Trust Verification in Connected Vehicles Using Unsupervised Variational Autoencoder

Ramzi Boutahala[†], Hacène Fouchal[†], Marwane Ayaida[‡†], and Shiwen Mao[§]
[†]Université de Reims Champagne Ardenne, CReSTIC EA 3804, 51097 Reims, France
[‡]Univ. Polytechnique Hauts-de-France, CNRS, Univ. Lille, UMR 8520 - IEMN, F-59313 Valenciennes, France
[§]Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849-5201 USA
Email: ramzi.boutahala@univ-reims.fr, hacene.fouchal@univ-reims.fr, marwane.ayaida@uphf.fr, smao@ieee.org

***Abstract*—Cooperative Intelligent Transport Systems (C-ITS) are of great importance in our daily lives. They offer additional means for safer roads thanks to exchanged data between actors (i.e., vehicles and Road Side Units (RSU)). Signatures (that are computed using various Pseudonym Certificates (PC)) are included in all the sent messages. Each vehicle periodically sends application beacons (denoted by CAM (Cooperative Awareness Message)). The integration of the signature and certificate in each transmitted CAM could consume a considerable portion of the communication channel bandwidth. In this study, we propose a new lightweight authentication mechanism using an unsupervised variational autoencoder. Instead of exhaustive authentication, our approach allows vehicles to authenticate each other once and then send only unsigned CAMs based on the trust established during authentication. In order to check this trust level, we proposed to use an unsupervised deep learning mechanism, which continuously measures the variation of the neighbor's behavior. When this variation reaches an unacceptable level, the vehicle assumes that the sender may be compromised. As a result, it proceeds to the authentication of the sender. We have implemented these mechanisms over the OMNET++ network simulation environment. Our simulation study shows that the proposed approach reduces the overhead generated by the authentication algorithms by around 48.9%.**

***Index Terms*—Clustering, Security and privacy, Signature, Authentication, Variational autoencoder, Deep learning, C-ITS.**

I. Introduction

Cooperative Intelligent Transportation Systems (C-ITS) enable dynamic, real-time interaction between vehicles, drivers, and infrastructure, representing a significant advance towards greater road safety. Cooperative intelligent transportation systems have the potential to solve many problems including accidents, traffic congestion, and environmental pollution by alerting the driver about disturbing events. C-ITS provide services that improve driving through cooperation between road infrastructure, drivers, and vehicles. These services allow vehicles to communicate with each others by exchanging messages. Due to the sensitivity of these messages, they are susceptible to manipulation in several types of cyber attacks. Therefore, it is very important to ensure the integrity of these messages and authenticate them.

In the Europe, the European Telecommunications Standards Institute (ETSI) has proposed relevant protocols to ensure the safe exchange of specific messages. These protocols define policies for managing security certificates, signature and encryption algorithms, and the structure of secure cooperative awareness messages (CAMs). CAMs are a specific type of C-ITS messages that are sent periodically to share vehicle status information, such as GPS coordinates, heading, and speed. Vehicles broadcast their CAMs, which consist of payload data, a certificate, and a signature. Each vehicle must sign its CAMs using its certificates. However, the integration of the signature and certificate into each transmitted CAM will have a significant impact on the load and bandwidth consumption of the communication channel.

In this paper, we propose a novel lightweight authentication mechanism using an unsupervised variational autoencoder. This approach aims to reduce the amount of bandwidth consumed by the signatures and certificates included in CAMs that are exchanged between vehicles. This approach has several advantages: it avoids the risk of high communication channel overhead, and reduces the CAM latency while maintaining security. Instead of exhaustive authentication, our approach allows vehicles to authenticate each other once and then send only unsigned CAMs based on the trust established after the authentication phase. Then, each vehicle checks whether the behavior of its neighboring vehicles is normal to ensure that data is not manipulated and to maintain trust. For this purpose, we proposed an unsupervised variational autoencoder mechanism, which continuously measures the variation of neighbor's behavior. When this variation reaches an unacceptable level, the vehicle assumes that the sender may be compromised and relaunch the authentication process. We have implemented our architecture on the Artery framework, using the OMNET++ network simulator and the SUMO road traffic simulator, in order to demonstrate the efficiency of our proposal. Our simulation study shows that the proposed approach reduces the overhead generated by the authentication algorithms by round 48.9%.

