REPLACE: Real-time Security Assurance in Vehicular Platoons Against V2V Attacks

Srivalli Boddupalli, Ashwini Hegde, and Sandip Ray

Abstract— **Connected Autonomous Vehicular (CAV) platoon refers to a group of vehicles that coordinate their movements and operate as a single unit. The vehicle at the head acts as the leader of the platoon and determines the course of the vehicles following it. The follower vehicles utilize Vehicle-to-Vehicle (V2V) communication and automated driving support systems to automatically maintain a small fixed distance between each other. Reliance on V2V communication exposes platoons to several possible malicious attacks which can compromise the safety, stability, and efficiency of the vehicles. We present a novel distributed resiliency architecture, REPLACE for CAV platoon vehicles to defend against adversaries corrupting V2V communication reporting preceding vehicle position. REPLACE is unique in that it can provide real-time defense against a spectrum of communication attacks. REPLACE provides systematic augmentation of a platoon controller architecture with real-time detection and mitigation functionality using machine learning. Unlike computationally intensive cryptographic solutions REPLACE accounts for the limited computation capabilities provided by automotive platforms as well as the real-time requirements of the application. Furthermore, unlike control-theoretic approaches, the same framework works against the broad spectrum of attacks. We also develop a systematic approach for evaluation of resiliency of CAV applications against V2V attacks. We perform extensive experimental evaluation to demonstrate the efficacy of REPLACE.**

I. INTRODUCTION

We are witnessing significant transformation in the transportation industry with the infusion of autonomy and connectivity in vehicular applications. These features have the potential to significantly improve safety, efficiency, and sustainability by reducing and eventually eliminating human errors [13]. Unfortunately, one consequence is the increasing vulnerability of vehicular applications to cyber-attacks. Recent research has shown that it is feasible, and even depressingly simple, to compromise the electronic components of a vehicle and subvert its driving functionality [6], [12]. A particularly vulnerable component of connected vehicles is vehicular communication. An adversary can subvert such communication through a variety of means including fabrication of false messages, mutating or preventing delivery of legitimate messages, jamming the communication channels [2], [11], [17], etc. Adoption of connected autonomous vehicular applications depends critically on resiliency techniques to protect against such attacks.

In this paper, we consider a quintessential connected vehicle application, *viz.*, connected autonomous vehicular platooning ("platooning" for short). In platooning, vehicles are organized into strings where the goal is for each vehicle to adapt its velocity in accordance with the rest of the platoon while maintaining a fixed headway from its immediate neighbor. Vehicles in a platoon coordinate their movement by exchanging position, velocity, and acceleration values through vehicle-to-vehicle (V2V) messages. Platooning enables long strings of vehicles to safely and efficiently follow each other, thereby reducing fuel costs and travel time. However, attacks on V2V communication can subvert the application, which could lead to catastrophic accidents, inefficient utilization of the roadway, ghost traffic jams, string instability, etc.

Our primary contribution is a real-time resiliency solution for platooning. We develop an architecture, REPLACE (for "Resilient Platoon Controller"), that systematically augments a platoon controller with architectural components for real-time detection and mitigation of V2V attacks using machine learning (ML). This can be installed in any vehicle participating in the platoon and offers protection against V2V compromises. We refer to this vehicle as the *ego vehicle*. The key insight behind REPLACE is that it is possible to enable an ML system to learn (and predict) the normal behavior of the ego vehicle in a stable platoon, and consequently detect an adversarial attack with perceptible impact as an anomaly. We show how to make this idea work for realistic platooning implementations and the trade-offs and design choices involved to account for real-time constraints and the limited computation available in automotive ECUs.

The paper makes several important contributions. First, REPLACE represents, to our knowledge, the first comprehensive real-time resiliency framework for multi-vehicle CAV applications in general and platooning in particular against V2V attacks. Furthermore, the resiliency solution "works" even when multiple vehicles in the platoon are targeted by an adversary simultaneously. Secondly, we present a comprehensive taxonomy of V2V attacks that enables systematic navigation of the attack space. Third, we provide a flexible experimental methodology for evaluating resiliency of CAV applications against V2V attacks.

The remainder of the paper is organized as follows. Section II provides the relevant background. In Section III we discuss the threat model considered in the work and develop an attack taxonomy based on the threat model. In Section IV, we provide an overview of REPLACE, focusing in particular on the various constraints that need to be satisfied and the design choices involved. We present our evaluation results in Section V. We discuss related work in Section VI and

*Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611. USA. Email: bodsrivalli12@ufl.edu, ashwini.hegde@ufl.edu, sandip@ece.ufl.edu.

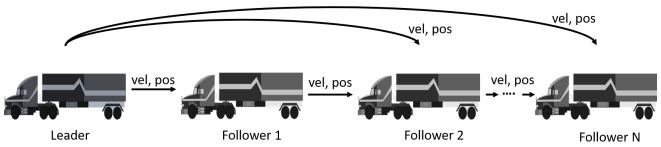

Fig. 1: A Vehicular Platoon System

conclude in Section VII.

