Multi-domain Denial-of-Service Attacks in Internet-of-Vehicles: Vulnerability Insights and Detection Performance

Roshan Sedar*, Charalampos Kalalas*, Jesus Alonso-Zarate†, Francisco Vázquez-Gallego†
*Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), Barcelona, Spain
†i2CAT Foundation, Barcelona, Spain
{roshan.sedar, ckalalas}@cttc.es, {jesus.alonso, francisco.vazquez}@i2cat.net

***Abstract*—The transformative Internet-of-Vehicles (IoV) paradigm comes inadvertently with challenges which involve security vulnerabilities and privacy breaches. In this context, denial-of-service (DoS) attacks may perniciously affect the normal operation of IoV systems by causing extensive periods of network unavailability where legitimate vehicles are prevented from accessing vehicular services. In this paper, we offer an in-depth vulnerability assessment of 5G-enabled IoV systems when DoS attack variants are launched at multiple network domains. We further evaluate the resilience of an IoV-tailored authentication mechanism against DoS attacks under various configurations. A data-driven detection scheme is also proposed to address DoS variants in the radio access network, which take the form of false data injection attacks on the exchanged vehicular information. Our performance assessment with the aid of an open-source dataset reveals that the proposed scheme is able to accurately detect DoS traffic originated from malicious vehicles.**

***Index Terms*—IoV, DoS, authentication, cuckoo filter, reinforcement learning, data-driven detection.**

I. Introduction

With the increasing level of driving automation in 5G-enabled vehicular use cases, vehicular communication becomes highly vulnerable to malicious actors, opening up entirely new questions from a security and privacy perspective that have not been addressed in a similar context before. This is particularly important for Internet-of-Vehicles (IoV) systems, where vehicles become highly aware of their surroundings resulting in increased connectivity levels with each other and with relevant roadside entities [1]. Vulnerabilities and security breaches in IoV render the attack surface sufficiently large with multi-faceted threat vectors, which an adversary may maliciously exploit to intrude in the system. As safety and security are tightly coupled in IoV environments, security attacks may compromise the safety of road users and lead to serious accidents. Novel security mechanisms are thus essential to address such scenarios and reduce the extent of their detrimental effects on safety-critical vehicular use cases.

Among various attack types, denial-of-service (DoS) attacks constitute one of the major threats against network and service availability [2]. Availability ensures that the vehicular network is functional and the services/information are available at any time for its users. In DoS attacks, an attacker tries to prevent legitimate users from accessing the network and services, and causes severe traffic disruption which may destabilize the IoV communication system and threaten user safety. In particular, DoS attackers typically flood the network with traffic of higher frequency than the system can handle, resulting in overflow and extensive periods of network unavailability where legitimate users cannot be served. DoS is generally considered as one of the most frequent and effective attacks against vehicular networks [3]. When DoS attacks are launched from spatially distant physical locations, this results in the distributed DoS (DDoS) attack variants [4]. DoS attacks may also be launched against multiple network domains in the 5G architecture, such as the core, edge and radio access network (RAN), and may target multiple levels of the protocol stack, i.e., application, transport, network, data link and physical layers. Therefore, security mechanisms are often necessary to be deployed in a multi-domain context, aiming to provide protection for end-to-end vehicular services against such threats.

Considering multi-domain IoV environments, this paper thoroughly explores multiple DoS attack types launched against different components of the vehicular system. In particular, in an effort to shed light on the vulnerabilities of the IoV-tailored 5G authentication and key agreement (5G-AKA) mechanism proposed in [5], we examine the feasibility of a DoS attack execution against the authentication server function (AUSF) which may significantly degrade system performance. We further investigate the impact of various design configurations on the resilience of the authentication scheme, and provide useful insights to decrease the effectiveness of such attacks. While DoS attacks may compromise core functional components involved in IoV authentication, the availability of the communication channel in the RAN domain may also be disrupted by such malicious actions. In this context, a data-driven reinforcement learning (RL) approach is introduced to detect and identify DoS attacks in IoV time-series data exchanged in IoV environments. While such attacks may be difficult to detect due to the attacker's erratic behavior over time, our proposed scheme is shown to accurately detect such malicious behaviors, and yields superior performance compared to a benchmark approach. RL-based DoS detection is capable of improving detection experience over time while adapting to the rapidly changing IoV environments, without relying on security threshold values [6].