The remainder of this paper is structured as follows. Section II introduces the related work. Section III describes the proposed architecture within this study. Section IV outlines the simulation tests, which includes the simulation settings and the result analysis, before concluding this work in Section V.

II. Related Work

This section presents some important works about the security for connected vehicles.

In the context of connected vehicles, the authors of [3] proposed an authentication protocol that did not use a central authority. In order to authenticate vehicles, this protocol uses only the message signatures to reduce the authentication time and overhead. However, in case of an attack, the revocation list may be expanded quickly, since each vehicle uses multiple pseudonyms that need to be revoked as a whole if the vehicle is compromised.

The study in [4] has proposed an alternative authentication method, called Trust-Based Authentication Technique (TBAT). TBAT uses trust degrees to choose the most convenient cluster-heads. It ensures that all communications are securely signed and encrypted by the sender using the public-private key cryptography mechanism. However, in an open cooperative environment, where messages need to be exchanged without any delays or encryption, this technique could drastically increase the latency.

On the other hand, an authentication scheme based on signature that preserves privacy was proposed in [5]. To address the issue of managing certificates, the authors proposed to divide the network into different domains. In addition, they used a Hash Message Authentication Code (HMAC) to reduce the time consumed when using a certificate revocation list. This approach reduces both the time to verify the message integrity, and the number of invalid messages, thereby lowering the cost and overhead of authentication.

Further, the authors in [6] proposed an authentication mechanism that used a clustering algorithm to overcome the challenges of cryptography usage in VANETs due to frequent changes in the vehicles' positions. The main objective here is to create stable clusters and be trustworthy in all of the network. They have also suggested ways to detect malicious vehicles, and the cluster heads were chosen from the most trusted vehicles. The routing efficiency was guaranteed by the network stability and these trusted cluster heads. For enhanced network security, few vehicles are selected to monitor their neighbouring vehicles. In this work also, the signature and asymmetric cryptography were used. Therefore, this work suffers from the same drawbacks as the work in [4].

The authors in [8] put forward two proposals using cryptography to ensure privacy. The first one aims to fight against eavesdropping using zone-encryption, and it was combined with a scheme that ensures anonymous authentication to permit only non-malicious vehicles to send messages. The primary disadvantage of this method is that it introduces an overhead of 224 bytes for cryptography within each message, thereby consuming large bandwidth and increasing latency. The second proposal was better adapted to the vehicular environment, enabling vehicles to distribute keys among themselves. This proposal uses compact group signatures, which allows for reduced security overhead in bandwidth with a minor impact on storage cost, while ensuring a high level of privacy. However, in case of an attack, the revocation process could be complex as it does not guarantee non-repudiation.

In [9], authors introduced a new authentication mechanism called Certificate Less Aggregate scheme based on Traceable Ring Signature (CLA-TRS). This innovative technique uses a ring signature in conjunction with bi-linear matching on an elliptic curve. Thus, it ensures privacy, while reducing the time for signature verification. Meanwhile, authors in [10] proposed an authentication message approach that merges identity-based signatures with ring signatures. The low efficiency of this approach is due to the time that is consumed in message signing and their verification. Finally, the authors in [11] also used the ring signature and bi-linear matching in their approach. In addition, they incorporated batch signatures to reduce the verification overhead. However, it is still not sufficient in terms of single signature and verification.

All the aforementioned approaches attempted to reduce the security fingerprint by proposing lightweight securing mechanisms. Yet, none of these methods has suggested an intermittent activation or switch off of the security mechanism as a means to save bandwidth.

Machine learning approaches have been intensively studied in recent years. An effective anomaly detection concept is required to represent the anomalous behavior of processes. Several works have already proposed LSTM methods to detect various types of anomalies. For instance, authors in [14] used a recurrent variational autoencoder to model breathing and Kullback–Leibler (KL) divergence to compare the output with the input, acting as a sleep apnea detector. This model detects sleep apnea using the amplitude of a breathing signal and a threshold. On the other hand, clustering was utilized in [17] to choose a single layer of sparsely placed promiscuous monitors. These monitors leverage statistical anomaly detection to evaluate any routing misbehavior.