II. Background

A. Platooning Basics

A platoon is a string of vehicles operating as a single unit by coordinating their movements through mutual information sharing via V2V communication and on-board sensor systems. Vehicles typically share their instantaneous position, velocity, acceleration or driving directives as V2V messages. The goal of a platoon is to achieve maximum fuel efficiency for all the vehicles while maintaining a safe optimal headway. In the absence of V2V communications, a platoon must maintain larger gaps between vehicles to ensure sufficient response time for the on-board sensors. However, with V2V communication this response time is drastically reduced and the vehicles can safely maintain a smaller headway between them. Connected autonomous platoon applications are designed as either centralized or distributed control systems with various information flow topologies. In a centralized design, driving decisions of each vehicle in the platoon are dictated by the driving directives from the "*leader*, *i.e.*, the first vehicle in the platoon. In a distributed platoon system, each vehicle has its own control system that determines how it adapts its velocity. A platoon controller can utilize information from any subset of vehicles in the platoon in its decision-making.

B. A Platooning Implementation

While the general framework of REPLACE can be applied on top of any platooning implementation, for our evaluation, we consider the implementation of distributed CAV platoon system proposed by Santini *et al.* [16]. In this implementation, each follower vehicle utilizes V2V communication from the leader and its immediate preceding vehicle providing the instantaneous velocity and position. The distributed controller in each vehicle is designed to minimize the relative velocity to zero with respect to the leader and maintain a constant headway close to 0.8s from its immediate preceding vehicle. Figure 1 shows the corresponding information flow topology.

III. Introduction to Platoon Security

CAV platoons are susceptible to adversaries that exploit vulnerabilities in either the vehicles' hardware, software or communication interfaces. Our research is focused on adversaries that compromise V2V communications among the vehicles of the platoon. Since V2V communication provides crucial perception information, these attacks can mislead the victim vehicles into making unsafe or inefficient driving decisions resulting from inaccurate assessment of the platoon behavior. In this section we develop a comprehensive taxonomy of V2V adversaries exploiting platoon systems. While the taxonomy is used specifically for evaluation of REPLACE, the principle is applicable to other CAV applications and more complex threat models.

A. Threat Model

Our threat model accounts for adversaries capable of corrupting the V2V communication originating from any follower (*i.e.*, non-lead) vehicle in the platoon. In other words, one or more vehicles in the platoon may receive corrupted V2V communication from their preceding vehicles. For our platoon implementation, since the preceding position is the primary parameter communicated through V2V, adversarial corruption can only result in wrong, misleading or unavailable value of this parameter. V2V communication from the leader to the rest of the platoon is assumed to be trusted. Furthermore, the on-board sensors, decision computing units and actuarial modules of each vehicle in the platoon are considered to be trusted.[1] REPLACE is agnostic to the origin of the attack: it can be a compromised communication network component or ego vehicle on-board infrastructure, or a rogue preceding vehicle, *e.g.*, denial of message delivery can be caused by exploiting the hardware/software modules of the ego vehicle or through malicious interference with the communication protocol.

B. A Taxonomy Of Adversaries

The V2V attack space is complex and diverse. Popular communication attack mechanisms include denial of service (DoS) through jamming or flooding, masquerading, wormhole, man-in-the-middle (MITM), etc. Since automotive security and CAV applications are both relatively nascent research areas, the attack surface for vehicular communications is continually evolving with new, previously unknown zero-day attacks discovered frequently. It is infeasible for a resiliency architecture to be continually updated in response to newly discovered attacks: a viable resiliency solution should protect the ego vehicle against known attacks but additionally be robust against new attacks discovered in future.

Our approach to address this crucial challenge is to develop a comprehensive taxonomy of adversaries in CAV applications. The key insight is that although the attack *mechanisms* evolve (*e.g.*, through man-in-the-middle, compromised infrastructure component, channel jamming, etc.), any V2V attack can be sufficiently characterized with the help of a few well-defined features. For instance, an adversary can manipulate a V2V message only in the following ways: (1) mutate a message in flight, (2) create a new (synthetic) message; and (3) prevent delivery of messages to the ego vehicle. Using such insights, our taxonomy (Fig. 2)

[1] Sensors and computing units are also vulnerable to cyber-attacks [1], [2], [7], [14]. Nevertheless, since the modes of operation for attacking sensors and V2V communication are distinct from each other, it is reasonable to assume that the sensors remain trusted in the context of a V2V adversary.

