 standard [10], define services and protocols for securely exchanging BSMs in both DSRC and C-V2X. Under 1609.2, every BSM is digitally signed using the Elliptic Curve Digital Signature Algorithm (ECDSA) and a certificate issued by a VPKI. ECDSA provides both authentication and integrity protection for the message, allowing receivers to verify if the message or its signature has been tampered with in transit (among other things).

However, in case of a verification failure, ECDSA is not able to determine whether it is the result of malicious activity at the upper layers or of communication errors at the lower layers. Communication errors are supposed to be detected (and possibly corrected) at the PHY layer, but the CRC32 error detection mechanism in DSRC is notoriously unreliable at detecting message modifications that cannot be corrected by error correction codes. As we will show, this implicit reliance on the PHY layer to detect and drop corrupted packets is a serious vulnerability in the 1609.2 security protocols.

Another requirement of 1609.2 is the use of privacy-preserving "pseudonym certificates" that change periodically in place of a persistent certificate. The pseudonym certificate within each BSM then contains a plaintext pseudonym that cannot be linked to the vehicle's permanent identity by any tracker or eavesdropper. While 1609.2 does not define any validity period for pseudonyms, 5-minute is a common reference point [5]. So, we assume a fresh pseudonym will be deployed every five minutes and the old ones will never be reused.

B. Misbehavior Detection in V2V

The Security Credential Management System (SCMS) is the U.S. Department of Transportation's proposed VPKI framework that outlines (among other things) the architecture for distributing, updating, and revoking certificates of vehicles [5]. To facilitate certificate revocation, the SCMS proposal includes a requirement that misbehaving units be detected and reported to a central misbehavior authority. However, the proposal does not specify a method that should be used for detecting misbehavior, an obvious precondition to reporting misbehavior.

[1]Available at https://github.com/twardokus/v2verifier

In the absence of a standard misbehavior detection scheme, a variety of methods have been proposed in the literature to detect misbehaving vehicles. These include machine learning [11], out-of-band [12], and reputation-based [6] schemes. In the reputation-based scheme [6], each vehicle maintains a local reputation database for all of the vehicles it encounters. Every time a message is received, the sender's reputation is updated based on the properties of the message. As one such message property could be the outcome of signature verification, it follows that under this or similar schemes, the receiver would decrement the reputation of a vehicle each time an unverifiable message is received from that vehicle. Therefore, if a similar misbehavior detection scheme is used in a VPKI, a vehicle can be "discredited" – wrongly made to look like a misbehaving vehicle – by an intelligent attacker who causes repeated verification failures.

III. V2Verifier – A Testbed for V2V Security

As an integral part of exposing the vulnerability exploited by our attack, we present *V2Verifier*, an open-source SDR testbed featuring the first open-source implementation of the IEEE 1609.2 security services. *V2Verifier* uses Universal Software Radio Peripherals (USRPs) to emulate vehicles exchanging Secure Protocol Data Units (SPDUs), which are BSMs secured using 1609.2 protocols. This configuration provides a flexible and affordable environment for V2V security testing and experimentation. We present *V2Verifier* here for the first time, along with our attack, to showcase its capacity to identify V2V security threats that are not apparent from theoretical or closed-source code analysis.

1) Architecture: *V2Verifier* has a modular design with the upper layer protocols and services (e.g., WSMP, 1609.2) implemented independently from the lower layers. This allows the current PHY/MAC layers (DSRC) to be easily replaced with either an upgrade of DSRC (e.g., 802.11bd [13]) or an alternative protocol like C-V2X. Both WSMP and 1609.2 are highly flexible protocols with few mandatory (and many optional) features, so we have implemented the mandatory features of each and left it for future users to add additional features as needed. We intend to add more options for both protocols as a part of our future work.