III. Our Proposed Approach

In the C-ITS context, each vehicle periodically broadcasts, within a 100 ms window, its CAM message to neighbors to indicate its current status. The CAM messages are mainly composed of a data payload, a certificate, and a signature hash. It is important to notice that the size of the security overhead is three times larger than the data payload. At the same time, the broadcast CAMs significantly increase the load of the communication channel and consume considerable bandwidth. In [15] and [16], we proposed two approaches that aim to reduce the security information based on the trust established during the first authentication phase within a cluster. However, to ensure that vehicle communications are highly secured, we propose, in the current work to combine the aforementioned approaches with an unsupervised deep learning mechanism to detect anomalies in vehicle behavior, and thus to verify the trust established during the authentication phase. The procedure of the overall approach will be described in the remainder of this section.

A. Cluster Dynamics in Trust-Based Approach

1) Cluster Construction: In our study, we assume that a set of vehicles drive near each other while broadcasting signed CAMs. A random vehicle then decides to initiate an authentication phase to form a trusted cluster among them. Therefore, it broadcasts a CAM with a request of certificate, containing a hash ID of all its neighbors as shown in Fig. 1. Upon receiving this CAM, the vehicles participate in the authentication process by sending their signed CAMs. During this phase, each vehicle maintains a list of its authenticated neighbors, and it will react as a cluster head to make sure that all its neighbors are authenticated. This allows us to manage the authentication phase in a dynamic environment such as vehicular networks.

2) Cluster Update: When vehicles form clusters, it means that they trust each other and the communication between them is done using unsigned CAM messages. In order to ensure that no data manipulation can be done and to guarantee a trusted environment among all neighbors during this phase, the vehicles periodically check the behavior of their neighbors using a deep learning model to detect anomalies. This model will be detailed later in this section. If an anomaly is detected, the vehicles send a certificate request to re-authenticate the vehicle. If a new vehicle enters the cluster, the first vehicle that detects it requests its certificate. If a vehicle leaves the cluster, the vehicles update their list of authenticated neighbors by removing it from the list.

B. Trust Management

In this section, we describe how our approach maintains a high level of security, while reducing the security overhead. As mentioned in Section III-A, the dynamics of a cluster is related to managing the trust of each vehicle towards all other cluster members. During this phase, each vehicle frequently checks the behaviors and the trajectories of its neighbors. To do so, each vehicle uses the content of the CAMs that are received from its neighbors as parameters to detect unsound CAMs. The key idea here is that the vehicle maintains its trust in the sender and still accepts its unsigned CAMs, when the vehicle considers that the CAMs sender is consistent.

In order to measure the soundness of a vehicle's behavior, we introduce a recognition model. This model is used as an approximation process to a more complex estimation that cannot be easily computed. Instead of relying on a predefined mathematical expression to define the thresholds for detecting unsound CAMs, as we have proposed in [16], the model in this approach is trained with calibrated data to learn how to determine these parameters and to achieve higher accuracy. We therefore propose an *unsupervised deep learning mechanism*, which continuously measures the variation of the neighbor's behavior. When this deviation reaches an unacceptable level, the vehicle assumes that the sender may be compromised, and then requests its certificate to authenticate it once again. This mechanism is detailed in the following.

1) Background of Variational Autoencoder: An autoencoder is an unsupervised neural network that attempts to learn the optimum encoding-decoding technique from data [13], while a Variational autoencoder (VAE) is a probabilistic model which combines Bayesian inference with the autoencoder framework [18]. VAE models the relationship between the observed variables x, latent random variables z, and a set of parameters, represented by θ. These elements are combined in a probability model, where the prior on z is represented by $p(z)$, and $p_\theta(x|z)$ is the probability of an observation. This is given by the following equation:

$$p(x) = \int p_\theta(x|z)p(z)dz. \tag{1}$$

However, the integral operation $p_\theta(x)$ is computationally intractable. Therefore, the VAE uses a variational approximation $q_\phi(z|x)$ in stead of the true posterior $p_\theta(z|x)$. Here, $q_\phi(z|x)$ with parameters ϕ serves as the encoder, and $p_\theta(x|z)$ with parameters θ serves as the decoder. According to Jensen's inequality, this VAE model is able to find optimal values for the parameter sets ϕ and θ by maximizing a lower bound on the log-likelihood given by [19]:

$$\max \mathcal{L} = -D_{KL}(q_\phi(z|x)||p(z)) + \mathbb{E}_{z\sim q_\phi(z|x)}[p_\theta(x|z)], \tag{2}$$

where D_{KL} stands for the Kullback-Leibler divergence, which regularizes the latent z variables, and the second term is the autoencoder. VAE uses a technique called re-parameterization to simplify the learning process. It calculates the latent vector z from the mean vector $\mu_\phi(x)$ and the variance vector $\sigma^2_\phi(x)$, as follows:

$$z = \mu_\phi(x) + \sigma_\phi(x)\varepsilon, \tag{3}$$

where ε follows the standard Gaussian distribution $\mathcal{N}(0, 1)$, adding a randomness factor to the latent space.

The lower bound of the log-likelihood, denoted by $\mathcal{L}$, can then be approximated as:

$$\mathcal{L} \approx 0.5 \times \sum_j (1 + \log((\sigma_j^2)(x)) - (\mu_j^2)(x) - (\sigma_j^2)(x)) + \frac{1}{M} \times \sum_l \log(p_\theta(x|z_l)), \tag{4}$$

where M is the total number of samples in z, and J is the size of z. The VAE loss function is often formed from the mean square error (MSE), which serves as the reconstruction loss, and the KL divergence to evaluate performance, as:

$$\text{Loss} = \text{MSE} + \text{KL}. \tag{5}$$

MSE is calculated as follows:

$$\text{MSE} = \frac{1}{N}\sum(x - x')^2, \tag{6}$$

where x, x', and N represent the original input, the reconstructed data, and the total number of samples, respectively.

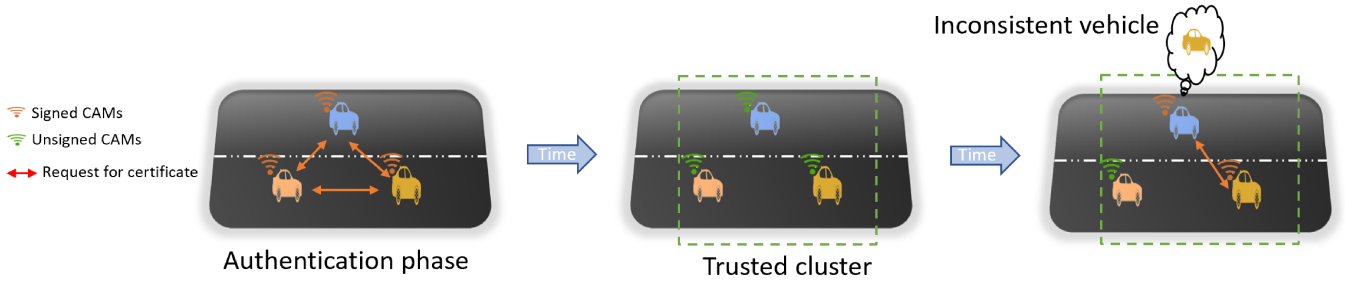

Fig. 1. The procedure for cluster construction.

2) Offline Variational Autoencoder Training Phase: The variational autoencoder is based on unsupervised learning. This means that there is no need to collect labeled data for normal and abnormal vehicle behavior, which is very interesting in our case since labeled data could be complex and expensive to obtain. In the context of C-ITS, we have generated a data-set of 40 million CAM messages generated by simulating a thousand of vehicles driving on the A4 highway between Reims and Paris in France. A vehicle broadcasts a CAM every 100 ms. We have then pre-processed these CAMs to extract 6 important parameters that allow us to observe vehicle behaviors. In this part of the pre-processing phase, we have used Vincenty's formula to compute the three-dimensional geodesic distance (latitude, longitude, and altitude) between the steps taken by each vehicle indicated in its CAM. Our input dataset is structured with the following variables: Heading, Speed, YawRate (rate of rotation compared to the vertical axis), curvature, longitudinal acceleration, and distance.