The remainder of the paper is organized as follows. Section II describes a DoS attack targeting an IoV-tailored authentication mechanism, and presents an evaluation of its resilience under various configurations. Section III elaborates the RL-based detection mechanism proposed to address multiple DoS attack variants in RAN, and presents a performance comparison against a benchmark approach. Finally, Section IV is reserved for conclusion and discussion of the path forward.

II. Denial-of-Service Attack in IoV-tailored 5G-AKA Procedure

A. Vehicle authentication based on cuckoo filter

As illustrated in Fig. 1, the 5G-AKA procedure enables mutual authentication between the vehicle and the network, and provides keying material that can be used in subsequent security procedures. However, in highly dense vehicular scenarios, the excessive signalling overhead required for security context establishment in 5G-AKA may result in increased latency beyond the acceptable levels [7]. In [5], an enhancement of the standard 5G-AKA mechanism was introduced, aiming to address highly dense IoV connectivity scenarios. The proposed scheme leverages the space-efficient advantages of a cuckoo filter implementation to enable the authentication of multiple vehicles at a time with controllable false positive rates. Its key phases mainly involve the signalling exchange between the AUSF and the security anchor function (SEAF), and are outlined as follows:

1) Cuckoo filter generation: The AUSF generates a cuckoo filter where the expected response (XRES^*) values from the received 5G home environment authentication vectors (5G HE AVs) are inserted. The cuckoo filter, hereinafter denoted as $\text{CF}_{\text{XRES}^*}$, offers a compact probabilistic way to represent an XRES^*, by storing its fingerprint f_{XRES^*} in a hash table consisting of an array of buckets. Each inserted XRES^* has two candidate buckets determined by the hash functions h_1 and h_2. In particular, the indices of the candidate buckets for an XRES^* are computed based on partial-key cuckoo hashing [8], as $i_1 = h_1(\text{XRES}^*)$ and $i_2 = i_1 \oplus h_2(f_{\text{XRES}^*})$. Using the item insertion operation for each XRES^*, the AUSF stores the fingerprints corresponding unequivocally to the vehicles which requested network registration. Each fingerprint in $\text{CF}_{\text{XRES}^*}$ consists of a bit string derived from XRES^* using a hash function, and its length is determined based on the target false positive rate. The 5G serving environment authentication vector (5G SE AV) is then constructed by concatenating $\text{CF}_{\text{XRES}^*}$ with the random challenge (RAND) and the authentication token (AUTN), as 5G SE AV = RAND $\|$ $\text{CF}_{\text{XRES}^*}$ $\|$ AUTN, and it is subsequently sent to the SEAF.

2) RES^* verification: After receiving an authentication response message from a vehicle, the SEAF extracts the RES^*, calculates its fingerprint f_{RES^*}, and performs a set membership query in $\text{CF}_{\text{XRES}^*}$. In particular, the SEAF calculates the output of $h_1(\text{RES}^*)$ and $h_2(\text{RES}^*)$ to derive the indices of the two candidate buckets where f_{RES^*} may be stored. If, in neither of the two locations in $\text{CF}_{\text{XRES}^*}$, the stored fingerprint f_{XRES^*} does not coincide with f_{RES^*}, then authentication fails.

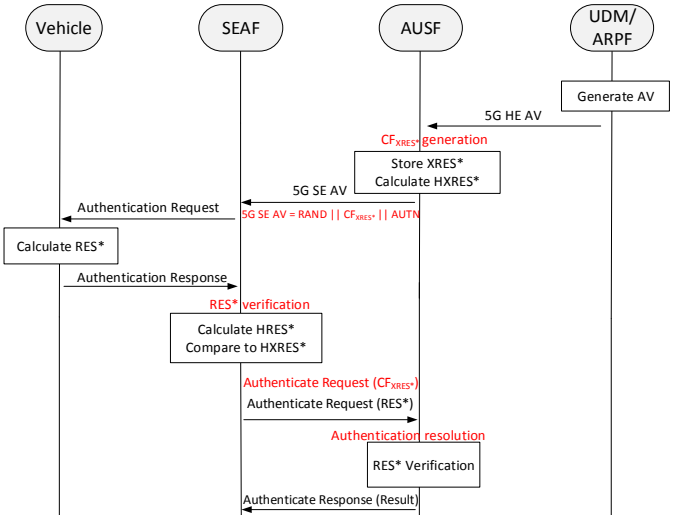

Fig. 1: Signalling flow in standardized 5G-AKA procedure [9] and proposed extensions (highlighted in red).

Otherwise, if f_{XRES^*} is identical to f_{RES^*} in either of the two buckets, then the authentication may be successful, as a false positive may have occurred.