2) Operation: A series of ordered GPS coordinates (i.e., vehicle paths), provided by an external traffic simulator, are used to calculate the senders' speed and heading (from positional and angular change over time) and BSMs are generated containing this information. Each BSM is then signed using ECDSA and, together with the resulting signature and the vehicle's signing certificate, packed into an SPDU structure defined by 1609.2. The SPDU forms the payload of a WAVE Short Message (WSM), which is encapsulated in an 802.11p frame and transmitted using a USRP. The receiver verifies the BSM's signature and then updates its local tracking system with the motion data received from the other "vehicle." This updated data is visually represented on a graphical interface, explained next.

Fig. 2. A snapshot of the receiver's (green vehicle's) perspective in *V2Verifier*. Each nearby (red) vehicle's location is updated with each message from that vehicle, and the validity of the most recent message is indicated by translucence of vehicles whose last message did not contain a valid signature.

3) Graphical Interface: To represent the real-world effects of BSMs being transmitted by multiple vehicles in roadway environments, we further developed a simple interface to visualize the impact of different attacks on V2V. From the receiver's perspective, it displays all nearby vehicles on a map grid. Fig. 2 shows an example with four vehicles positioned per their last BSM (irrespective of verification outcome), including one vehicle adjacent to the receiver shown transparently to indicate verification failure of its most recent message.

IV. Targeted Discreditation Attack

While experimenting with sending and receiving BSMs using *V2Verifier*, we observed an unexpectedly high rate of signature verification failures due to communication errors between USRPs. Following this observation, we determined that 1609.2-based message authentication cannot reliably distinguish between malicious and environmental causes of signature verification failures. We then conceived an attack to exploit this vulnerability, which potentially exists in a number of DSRC and C-V2X systems that use 1609.2 for security. We model an active attacker who seeks to induce targeted decoding errors in messages sent by a specific, targeted vehicle, causing those messages to be unverifiable by one or more receivers. Using a reputation-based misbehavior detection scheme similar to [6], we show how an attacker can cause a vehicle to be discredited and expelled from a VPKI.

A. Threat Model

We consider the scenario with three actors depicted in Fig. 3. Alice, the target, is a legitimate member vehicle in a VPKI making use of 1609.2 message authentication and pseudonyms to securely send her BSMs. Bob is another legitimate member vehicle travelling within communication range of Alice. Note that there may be other vehicles around Alice but, without loss of generality, we consider one representative vehicle (Bob) for our model. A jammer, Jane, is positioned within the communication ranges of both Alice and Bob.

We assume that Jane is capable of sniffing and transmitting on the 5.9 GHz band (e.g., using a USRP) and visually identifying Alice to begin acquiring her current pseudonym. However, we do not assume Jane has *a priori* knowledge of

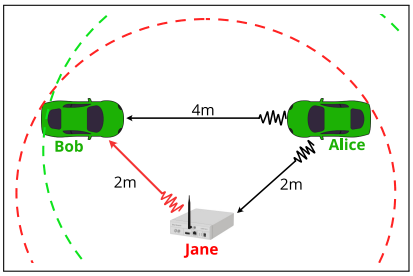

Fig. 3. Threat model for targeted discreditation attack. (The distance values and communication ranges shown above are according to our experimental setup only, and are not necessarily part of the model.)

Alice's pseudonym. If Alice is mobile, Jane may require a vehicle to follow Alice; however, this requirement is not strict. For example, if Alice is in slow-moving traffic and remains in range of Jane's transmitter for a long period of time, then Jane does not need to be mobile. We further assume that Alice starts with a perfect reputation which resets to the highest level whenever she changes her pseudonym, making it more difficult for Jane to discredit her.

B. Reactive Jamming Discreditation Attack

As noted in Section II, each 1609.2 pseudonym has a 5-minute validity period. Consequently, Jane has at most five minutes to complete her attack before Alice changes her pseudonym and erases any reputation degradation caused by Jane, forcing Jane to start her attack all over again. In total, Jane's attack has three phases:

1) Target Pseudonym Acquisition: The first attack phase aims at acquiring Alice's current pseudonym. Outside of dense environments, Jane may simply use received signal strength (RSS) to isolate Alice's transmissions from others. This is feasible in scenarios where brief intervals place Alice much closer to Jane than other vehicles. For example, Jane may follow Alice on a secondary highway where no other vehicles are nearby. In dense environments, more complex techniques such as direction-finding with a multi-antenna receiver [14] can be used. Regardless of the specific method used, Jane has several options to isolate Alice's signal and extract her current pseudonym from one of her messages.