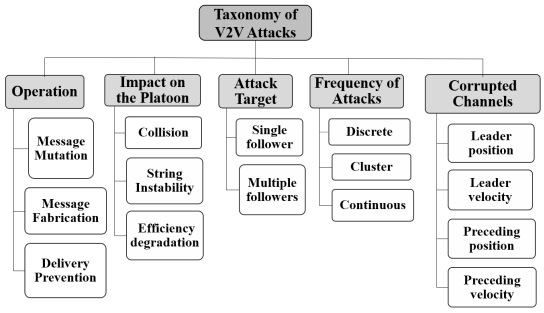

Fig. 2: Taxonomy of V2V Communication Attacks

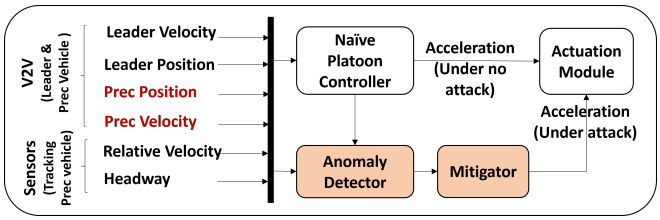

Fig. 3: REPLACE On-board Architecture

decomposes a communication attack into five classifying features: (i) operation, (ii) impact on the platoon, (iii) attack target, (iv) stealth or frequency of attacks, and (v) corrupted channels. Note that by abstracting the attack mechanism, any V2V attack can be characterized by these five features. For instance, delivery prevention accounts for diverse attack mechanisms including jamming, flooding, denial-of-service, etc.

IV. REPLACE ARCHITECTURE

Fig. 3 shows the on-board architecture of REPLACE. The key idea is to augment the platoon controller with two additional components: (1) Anomaly Detector and (2) Mitigator. The goal is to locally mitigate the adverse effects of malicious V2V communications received at the ego vehicle, thereby securing the platoon as a whole. Algorithm 1 defines the high-level functionality of REPLACE. We discuss the two components in more detail below. In summary, the Anomaly Detector determines if the inputs received at any instant are anomalous; if so, the Mitigator is invoked to override the corrupted naive controller output with a safe and efficient alternative.

A. Anomaly Detector Functionality

The Anomaly Detector component is an ML-based predictor implemented using a Random Forest (RF) regressor (See Section IV-C). The Anomaly Detector monitors V2V inputs received at each cycle and computes an expected normal response $\mathcal{N}$, which is compared with the controller response $\mathcal{R}$. If the deviation between $\mathcal{N}$ and $\mathcal{R}$ exceeds a pre-defined threshold then the inputs are treated as anomalous. According to the threat model considered in this work, an abnormality in the response indicates potentially corrupted preceding position values. The RF model is trained to estimate the normal response at cycle t by taking the readings from cycle $t-1$ as inputs. Note that the resiliency system is designed to prevent the propagation of the adverse effects from one cycle to the next (see Mitigator functionality below). Consequently, the corrected (*i.e.*, mitigated) inputs from $t-1$ can be safely used even under attack. Furthermore, the estimated output reference from the RF remains unaffected by any malicious communication received in cycle t.

B. Mitigator Functionality

The Mitigator is triggered when an anomaly is detected, to override the corrupted naive controller output with a safe and efficient alternative. Algorithm 2 describes the Mitigator functionality. It involves the following steps.

1) ***Correct the anomalous value*** with the help of previous preceding velocity and position values and by applying kinematics equations. This is essential to prevent propagation of corrupted inputs to the subsequent cycles of operation.
2) ***Compute alternate responses*** by re-invoking the naive controller with corrected input. Multiple alternatives are generated by changing the controller constant, T_{gap} from the original value of 0.8s to 1.2s in steps of 0.5. This is done in order generate conservative alternatives that account for possible inaccuracies in the previous correction step and ensure safety under attack.
3) ***Find the optimal response*** that meets the safety and efficiency constraints. We ensure safety of the system by evaluating all the alternatives under a hypothetical worst case scenario in terms of safety, *i.e.*, where the preceding vehicle decelerates at its maximum rated value. All the alternatives that are deemed unsafe under this scenario are discarded and the most efficient alternative among the safe choices is applied.

Algorithm 1 REPLACE Functionality: Ego Vehicle $\mathcal{E}$

1: $vel_{\mathcal{L}}, pos_{\mathcal{L}}, vel_{\mathcal{P}}, pos_{\mathcal{P}} \leftarrow ReadV2V()$
2: $a_{\mathcal{E}} \leftarrow NaiveCtrl()$
3: $a_{\mathcal{E}}^{ref} \leftarrow RFRegressor()$
4: $anmly\_flag \leftarrow Detector(a_{\mathcal{E}}^{ref}, a_{\mathcal{E}}, threshold)$
5: **if** no communication received **then**
6: $\quad no\_comm \leftarrow$ TRUE
7: **if** anomly_flag **or** no_comm is TRUE **then**
8: $\quad a_{\mathcal{E}} \leftarrow Mitigator()$ ***mitigator invoked***
9: $throttle, braking \leftarrow ActuarialControl(a_{\mathcal{E}})$

C. Off-line ML Training

Given the computational limitation of automotive platforms and the real-time requirements of the resiliency applications, any resiliency solution must be computationally light-weight. REPLACE ensures viability in this environment through the following observations.