3) Authentication resolution: After the RES^* verification phase, the SEAF notifies the AUSF about the resolution of the vehicle authentication. Using the item deletion operation in $\text{CF}_{\text{XRES}^*}$, the SEAF removes the non-matching fingerprints, leaving stored only those fingerprints correctly authenticated during the previous phase. The $\text{CF}_{\text{XRES}^*}$ is then transmitted to the AUSF as part of the authenticate-request message.

B. Denial-of-service attack description

The performance analysis in [5] reveals that a properly designed $\text{CF}_{\text{XRES}^*}$ can significantly improve the authentication efficiency of the standardized 5G-AKA scheme. Gains in terms of end-to-end latency and protocol overhead can also be attained, even for high vehicle densities. The introduced space cost, in terms of required number of bits per inserted XRES^*, remains close to the information-theoretic lower bound, even for stringent false positive rate requirements.

Despite the benefits of the extended 5G-AKA scheme, vulnerabilities in the $\text{CF}_{\text{XRES}^*}$ generation phase may give rise to DoS attacks which take the form of consecutive XRES^* insertion failures. As discussed in [10], such attacks aim at forcing insertion failures in the filter, and can be executed without prior knowledge of the implementation components, e.g., hash functions h_1 and h_2 used in the filter. An XRES^* insertion attack may eventually result in service unavailability, since the $\text{CF}_{\text{XRES}^*}$ becomes unable to locate a free bucket to store an XRES^*, regardless of the number of displacements allowed during insertion.

To execute an XRES^* insertion attack, the attacker first creates a set of candidate XRES^* and queries the $\text{CF}_{\text{XRES}^*}$ for each of them, keeping in a set N those XRES^* that return a negative. Then, the attacker inserts an arbitrary XRES^*_k in $\text{CF}_{\text{XRES}^*}$, and queries again the XRES^* belonging in N. Those XRES^* returning a positive are then added to a set

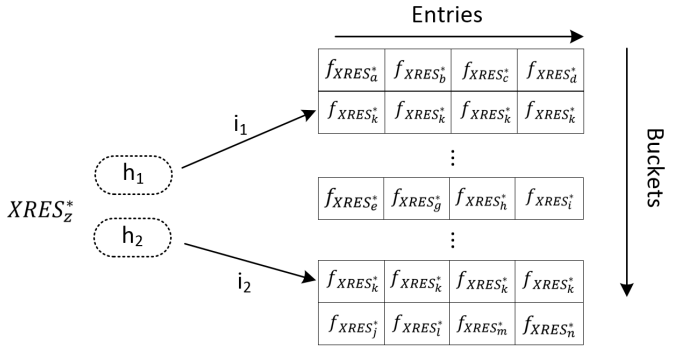

Fig. 2: XRES* insertion failure in $\text{CF}_{\text{XRES}^*}$ when the last element XRES_z^* belonging in set P is inserted.

P. Based on the cuckoo filter principles, the XRES* stored in P have generated a false positive with XRES_k^*, sharing the same location indices and fingerprint $f_{\text{XRES}_k^*}$. The attacker iterates the aforementioned procedure until the number of XRES* stored in P reaches $2b$, where b denotes the number of entries per bucket. Those $2b$ XRES* are then inserted to the $\text{CF}_{\text{XRES}^*}$, resulting in an insertion failure for the last inserted $\text{XRES}_z^* \in P$, as illustrated in Fig. 2. This occurs due to the fact that $f_{\text{XRES}_k^*}$ is already stored in $\text{CF}_{\text{XRES}^*}$, and the maximum number of fingerprints in two buckets is limited to $2b$.

C. Filter design

The effectiveness of XRES* insertion attacks highly depends on the $\text{CF}_{\text{XRES}^*}$ parameter configuration during the generation phase at the AUSF. In order to assess the $\text{CF}_{\text{XRES}^*}$ vulnerability in such type of attacks, the XRES* insertion failure probability needs to be determined. Let us consider a $\text{CF}_{\text{XRES}^*}$ of m buckets with b entries per bucket, and that the AUSF receives multiple XRES* corresponding to a number of N vehicles requesting network registration. We further assume that $m = cN$ for a constant $c > 0$, while f denotes the fingerprint size (in bits). An arbitrary XRES_k^* is inserted in the filter and its fingerprint $f_{\text{XRES}_k^*}$ is stored. According to the item insertion principles of the cuckoo filter, if $j-1$ XRES* have the same buckets with XRES_k^*, the following must simultaneously hold: *i*) their location indices derived from h_1 and h_2 are the same, which occurs with probability $2/m$, and *ii*) they have the same fingerprint, which occurs with probability $1/2^f$. Therefore, the conditional probability of j in total XRES* sharing the same two buckets is $\left(2/m \cdot 1/2^f\right)^{j-1}$, and when $j = 2b+1$, an insertion failure occurs. The XRES* insertion failure probability can thus be expressed as

$$p_f = \left(\frac{2}{2^f \cdot m}\right)^{2b} = \left(\frac{2}{2^f \cdot cN}\right)^{2b}. \quad (1)$$