We have considered our data input in a time window of 10 steps, which is equivalent to 1s, to represent the data from 10 consecutive CAMs. These time windows will be processed by a long short-term memory network (LSTM). First, the LSTM encodes the vehicle data sequence in the time window. Then, the generated outputs are used to derive estimations for the mean vector $\mu_\phi(x)$ and the variance vector $\sigma^2_\phi(x)$ using two linear modules. At the end, the sampled z is fed to another LSTM network to decode the estimated mean and variance vectors. Finally, we obtain a reconstruction of our input data for the same time window. Using reconstruction loss (MSE) and KL divergence, VAE trains to produce similar reconstructions. Once the reconstructed input data is obtained, we calculate the maximum mean absolute error loss value on the training samples. This will be the threshold to determine whether the vehicle CAMs' data in this time window are coherent or not.

3) Processing Phase: During the dynamic clustering phase, vehicles periodically activate the VAE model in a time slot to check the CAM messages received from their neighbors, as shown in Fig. 2. First, each vehicle processes the last 10 CAM messages that are received from each neighbor. This pre-processing step is performed in the same way as in the learning phase. Then, the vehicle reconstructs the time window input and compares the reconstruction loss with the threshold set in the learning phase. If the reconstruction loss for a sample exceeds this threshold, the vehicle labels these samples as inconsistent. After detecting an inconsistency, the vehicle sends a request of certificate to the suspected vehicle. If there is no answer, the vehicles leave the cluster and start to build a new cluster.

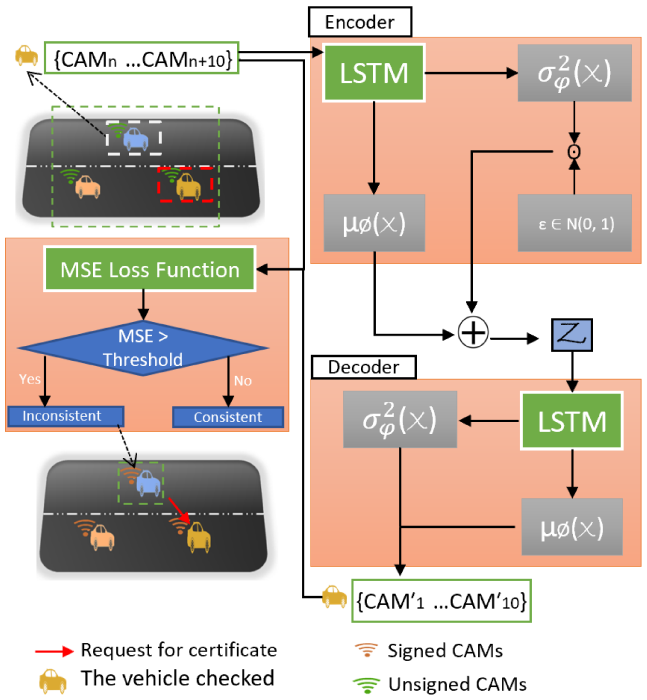

Fig. 2. Architecture of the proposed variational autoencoder-based process for unsupervised inconsistency detection.

IV. Performance Evaluation

To ensure an accurate evaluation of the proposed approach in a realistic simulation environment, we used several simulators and frameworks in this study. Firstly, we used the OMNET++ network simulator [21], which is developed in the form of independent modules, which can be combined

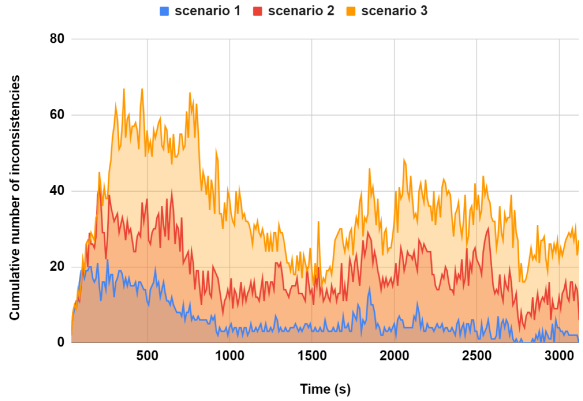

Fig. 3. The cumulative number of inconsistencies every 10s in the three scenarios.