Algorithm 2 Mitigator Functionality

1: $pos_{\mathcal{P}} \leftarrow KinematicDeduction()$ ***Rectify anomaly***
2: $alt\_resps \leftarrow ComputeAlternateRepsonses()$
3: $SimulateWorstCase()$
4: **for** resp in alt_resps **do**
5: $tgap \leftarrow ComputeTgap(worst\_case)$
6: **if** tgap $<$ safe_tgap **then**
7: Discard unsafe candidate
8: $a_{\mathcal{E}} \leftarrow FindOptimalCandidate(safe\_resps)$
9: **return** $a_{\mathcal{E}}$

1) The computationally intensive component of an ML solution is training. Consequently, REPLACE does not require real-time training on automotive platforms. Rather the usage model of REPLACE is that the ML components are trained off-line, possibly in the cloud, and the trained instances are downloaded to the automotive platform. Furthermore, real-time connectivity to cloud is not required during driving. Model update can be performed periodically via a secure connection.
2) Real-time prediction component uses the trained ML models created above. Although computationally far less demanding than training itself, a complex ML model does incur computational and storage costs during inference. To address this issue, the REPLACE architecture is designed to be independent of model specifics. We consequently choose the most light-weight ML model that still performs effective prediction under a resource-constrained environment. Based on significant experiments with several ML models, we found RF regressor to be optimal for our resiliency solution. Nevertheless, we leave open the possibility of other ML models being potentially appropriate based on field data. The choice in a specific case would depend on the trade-offs necessary between accuracy and computation cost.

Table I provides training hyper-parameters and input-output features used for our RF regressor. In our implementation, training is carried out to achieve accurate anomaly detection when deployed in any vehicle in the platoon regardless of its relative position with respect to the platoon leader. Note that this is an important design choice that allowed the distributed resiliency architecture to seamlessly generalize to platoons of arbitrary lengths.

D. Anomaly Detection Threshold Selection

In addition to model quality, detection accuracy relies heavily on the appropriate choice of the anomaly threshold. This value is determined after a series of iterations to balance the trade-off between false positives and false negatives. False negatives represent missed anomalies while false positives indicate normal samples incorrectly identified as anomalous. A lower threshold can potentially reduce the number of false negatives while at the same time, could increase the false positives. Owing to the safety critical nature of the platoon application, we favor lower false negatives and tolerate false positives in our design.

TABLE I: RF Anomaly Detector Training Details

Input-output features	
Input Features	Ego Vehicle Params: $\{vel_{\mathcal{E}}, pos_{\mathcal{E}}\}$ Preceing Vehicle Params: $\{vel_{\mathcal{P}}, pos_{\mathcal{P}}\}$ Leader Params: $vel_{\mathcal{L}}$
Output Feature	Ego acceleration ($a_{\mathcal{E}}$)
Architecture and training hyper-parameters	
Architecture Parameters	Number of trees: 100 Max number of levels in tree: 10
Training hyper-parameters	Minimum samples split: 2 Split criteria: Gini index Sampling: Bootstrap

TABLE II: V2V Attacks Orchestrated

Attack ID	Operation	Target Vehicles	Bias	Attack Frequency
Attack 1	Mutation	Vehicle 4	+28	Continuous
Attack 2	Mutation	Vehicle 4	+30	Cluster
Attack 3	Mutation	Vehicles 3,5,7	+12 each	Continuous
Attack 4	Mutation	Vehicles 3,5,7	+18 each	Cluster
Attack 5	Delivery Prevention	Vehicle 4	-	Cluster
Attack 6	Delivery Prevention	Vehicles 3,5,7	-	Cluster

V. EXPERIMENTAL ANALYSIS

A. Experimental Setup

We developed an evaluation testbed by integrating data from a state-of-the-art automotive research simulator [15] with a software prototype of REPLACE. From the simulator we collected fine-grained driving trajectory data at a frequency of 100Hz, which represented the leader trajectory for the platoon. Subsequently, the controller prototype is invoked to create the trajectories of the follower vehicles. We simulated V2V communications at the same rate of 100Hz using the information flow topology explained in Section II-B. Putting it all together, we created a platoon with a leader followed by 7 vehicles. Vehicle ID 0 indicates the leader. Vehicle IDs 1-7 indicate the followers respectively with vehicle 1 being closest to the leader.