Fig. 3 depicts the XRES* insertion failure probability as a function of the number of vehicles N for various $\text{CF}_{\text{XRES}^*}$ configurations. In particular, we assess the vulnerability of $\text{CF}_{\text{XRES}^*}$ for different combinations of entries per bucket b, and fingerprint sizes f. It can be noticed that XRES* insertion failure probability registers an increase with rising number of vehicles, rendering the $\text{CF}_{\text{XRES}^*}$ more susceptible to insertion

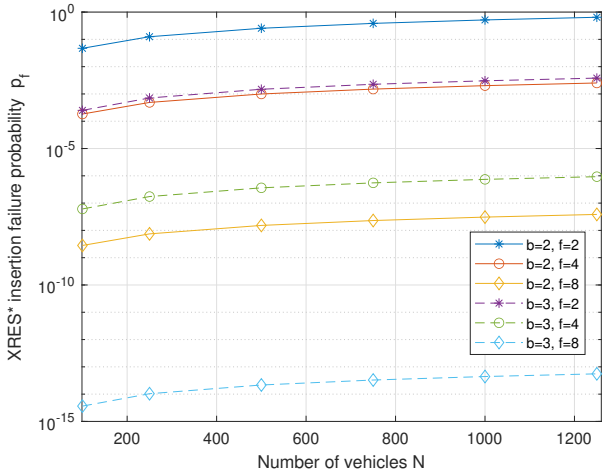

Fig. 3: XRES* insertion failure probability for various $\text{CF}_{\text{XRES}^*}$ configurations and increasing number of vehicles.

attacks. However, insertion failures are less probable when a larger bucket size is considered; thus, larger $\text{CF}_{\text{XRES}^*}$ are more resistant to such attacks. This can be intuitively explained as follows. When the number of entries per bucket increases, the set of candidate locations for the inserted XRES* expands, resulting in a lower probability for groups of $2b+1$ XRES* mapping to the same buckets. Therefore, a higher number of queries is required by the attacker to store at least $2b$ XRES* in P. In addition, XRES* insertion failure probability reduces when a longer fingerprint f is used, since the probability of an exact fingerprint match becomes lower. Thus, a larger fingerprint size renders $\text{CF}_{\text{XRES}^*}$ less prone to insertion attacks.

III. Denial-of-Service Attack in IoV-based RAN

Besides the DoS attacks which may often originate from malicious outsiders/intruders (i.e., exogenous to the original system) targeting the 5G-AKA procedure, a set of DoS attack variants can also be launched in the RAN domain by already authenticated insiders which possess valid system credentials. Insider attacks are often difficult to detect and contain, particularly when attackers behave intelligently while conforming to normal system behavior. For example, an authenticated vehicle may intentionally transmit false kinematic information in its broadcast messages and cause disruption in the RAN.

Fig. 4 illustrates an IoV deployment with an edge network empowered by software-defined networking (SDN) technology, which hosts vehicular services on the edge servers. The edge servers are deployed closer to roadside units (RSUs) to achieve low-latency communication for safety-related tasks. The participating vehicles periodically broadcast basic safety messages (BSMs) which convey mobility state information for a vehicle, including position, speed, acceleration and heading angle, as well as other relevant parameters. BSMs are typically transmitted in a standardized format with non-encrypted content in order to support safety applications, allowing all other participating entities in the RAN to read these BSMs.