2) Target Transmission Identification: For a brief window of five minutes from acquisition of Alice's current pseudonym, or until no more messages are received with Alice's known pseudonym (indicating pseudonym rotation has occurred), Jane continuously listens for incoming messages. While listening, she *partially* processes every incoming message until she identifies a message that contains Alice's pseudonym. To quickly determine whether Alice sent a given message, Jane preemptively decodes it on a per-symbol basis (using a modified Viterbi decoder) instead of waiting for the completion of the entire frame. Exploiting the known, rigid structure of 1609.2 SPDUs as well as the length and rate fields in the headers, Jane can quickly pinpoint the pseudonym field (see Fig. 4) and avoid decoding the rest of the frame. She can then compare the decoded pseudonym to Alice's pseudonym. If a match is identified, she can move to the next phase, where she wants to actively jam a specific portion of Alice's messages (the signature in Fig. 4). Otherwise, this process repeats until a message from Alice can be identified.

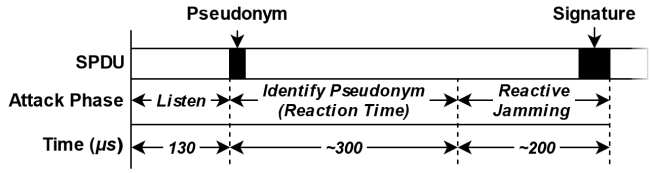

Fig. 4. Time intervals for each phase of the discreditation attack during a frame transmission, showing a minimum $\backsim 500\,\mu s$ reaction window for Jane.

3) Reactive Jamming Execution: As soon as a pseudonym match confirms that a message is from Alice, Jane transmits a short-lived ($< 500\,\mu s$) jamming signal within the current $100\,ms$ BSM interval. Because the pseudonym is placed before the midpoint of SPDUs (see Fig. 4), Jane has a relatively large window left before the signature field to react. In order to cause signature verification to fail at Bob, only a very small number of symbols need to be corrupted – just enough to defeat his Viterbi decoder – so a very brief jamming interval will be sufficient. When Bob cannot verify the signature, he will degrade his opinion of Alice's reputation. After several iterations, Bob's opinion of Alice will drop below a defined threshold and her certificate may be wrongly revoked.

C. Aggressive Jamming Strategy

So far, we have discussed a jammer designed to react solely on a per-message basis; however, Jane can actually attack multiple messages proactively by performing pseudonym detection (phase 2 above) only once. DSRC standards dictate a fixed transmission interval for BSMs, so once the arrival time of one message from Alice is recorded, her future messages' arrival times can be predicted with some degree of accuracy (in spite of short delays due to CSMA/CA) by adding intervals of $100\,ms$ (see Section II). Jane would then jam all messages that arrive at or around the expected times of Alice's future messages. This approach has the potential to more rapidly degrade Alice's reputation, as the increase in Jane's accuracy would proportionally reduce the time required to complete the attack. However, we expect that Jane would then unintentionally jam messages from vehicles other than Alice, possibly making Jane more detectable. We leave the optimization of this strategy for future work.

D. Detection and Mitigation Challenges

There are two broad approaches to preventing our attack: detecting and removing the attacker, or mitigating the underlying vulnerability. The first option requires distinguishing extremely brief jamming signals from random interference or channel noise. Because Jane transmits for less than $500\,\mu s$ on each attempt and she may not need to jam every one of Alice's messages, Bob's only indication of an anomaly is limited to his failure to verify Alice's signatures. This can be attributed to different causes (such as noise), making it hard to discern

Jane's actions. We note that a consensus-based misbehavior detection scheme would not be able to detect Jane either, as her jamming simultaneously impacts multiple receivers.