B. Attack Orchestration

We orchestrated 6 representative classes of attacks disrupting the platoon by causing a collision between one or more pairs of vehicles. The attacks are systematically orchestrated following our taxonomy to ensure that the attacks represent the V2V adversary spectrum. We fix the features "impact" and "corrupted channel" in all the representative attacks to be *collision* and *preceding position* respectively. All combinations of the remaining three features in our taxonomy are considered in our representative attacks. These attacks are orchestrated by generating fake position readings by adding a bias to the ground truth or by preventing delivery. Each attack scenario is 50 seconds long (5000 samples). The resiliency system remains dormant during the initial

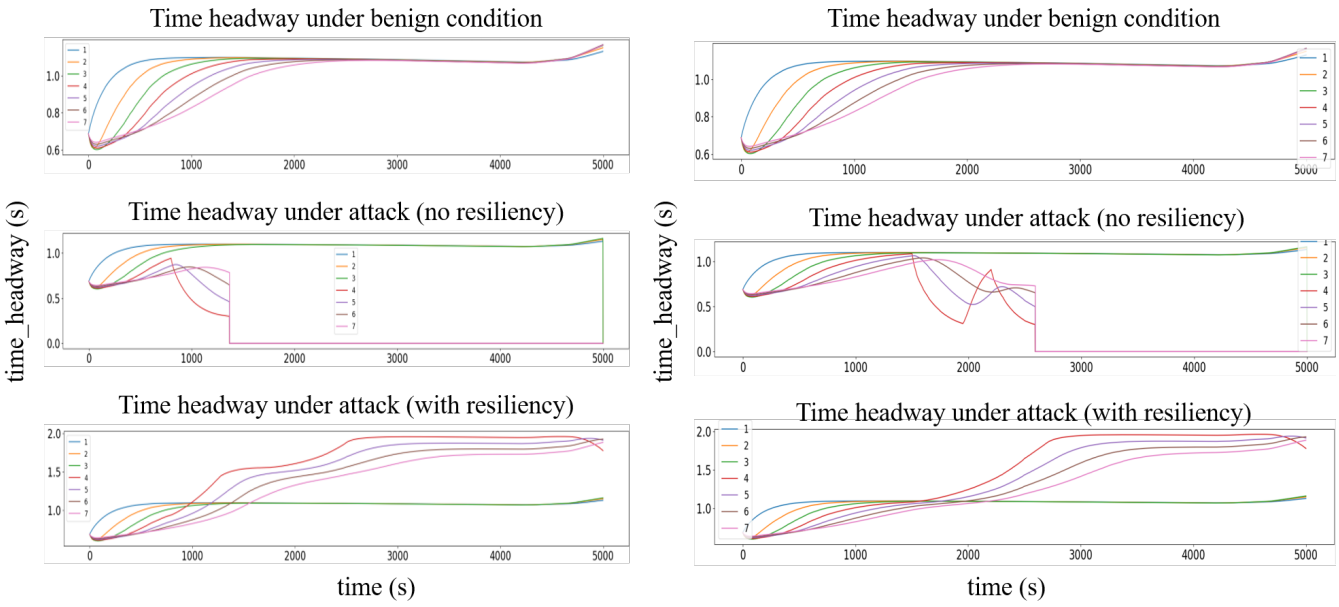

Fig. 4: Mutation attacks. (a) Attack 1: Continuous attack on a single target vehicle (ID: 3); (b) Attack 2: Cluster attack on a single target vehicle (ID: 3)