An RSU receives BSMs sent from vehicles within its communication range, and the edge cloud server aggregates

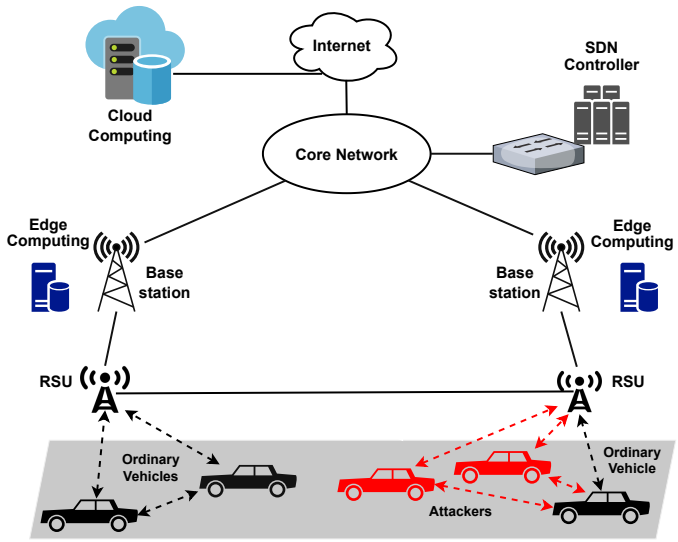

Fig. 4: An IoV network architecture empowered by software-defined networking for low-latency vehicular connectivity.

information from various RSUs deployed in a large geographical region. In principle, RSUs are assumed to be trustworthy infrastructure nodes while connected vehicles are more vulnerable to attacks, and can be compromised by adversaries at any time after joining the network.

A. Denial-of-service attack description

In the IoV scenario depicted in Fig. 4, an attacker is a *misbehaving* vehicle which transmits incorrect information embedded in the BSMs with malicious or selfish intents. Misbehaving vehicles are defined as malicious entities whose behavior deviates from the normal one and transmit intentionally falsified information to mislead other genuine IoV entities [6]. The attacker is considered as a rogue insider, and it is able to launch a wide range of DoS variants, by combining various malicious attacks. For example, a DoS attack may be executed in *Sybil* mode in order to conceal the real identity of the attacker. In the *Sybil* mode, the attacker uses multiple valid pseudonym certificates that have been extracted from compromised vehicles to realize this type of attack [11]. In turn, the victim vehicles are being tricked to accept incoming malicious messages due to the use of valid pseudonym certificates.

In the following, we briefly describe several DoS attack types considered in this work, as defined in the open-source vehicular reference misbehavior (VeReMi) extended dataset [12].

DoS attack: A misbehaving vehicle transmits BSMs at a very higher frequency than the acceptable limit set by the standard specifications. This results in a high volume of data transmission causing extensive periods of network congestion and unavailability of critical services for the legitimate vehicles.

DoS Random attack: In this attack, the attacker sets all BSM fields to random values and performs a typical DoS attack.

DoS Disruptive attack: An attacker vehicle may re-transmit previously transmitted BSMs by other legitimate vehicles. BSMs are selected at random and flood the network with stale data with the intention of disrupting genuine information from being propagated. The attacker increases BSMs transmission rate in order to realize the DoS disruptive attack.

DoS Random Sybil attack: The attacker changes pseudonym identities on every transmitted BSM while performing the DoS random attack.

DoS Disruptive Sybil attack: In this attack, the attacker changes pseudonyms on every re-transmission of previously received BSMs while performing the DoS disruptive attack.

B. DoS detection using reinforcement learning

1) Preliminaries: Markov decision processes (MDPs) provide a framework for sequential decision-making problems that can be utilized in modelling time-series anomaly detection. In particular, the action of DoS detection will change the environment based on the decision of either genuine or malicious behavior at time-step t; subsequently, the next decision at time-step $t+1$ will be influenced by the changing environment at previous time-step t. An MDP is defined as a tuple of five basic elements, i.e., $\mathcal{M} =< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma >$, where $\mathcal{S}$ denotes the set of states, $\mathcal{A}$ denotes the set of actions, $\mathcal{P} : \mathcal{S} \text{ X } \mathcal{A} \text{ X } \mathcal{S} \longmapsto [0, 1]$ denotes the state transition probabilities function, $\mathcal{R} : \mathcal{S} \longmapsto \mathbb{R}$ denotes the reward function which gives the set of possible rewards (i.e., reward set), and $\gamma \in (0, 1)$ denotes the discount factor which reflects the importance of the immediate and long-term future reward.

RL represents a family of algorithms for solving MDPs, which in turn can be applied in time-series anomaly detection [13]. As shown in Fig. 4, the aggregated information at the edge node (i.e., RSU) encompasses a time-series repository of received BSMs with intrinsic temporal and spatial interdependencies. The information contained in each BSM is constantly evolving along the vehicle trajectory while BSMs from neighboring vehicles exhibit high spatial dependency. Hence, misbehaving vehicles can be potentially detected by sequentially analyzing their mobility patterns using RL.