Mitigating the underlying vulnerability is also difficult. The fundamental problem, that ECDSA in 1609.2 is blind to the causes of signature failures, makes our attack effective against all forms of misbehavior detection schemes that rely solely on the incomplete knowledge about 1609.2 signature failures.

One misleadingly simple remedy would be to let the existing CRC32 mechanism in DSRC detect communication errors at the PHY layer, preventing the corrupted message from ever reaching the upper-layer 1609.2 protocols. However, there are some limitations to this approach. First, it assumes a perpetual ability of PHY-layer techniques to reliably detect such errors. Against a sophisticated attacker, this assumption may not hold up. CRC32 is well known to suffer from issues such as a lack of resistance to hash collisions. Aside from a jammer who may exploit a collision by chance, a sophisticated attacker may exploit such vulnerabilities and bypass the CRC32 error detection by crafting an intelligent jamming signal. Thus, we consider reliance on ill-suited mechanisms like CRC32 to be a poor long-term design choice for securing V2V systems. Second, V2V implementations are not guaranteed to use robust PHY-layer error detection mechanisms because compliance with 1609.2 does not require such considerations. So, we argue that 1609.2 should not treat those layers as sufficient to maintain its own security guarantees. Instead, 1609.2 would benefit from an alternative mechanism that can distinguish the error types, independently or jointly with the PHY/MAC layers, eliminating the root vulnerability.

V. Experimental Validation

In this section, we evaluate our attack through over-the-air experimentation with three USRPs representing Alice, Bob, and Jane (see Fig. 3). To implement a highly reactive DSRC/WAVE jammer, we heavily modified the transmitter and receiver from an existing open-source 802.11p implementation [15], and further modified the Viterbi decoder to terminate once the pseudonym field is decoded. We then evaluated our attack's performance in a controlled environment with Alice's message rate set to the default 10 BSMs per second.

A. Experimental Setup

We experimentally emulate legitimate BSM exchange using *V2Verifier* with two USRP B210s representing Alice and Bob. Jane is a reactive jammer using a USRP N210 with a UBX40 daughterboard, chosen over the B210 because the UBX40 has two independent transmit and receive chains. This allows Jane to begin jamming on the transmit chain almost instantaneously after the target pseudonym is identified on the receive chain. The switching delay on the USRP is therefore very small, ensuring a low reaction time overall. Alice and Jane transmit with equal power using $5\,dBi$ antennas.

B. Performance Evaluation

1) Success Rate: Defined as the percentage of Alice's packets correctly detected and successfully jammed by Jane, our experimental results show that Jane can achieve an 80% success rate.

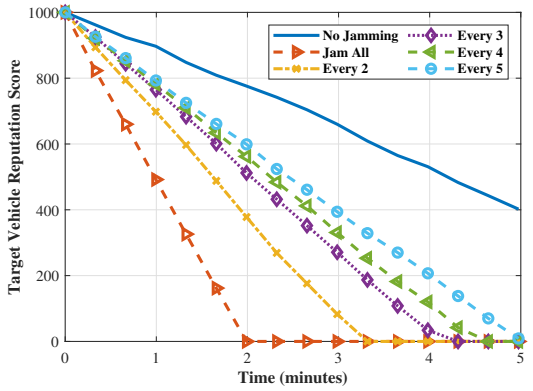

Fig. 5. Vehicle credibility over a 5-minute pseudonym validity period under normal conditions vs. various jamming periods, ranging from attempting to jam all messages through attempting to jam one of every five messages.

2) Reaction Time: To verify the feasibility of this attack, we needed to verify whether Jane's reaction time is short enough. It is difficult to estimate Jane's reaction time because her jamming signal will arrive at Bob in the middle of Alice's transmission. Instead, we configured Jane to react on a different channel than the one Alice and Bob were communicating on. A fourth USRP was set up to listen on this second channel and record the jamming signal's arrival time. By placing this monitoring USRP directly adjacent to Bob and comparing the arrival time of Jane's signal with that of Alice's signal, we can deduce Jane's reaction time. We determined the average reaction time to be close to $800\,\mu s$. However, this average is affected by several outliers, comprising roughly 20% of jamming attempts, where Jane took longer than $5\,ms$ (more than 6 times the average) to react to Alice's message. Although we cannot show a direct correlation, this 20% corresponds to the 20% failure rate noted above, strongly suggesting that when Jane fails to jam a particular message from Alice it is due to an abnormally delayed reaction. By removing these outliers, we determine that the true average reaction time is closer to $300\,\mu s$, well within the minimum $500\,\mu s$ reaction window (shown in Fig. 4).