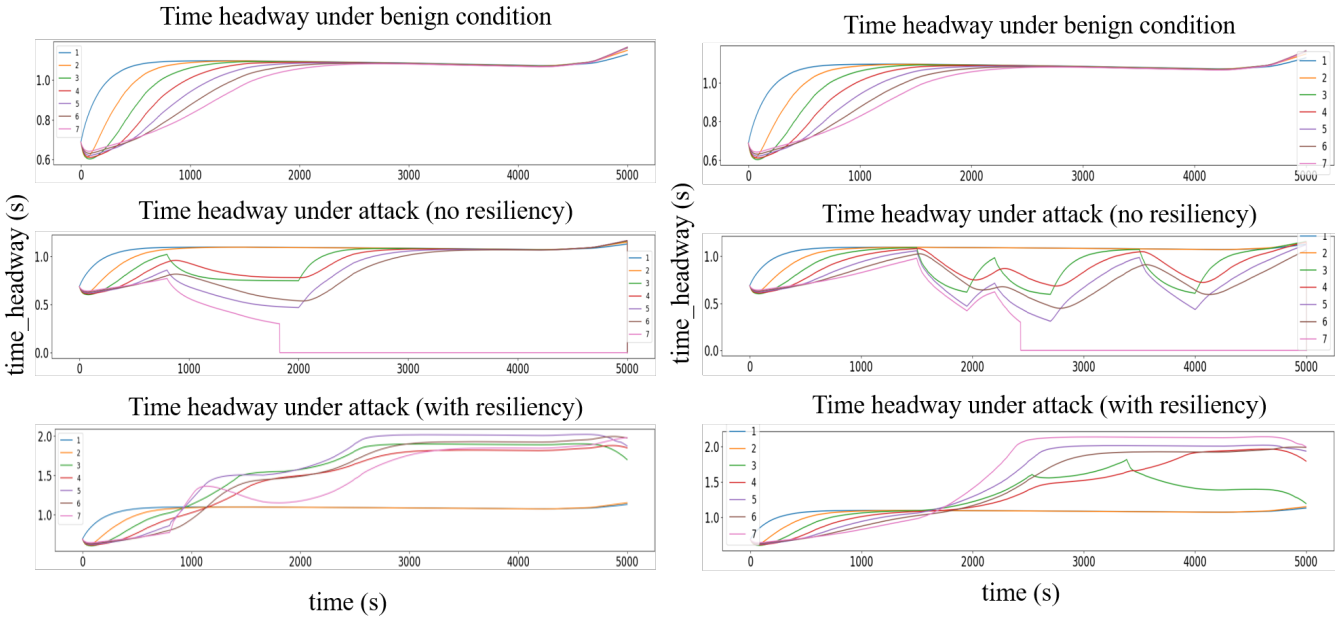

Fig. 5: Mutation attacks. (a) Attack 3: Continuous attack on multiple target vehicles (ID: {3, 5, 7}); (b) Attack 4: Cluster attack on multiple target vehicles ((ID: {3, 5, 7}))

platoon stabilization time, which is the first 5s (500 samples). No anomalies are injected in the reported communication during this stabilization time. We orchestrate various mutation attacks and delivery prevention attacks as shown in Table II. For a continuous attack, the adversarial activity takes place throughout the test scenario starting from sample 800. For cluster attacks, adversarial activity takes place between the sample intervals: {1500-1950}, {2200-2700}, and {3500-4000}. Additionally, under mutation attacks, a bias (positive or negative) is added to the ground truth either to increase the risk of collision.

C. Resiliency Evaluation

We use time headway as the metric to evaluate the efficacy of REPLACE under both benign and adversarial condition. Since the controller is targeted to keep a headway of 0.8s, the time headway between pairs of vehicles in practice under benign condition with no resiliency is roughly 1-1.2s. A

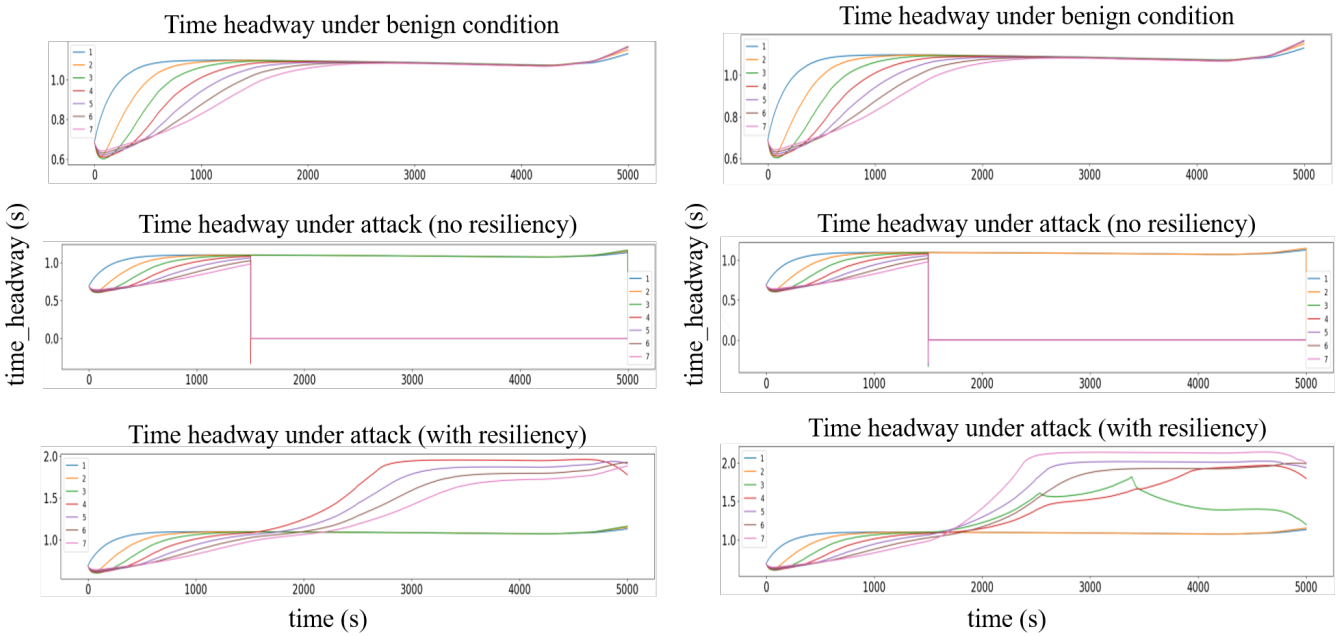

Fig. 6: Delivery prevention attacks. (a) Attack 5: Cluster delivery prevention attack on a single target vehicle (ID: 3); (b) Attack 6: Cluster delivery prevention attack on multiple target vehicles (ID:{3, 5, 7})

reduction in time headway represents a compromise in safety, with a headway near 0 representing a collision.