2) Reinforcement learning model: We consider an RL-based misbehavior detector which is deployed at the edge RSU node, acting as an agent that interacts with the IoV environment to learn the optimal detection policy π. Based on the current state s_t at time-step t, the agent takes an action a_t to maximise its reward r_t. The reward is offered to the agent by the environment, and subsequently the environment moves to a new state s_{t+1} following the MDP. This is done repeatedly until the optimal detection policy π is learned. In this work, value-based Q-learning method is adopted to train the RL model for estimating the action-value function $Q(s, a)$ [14].

In what follows, we briefly discuss the components pertaining to the RL model adopted in this work.

The **agent** takes the IoV time-series data and prior related decisions as inputs (i.e., state s_t), and generates the new decision made (i.e., action a_t) as output. The agent actions at each time-step t are selected by the detection policy π. Thus, the agent experience at each time-step, i.e., $e_t =< s_t, a_t, r_t, s_{t+1} >$, stores all the behaviors of the misbehavior

detector. By learning from experience, the misbehavior detector is consistently improved to obtain a better estimation of the $Q(s, a)$ function. This process is referred to as experience replay memory through which the model training is performed. The goal of the agent is to maximize the expected sum of future discounted rewards by learning the optimal detection policy. The discounted reward return is expressed as

$$R_t = \sum_{k=t}^{T} \gamma^{k-t} r_k, \tag{2}$$

where γ denotes the discount factor that specifies the importance of long-term rewards and T is the terminal step. The agent updates its model in order to improve the accuracy in decision-making [14]. The Q-value in Q-learning model can be updated iteratively according to

$$\begin{aligned} Q(s_t, a_t) \leftarrow & Q(s_t, a_t) \\ & + \alpha[r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)], \end{aligned} \tag{3}$$

where α denotes the learning rate.

The **environment** of the RL model controls the training of the agent. It takes the action a_t performed by the agent as its input, and consequently generates a reward r_t and the next environment state s_{t+1} for the agent. In this setting, the environment contains a large population of BSMs with misbehavior attack labels.

The **state** contains the sequence of previous actions denoted by $s_{action} =< a_{t-1}, a_t, ..., a_{t+n-1} >$, and the current BSM information denoted by $s_{time} =< X_t, X_{t+1}, ..., X_{t+n} >$. $X_t \in R^d$ is a d-dimensional feature vector at time-step t, including information on d different features. According to the state design, the next action taken by the agent depends on the previous actions and the current IoV information.

The **action** space is defined as $\mathcal{A}$ = {0,1}, where 1 indicates the detection of an attack and 0 represents the genuine behavior. The deterministic detection policy π can be expressed as a mapping, i.e., $\pi : \mathcal{S} \longmapsto \mathcal{A}$, from states to actions, where $\pi(s)$ denotes the action that the agent takes at state s. In a given state s_t, the agent selects the action based on the optimal detection policy given by

$$\pi^*(s) = \arg\max_{a \in \mathcal{A}} Q^*(s, a). \tag{4}$$

The **reward** r_t helps the agent to learn an effective detection policy, and it is offered as feedback (i.e., positive/negative) for an action a_t taken in state s_t. The reward r_t for an action a_t under state s_t is computed based on the ground truth values of BSMs. Concretely, a positive reward is given to the agent for correctly detecting an attack, i.e., true positive (TP) or a normal state, i.e., true negative (TN); otherwise, a negative reward is given to the agent for incorrect identification of a normal state as an attack, i.e., false positive (FP) or an attack as a normal state, i.e., false negative (FN). In safety-critical IoV scenarios, the correct identification of misbehavior is vital in order to mitigate potential hazardous situations. The agent is hence penalized more for FN actions than for FPs. The reward function can be expressed by

$$r(s, a) = \begin{cases} A & \text{if the action is a TP,} \\ B & \text{if the action is a TN,} \\ -C & \text{if the action is an FP,} \\ -D & \text{if the action is an FN,} \end{cases} \tag{5}$$

where $A, B, C, D > 0$, with $A > B$ and $D > C$.

3) Detection performance results: We hereby evaluate the performance of the RL-based detection approach introduced before, for the detection of the DoS attack variants (as described in subsection III-A), originated from rogue insiders.