3) Misbehavior Detection Performance: We implemented a basic misbehavior detection scheme on *V2Verifier* where each vehicle is given a reputation score from $0-1000$, with 1000 being a perfectly compliant vehicle and 0 being a vehicle identified as misbehaving and pending certificate revocation. Each time Bob cannot verify the signature on a message sent by Alice, he decrements his opinion of Alice's reputation by one point. Naturally, communication errors in the absence of jamming also affect the rate of degradation but we assume, for reasons described in Section IV-B, that the reputation score is reset each time a new pseudonym is issued.

In Fig. 5, we show the rate at which Alice's reputation degrades over time during Jane's attack compared with when there is no jamming. In a jamming-free scenario, our experiments show that communication errors alone may reduce

the reputation of a given vehicle by at most 50% over one 5-minute pseudonym validity period. In contrast, Jane can successfully degrade Alice's reputation to zero in as little as two minutes if she attempts to jam all of Alice's BSMs, or as much as five minutes if she jams only one out of every five (20% of) BSMs.

VI. Related work

In the following, we discuss the limitations of related work.

A. SDR Testbeds for V2V

SDR testbeds are being increasingly used for experimental study and evaluation of V2V protocols. In 2016, Stoica *et al.* [16] demonstrated an open-source 802.11p channel estimator with SDRs. In [17], SDRs were used to detect false position attacks at the PHY layer. In [14], real-time reconfiguration of SDRs was shown to facilitate complex, dynamic experimentation that commercial V2V equipment cannot perform. More recently, a system for modeling advanced V2V use cases was presented in [18] using a simulator front-end with USRPs exchanging over-the-air C-V2X communications.

In contrast to these and other existing works, *V2Verifier* distinctively focuses on *1609.2 security protocols*, making *V2Verifier* the first SDR testbed to facilitate experimentation with V2V security standards. Also, *V2Verifier* goes beyond these works by facilitating a broad range of experimental scenarios with V2V protocols without requiring commercial equipment or software, as well as by supporting integration of any existing (or future) V2V technology (e.g. DSRC, C-V2X, 802.11bd [13]) at the PHY/MAC layer.

B. Attacking V2V Misbehavior Detection Systems

There is little work in the literature that addresses the possibility of abusing a misbehavior detection scheme to improperly expel a targeted vehicle from a VPKI. Considering the twin pillars of our attack, reactive jamming and misbehavior detection, the only work we are aware of that is closely related to ours is [19], where a hybrid jamming attack is presented that aims to cause a DSRC-equipped transmitter to misbehave. By keeping the medium busy with random-length jamming intervals, BSMs that are supposed to be periodic are instead queued until the medium is briefly free of jamming. Then, the transmitter violates the 802.11p requirement to send one (and only one) BSM every $100\,ms$ by sending a burst of (outdated) queued messages. Our attack differs in three important ways: (1) while [19] causes misbehavior at the MAC layer, our PHY-layer attack creates anomalies at the upper layers; (2) the attack in [19] causes vehicles to actually misbehave, whereas ours only causes the inaccurate *perception* that a target vehicle is misbehaving; and (3) [19] is specific to DSRC, while our attack potentially works against any V2V protocol that uses 1609.2 for security.

VII. Conclusion and Future Directions

In this work, we exposed a serious vulnerability of misbehavior detection systems in VPKIs that use 1609.2 security protocols by demonstrating a novel, targeted jamming attack using our new USRP testbed for V2V security. We showed that the high false-positive rate of misbehavior detection based on 1609.2 message authentication is due to its inability to distinguish verification failures caused by misbehaving vehicles from those caused by our attack, calling for a more intelligent approach to misbehavior detection. In future work, we will investigate the effectiveness of our attack against other technologies (e.g., C-V2X) and how commercial equipment with stronger PHY-layer mechanisms may react. We will also investigate how an advanced attacker might enhance our reactive attack through proactive execution or other optimizations.