Figs. 4, 5 and 6 plot the resultant time headways between each pair of vehicles under (1) benign condition; (2) naive controller under attack, and (3) REPLACE under attack. Note that under Attacks 1 and 2 in Fig. 4 with no resiliency, there is a collision between vehicles 3 and 4 (the direct target of corrupted communication) at approximately sample 1300. This also results in subsequent collision among all the vehicles following 4 (indirect victims). More interestingly, consider the cluster attacks in Fig. 5. Under no resiliency, there is collision between vehicles 6 and 7. Note that while the direct targets of the attack are different (*i.e.*, vehicles 3, 5, and 7) the victims impacted most are 6 and 7. Finally, under Attack 5 and 6 shown in Figure 6, communication is jammed to a single vehicle 4 and a subset of vehicles 3, 5, and 7 respectively. Under Attack 5, collision occurs between vehicles 3 and 4, in turn causing a collision between all the subsequent vehicles following 4. Under Attack 6, collision occurs between vehicles 2 and 3 as well as all the subsequent pairs. Note that with REPLACE all the attacks are mitigated.

We make several important observations from these results. First, REPLACE clearly provides resiliency and ensures safe operation under all the different classes of attack. To achieve this, REPLACE incurs a small additional overhead due to mitigation, *e.g.*, the overall time headway under attacks increases slightly to values between 1.2-2s. Second, note that the attack orchestrations show that the impact of an attack may be on a vehicle that is not the direct target of the attack (*i.e.*, the vehicle actually receiving corrupted communication). Third, the impact of a mutation attack becomes more pronounced and the occurrence of collision is faster as the deviation between the reported communication and ground truth becomes larger. Furthermore, the longer the duration of uninterrupted adversarial activity, more impactful is the attack, *i.e.*, continuous attacks are more impactful than cluster attacks. This suggests a trade-off between stealth and impact in adversarial activities on V2V communications.

VI. RELATED WORK AND DISCUSSION

There has been significant recent research in designing V2V-based platoon controllers [4], [9], [16]. However, exploration of adversaries corrupting this communication has been limited. Engoulou *et al.* [8] presents a survey of the security challenges in VANETs and also proposes different secure architectures. Existing research in security of platoons involves control theoretic defenses against specific classes of adversarial activities such as denial of service attacks [1], [3], [5]. However, these solutions do not generalize to other V2V adversaries.

There has been some research on secure Cooperative Adaptive Cruise Control (CACC), which is a 2-vehicle car following application. CACC also forms the basis of some multi-car platooning implementations. Heijden *et al.* [18] propose a mechanism for misbehavior detection based on subjective logic that involves validating the exchanged position information between the participating vehicles. Nunen *et al.* [19] propose a control-theoretic approach to mitigate packet dropouts and communication failures in CACC. To discuss a few machine learning approaches, Iorio *et al.* [10] propose a mechanism for detecting injection attacks based on correlation between different parameters corresponding to vehicular movements. Alotibi *et al.* [2] propose a detection approach for platoon systems with a compromised leader

vehicle communicating falsified acceleration values to the rest of the platoon.

A unique aspect of REPLACE is *real-time resiliency* rather than offline detection. This goal dictated several components of the design and evaluation, including efficiency constraints imposed by vehicular electronics. Furthermore, one has to account for *sustained* attacks. Note from Section V that a continuous attack can have significant impact over time, even if the deviation from normal behavior is small. However, comprehending continuous behavior requires comprehending not only the attack progression but also the response from the ego vehicle. Finally, it is important to distinguish the multi-vehicle platooning considered here from 2-vehicle CACC. A resiliency system for multi-vehicle platoons must not only mitigate attacks on the direct victim vehicles but should minimize propagation of the adverse effects down the platoon, *e.g.*, recall that the impacts of some attacks are more pronounced on vehicles that did not directly receive corrupt communications. To address this, the detection system must distinguish between slight abnormalities resulting from ongoing mitigation by vehicles ahead of it from actual adversarial communication it receives. Correspondingly, mitigation requires accurate computation of optimal alternate responses to avoid inaccuracies from cascading and piling up further down the platoon.