Dataset: The VeReMi dataset [12], used in this work, models two road traffic densities under each attack type scenario: high-density (37.03 vehicles/km^2) and low-density (16.36 vehicles/km^2). For each attack scenario, a log file is generated for each participating vehicle, which contains BSM data transmitted by other vehicles over its entire trajectory. In addition, each attack scenario contains a ground truth file to record the observed behavior of all participating vehicles. BSMs contain position, speed, acceleration and heading angle information related to a vehicle's mobility.

For all simulations, the VeReMi dataset maintains the proportion between malicious and genuine vehicles between 30% and 70%. In our experiments, the RL model was trained using the high-density dataset under each attack scenario in order to learn and detect attack patterns more frequently; whereas, the low-density dataset was used to test the ability of the RL model in detecting attacks when anomalous data are less frequent within the BSM streams.

Results: In our experiments, the detection performance of the RL algorithm was evaluated based on the most commonly used metrics, i.e., accuracy, precision, recall and F1 score. A higher F1 score is considered as the indicator of a better detection performance, as it corresponds to the harmonic mean of precision and recall. For rewards, the values of $5, 1, 1, 5$ are assigned for $A, B, -C, -D$, respectively.

Table I demonstrates the detection performance per DoS variant over the test dataset. Results show that for all five DoS variants, RL-based detection is performed highly effectively with over 99.5% of F1 score. Such high levels of F1 scores demonstrate the superior capability in distinguishing misbehaviors from the genuine behavior with very low number of false alarms (i.e., FPs and FNs). This is further confirmed by the 100% recall values for the first four DoS variants and the recall value of 99.4% for DoS disruptive Sybil attack. In

TABLE I: Detection performance per DoS attack variant over the test dataset

Attack type	Accuracy	Precision	Recall	F1
DoS	0.9999	0.9999	1.0000	0.9999
DoS Random	0.9997	0.9996	1.0000	0.9998
DoS Disruptive	0.9991	0.9984	1.0000	0.9992
DoS Random Sybil	1.0000	1.0000	1.0000	1.0000
DoS Disruptive Sybil	0.9972	0.9998	0.9940	0.9970

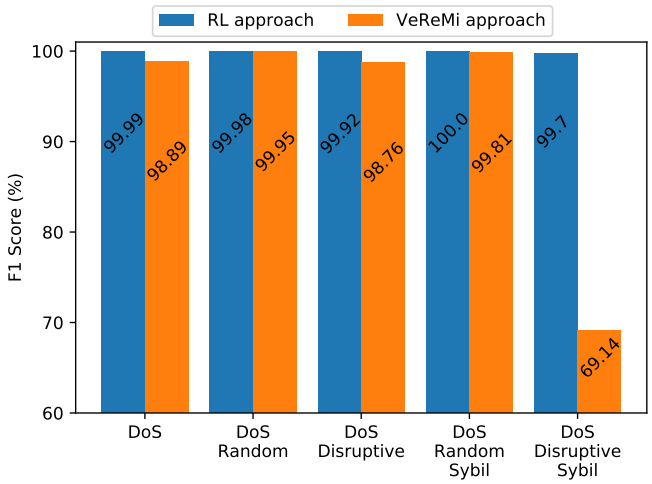

Fig. 5: Performance comparison of the proposed RL approach against the VeReMi approach (proposed in [12]) in terms of F1 score.

addition, Table I registers very high accuracy rates, i.e., close to 100%, for misbehavior detection which in fact is essential in safety-critical IoV scenarios.

Furthermore, we evaluate the detection performance of the proposed RL-based approach, using the VeReMi approach described in [12] as benchmark. The visual assessment of the comparative bars in Fig. 5 suggests that the RL-based detection clearly outperforms the benchmark approach in terms of the measured F1 scores for all DoS variants. It is worth noting that the detection performance of DoS disruptive Sybil attack using the benchmark scheme is considerably low with 69.14% of F1 score; whereas the RL-based approach achieves a significantly higher 99.7% of F1 score. The degraded performance of the benchmark scheme is attributed to the higher number of FN alarms, due to the plausible stale content being replayed with concealed identities.

IV. Conclusion and Path Forward

Vulnerabilities in large-scale multi-domain automotive environments give rise to a wide range of DoS attack variants, which may result in compromised functional components and end-to-end security issues against network users. This paper sheds light on security threats originated from DoS attackers against functional elements of an IoV-tailored authentication mechanism, and provides useful design insights to improve resilience based on appropriate cuckoo filter configurations. In addition, a data-driven methodology based on RL is introduced to accurately detect misbehaving vehicles launching DoS attacks in rapidly changing RAN environments. Our scheme yields superior performance in terms of F1 score compared to a benchmark approach.