References

[1] National Highway Traffic Safety Administration, "Technical report 11078-101414-v2a," 2014, Accessed: Feb. 20, 2020. [Online]. Available: https://bit.ly/35EggyG

[2] *Wireless Access in Vehicular Environments–Security Services for Applications and Management Messages*, IEEE Standard 1609.2-2016, 2016.

[3] D. Shepardson, "Toyota abandons plan to install U.S. connected vehicle tech by 2021," Apr. 2019, accessed: Jun. 13, 2020. [Online]. Available: https://reut.rs/3e17r5P

[4] Cohda Wireless, "MK6c EVK - Cohda Wireless," 2020, accessed: Oct. 26, 2020. [Online]. Available: https://bit.ly/2TCgCQt

[5] B. Brecht *et al.*, "A security credential management system for V2X communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 12, pp. 3850—3871, Dec. 2018.

[6] N. Magaia and Z. Sheng, "ReFIoV: A novel reputation framework for information-centric vehicular applications," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1810–1823, Feb. 2019.

[7] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments*, IEEE Standard 802.11p, 2010.

[8] *Summary of Rel-14 Work Items*, 3rd Generation Partnership Project Technical Specification 21.914 V14.0.0, 2018.

[9] *Wireless Access in Vehicular Environments (WAVE)–Networking Services*, IEEE Standard 1609.3-2016, 2016.

[10] *Wireless Access in Vehicular Environments (WAVE)–Certificate Management Interfaces for End Entities*, IEEE Standard 1609.2.1-2020, 2020.

[11] S. Gyawali, Y. Qian, and R. Q. Hu, "Machine learning and reputation based misbehavior detection in vehicular communication networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8871–8885, Aug. 2020.

[12] V. L. Nguyen, P.-C. Lin, and R.-H. Hwang, "Enhancing misbehavior detection in 5G Vehicle-to-Vehicle communications," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 9417–9430, Sep. 2020.

[13] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment: Enhancements for Next Generation V2X*, IEEE Standard P802.11bd.

[14] A. Abdelaziz *et al.*, "Beyond PKI: Enhanced authentication in vehicular networks via MIMO," in *Proc. IEEE Int. Workshop Signal Process. Advances in Wireless Commun. (SPAWC)*, Kalamata, Greece, Jun. 2018.

[15] B. Bloessl, M. Segata, C. Sommer, and F. Dressler, "An IEEE 802.11a/g/p OFDM receiver for GNU Radio," in *Proc. Second Workshop Softw. Radio Implementation Forum*, Hong Kong, China, 2013, pp. 9–16.

[16] R.-A. Stoica, S. Severi, and G. T. F. de Abreu, "On prototyping IEEE 802.11p channel estimators in real-world environments using GNURadio," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Gothenburg, Sweden, Jun. 2016, pp. 10–15.

[17] S. Kuk, H. Kim, and Y. Park, "Detecting false position attack in vehicular communications using angular check," in *Proc. ACM Int. Workshop Smart, Auton., and Connected Veh. Syst. Services (CarSys '17)*, Snowbird, UT, USA, Oct. 2017, pp. 25—-29.

[18] W. Zhang, S. Fu, Z. Cao, Z. Jiang, S. Zhang, and S. Xu, "An SDR-in-the-loop Carla simulator for C-V2X-based autonomous driving," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Toronto, ON, Canada, Jul. 2020, pp. 1270–1271.

[19] S. Hussein, M. S. Mohamed, and A. Krings, "A new hybrid jammer and its impact on DSRC safety application reliability," in *Proc. IEEE Annu. Inf. Technol. Electron. Mobile Commun. Conf. (IEMCON)*, Vancouver, BC, Canada, Oct. 2016.