VII. CONCLUSION AND FUTURE WORK

We presented a novel distributed resiliency architecture, REPLACE for CAV platoon vehicles to defend against adversaries corrupting V2X communication reporting preceding vehicle position. This is one of the only real-time detection and mitigation infrastructures developed for platoons that can comprehensively defend against the entire spectrum of communication attacks. We developed a taxonomy of attacks to systematically navigate the adversarial space. We orchestrated various representative collision-causing attacks targeting one or more vehicles in the platoon. We implemented a flexible evaluation testbed integrating realistic driving data from a state-of-the-art automotive simulator. We demonstrated the efficacy of our ML-based resiliency system in preventing collisions under various communication attacks targeting one or more vehicles in the platoon. In addition to preserving safety of the platoon, the mitigation ensures optimal efficiency close to ideal platoon operation even under attack for each vehicle.

In future work, we will extend the threat model to account for other corrupted channels in addition to preceding position. We will carry out attack impact analysis and develop effective detection and mitigation systems to defend against adversaries exploiting this communication.

Acknowledgements: This research has been supported in part by the National Science Foundation under Grant No. CNS-1908549.

REFERENCES

[1] Z. Abdollahi Biron, S. Dey, and P. Pisu. Real-time detection and estimation of denial of service attack in connected vehicle systems. *IEEE Trans. Intelligent Transportation Systems*, 19(12):3983–3902, 2018.

[2] F. Alotibi and M. Abdelhakim. Anomaly detection for cooperative adaptive cruise control in autonomous vehicles using statistical learning and kinematic model. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–11, 2020.

[3] M. H. Basiri, N. L. Azad, and S. Fischmeister. Attack resilient heterogeneous vehicle platooning using secure distributed nonlinear model predictive control. In *2020 28th Mediterranean Conference on Control and Automation (MED)*, pages 307–312. IEEE, 2020.

[4] C. Bergenhem, S. Shladover, E. Coelingh, C. Englund, and S. Tsugawa. Overview of platooning systems. In *Proceedings of the 19th ITS World Congress, Oct 22-26, Vienna, Austria (2012)*, 2012.

[5] J. J. Blum, A. Neiswender, and A. Eskandarian. Denial of service attacks on inter-vehicle communication networks. In *2008 11th International IEEE Conference on Intelligent Transportation Systems*, pages 797–802. IEEE, 2008.

[6] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno. Comprehensive experimental analyses of automotive attack surfaces. SEC'11, page 6, USA, 2011. USENIX Association.

[7] R. G. Dutta, F. Yu, T. Zhang, Y. Hu, and Y. Jin. Security for safety: A path toward building trusted autonomous vehicles. In *ICCAD*, 2018.

[8] R. G. Engoulou, M. Bellaïche, S. Pierre, and A. Quintero. Vanet security surveys. *Computer Communications*, 44:1–13, 2014.

[9] S. Gong and L. Du. Cooperative platoon control for a mixed traffic flow including human drive vehicles and connected and autonomous vehicles. *Transportation research part B: methodological*, 116:25–61, 2018.

[10] M. Iorio, F. Risso, R. Sisto, A. Buttiglieri, and M. Reineri. Detecting injection attacks on cooperative adaptive cruise control. In *2019 IEEE Vehicular Networking Conference (VNC)*, pages 1–8, 2019.

[11] M. Jagielski, N. Jones, C.-W. Lin, C. Nita-Rotaru, and S. Shiraishi. Threat detection for collaborative adaptive cruise control in connected cars. In *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, pages 184–189. ACM, 2018.

[12] C. Miller and C. Valasek. Remote exploitation of an unaltered passenger vehicle. *Black Hat USA*, 2015:91, 2015.

[13] National Highway Traffic Safety Association. Road Accidents In USA. See URL: `https://www.recalls.gov/nhtsa.html`.

[14] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl. Remote attacks on automated vehicles sensors: Experiments on camera and lidar. *Black Hat Europe*, 11:2015, 2015.

[15] Realtime-Technologies. Physical Automotive Simulator. See URL: `https://www.faac.com/realtime-technologies/products/rds-1000-single-seat-simulator`.

[16] S. Santini, A. Salvi, A. S. Valente, A. Pescapè, M. Segata, and R. L. Cigno. A consensus-based approach for platooning with inter-vehicular communications. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 1158–1166. IEEE, 2015.

[17] A. Tiwari, B. Dutertre, D. Jovanović, T. de Candia, P. D. Lincoln, J. Rushby, D. Sadigh, and S. Seshia. Safety envelope for security. In *Proceedings of the 3rd International Conference on High Confidence Networked Systems*, HiCoNS '14, pages 85–94, 2014.

[18] R. W. van der Heijden et al. Enhanced position verification for vanets using subjective logic. In *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Sep. 2016.

[19] E. van Nunen et al. Robust model predictive cooperative adaptive cruise control subject to v2v impairments. In *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Oct 2017.