In the path forward, we aim to integrate our RL-based DoS detection method as part of a security closed-loop demonstration in the context of INSPIRE-5Gplus project [15]. In particular, the closed-loop instantiation of our data-driven DoS detector will involve several architectural components of a zero-touch security management framework, namely data collection, security analytics and decision engine. Data collection will perform the fusion of IoV network traces that are streamed from the data plane using virtual machines which emulate the representation of vehicles within the RAN. Security analytics will sequentially analyze the incoming streaming vehicular data reports based on mobility patterns, to instruct the RL algorithm for the detection of DoS attacks. Finally, the decision engine, upon detection of DoS attacks, provides the verdict to a security orchestrator to apply a pre-determined security policy, e.g., malicious traffic to be isolated, dropped, or blocked.

Acknowledgment

This work has been partly supported by the H2020-INSPIRE-5Gplus project (Grant agreement No. 871808), by the CHIST-ERA-17-BDSI-003 FIREMAN project funded by the Spanish National Foundation (Grant PCI2019-103780), by the H2020-CARAMEL project (Grant agreement No. 833611), and by the Grant PID2020-112675RB-C43 funded by MCIN/AEI/10.13039/501100011033.

References

[1] O. Kaiwartya, A. H. Abdullah, Y. Cao, A. Altameem, M. Prasad, C. Lin, and X. Liu, "Internet of Vehicles: Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects," *IEEE Access*, vol. 4, pp. 5356–5373, 2016.

[2] N. Trkulja, D. Starobinski, and R. A. Berry, "Denial-of-Service Attacks on C-V2X Networks," 2020. [Online]. Available: arXiv:2010.13725.

[3] A. Ghosal and M. Conti, "Security issues and challenges in V2X: A Survey," *Computer Networks*, vol. 169, p. 107093, 2020.

[4] F. Sakiz and S. Sen, "A survey of attacks and detection mechanisms on intelligent transportation systems: VANETs and IoV," *Ad Hoc Networks*, vol. 61, pp. 33 – 50, 2017.

[5] C. Kalalas and J. Alonso-Zarate, "Lightweight and space-efficient vehicle authentication based on cuckoo filter," in *2020 IEEE 3rd 5G World Forum (5GWF)*, 2020, pp. 139–144.

[6] R. W. van der Heijden, S. Dietzel, T. Leinmüller, and F. Kargl, "Survey on Misbehavior Detection in Cooperative Intelligent Transportation Systems," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 779–811, 2019.

[7] P. Bagga, A. K. Das, M. Wazid, J. J. P. C. Rodrigues, and Y. Park, "Authentication Protocols in Internet of Vehicles: Taxonomy, Analysis, and Challenges," *IEEE Access*, vol. 8, pp. 54 314–54 344, 2020.

[8] B. Fan, D. G. Andersen, M. Kaminsky, and M. D. Mitzenmacher, "Cuckoo Filter: Practically Better Than Bloom," in *Proceedings of the 10th ACM International on Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 75–88. [Online]. Available: https://doi.org/10.1145/2674005.2674994

[9] ETSI TS 133 501 v15.4.0, "5G; Security architecture and procedures for 5G System ," May 2019.

[10] P. Reviriego and D. Larrabeiti, "Denial of Service Attack on Cuckoo Filter Based Networking Systems," *IEEE Communications Letters*, vol. 24, no. 7, pp. 1428–1432, 2020.

[11] J. R. Douceur, "The Sybil Attack," in *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, ser. IPTPS '01. Berlin, Heidelberg: Springer-Verlag, 2002, p. 251–260.

[12] J. Kamel, M. Wolf, R. W. van Der Heijden, A. Kaiser, P. Urien, and F. Kargl, "Veremi extension: A dataset for comparable evaluation of misbehavior detection in vanets," in *2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland, Jun 2020.

[13] C. Szepesvári, "Algorithms for reinforcement learning," *Synthesis lectures on artificial intelligence and machine learning*, vol. 4, no. 1, pp. 1–103, 2010.

[14] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3, pp. 279–292, 1992.

[15] INSPIRE5G-plus: Intelligent security and pervasive trust for 5G and beyond, "Deliverable 3.3: 5G security new breed of enablers," https://www.inspire-5gplus.eu/wp-content/uploads/2022/03/i5-d3.3_5g_security_new_breed_of_enablers_v1.0.pdf, 2022, [Online].