to form complex systems. Designed to model communication systems, networks, multiprocessors and other distributed systems. Omnet++ is based on the C++ language and also has its language NED (Network Description language) to define the network topology. It is based on discrete event scheduling rather than continuous-time scheduling. We also used Artery framework [20], a simulation framework dedicated to V2X (Vehicles to Everything) communication. Artery is based on Omnet++, and takes advantage of its modularity and ability to simulate discrete events. It is an open-source simulation framework, that allows for real-time communication of road traffic data throughout the simulation process. We used a traffic simulator, Simulation of Urban Mobility (SUMO), which will be used to generate the road traffic mobility [22]

Artery provides each vehicle with the standard C-ITS protocol stack with its security mechanisms. To exchange CAMs over networks, Artery uses the IEEE 802.11p physical layer implemented by the framework VEINS. In the first scenario, we introduced a batch of 10 vehicles all at once into the highway. Moving on to the second and third scenarios, we increased the complexity by launching 20 and 30 vehicles, respectively. Please note that we have configured the SUMO module to make vehicles moving using a sigma value of 0.5, which indicates a moderate level of random variation in each vehicle's speed.

We configured the vehicles to check their neighbors every 5 seconds, and then we captured CAM inconsistencies during the simulations. Fig. 3 shows the cumulative number of inconsistencies every 10 seconds of the three scenarios. We find that our VAE considers several cases of anomalies as CAMs inconsistencies as time elapses. This is initially due to variations in the speed of vehicles as they enter the highway and adjust their speeds to the speed limit. We also noticed that the number of inconsistencies decreases quickly in scenarios 1 and 2 compared to scenarios 3. This is due to the fact that in scenario 1, vehicles have the ability to distance themselves more quickly compared to scenario 2 and 3, due to a lower total number of vehicles.

Table I presents the percentage of inconsistencies detected

TABLE I
THE PERCENTAGE OF INCONSISTENCIES DETECTED BY THE VEHICLES AND THE NUMBER OF INCONSISTENT EVENTS COMPARED TO THE NORMAL CAM

S	Real Anomaly		Anomaly Detected		Sound Cams	
	Number	%	Number	%	Number	%
1	1940	0.41%	2187	0.46%	470227	%99.59
2	6199	0.42%	7741	0.51%	1495743	%99.58
3	12018	0.45%	15531	0.58%	2662441	%99.55

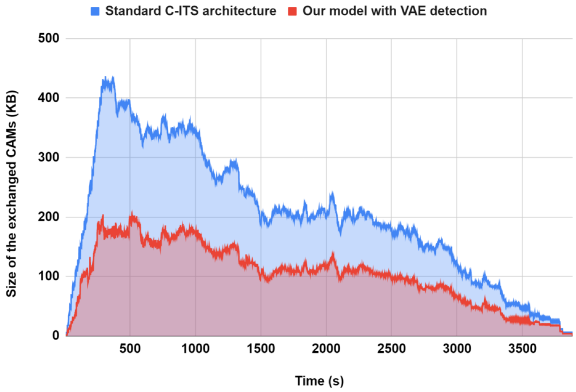

Fig. 4. The size of the data exchanged in Scenario 3 every 1s.

by the vehicles and the number of inconsistent events compared to the total number of CAM messages exchanged during the simulation. In all scenarios, more than 99.5% of the CAM messages were consistent. Our model was able to detect fewer anomalies in vehicle behavior. These results demonstrate the effectiveness of our model in understanding and detecting real-time anomalies in the CAM data exchanged between vehicles. Data never seen during training were detected as inconsistencies. This observation can be explained by the use of the MAE, which measures the discrepancy between the original data and the reconstructed data; a large difference means an anomaly.

During the simulation, we recorded the number of signed and unsigned CAMs received by each vehicle. Then we calculated the size of these messages exchanged every second. According to the C-ITS standard, the signed CAM size is around 300 bytes, while the unsigned CAM size is around 100 bytes. Fig. 4 shows a comparison of the size of CAMs exchanged during the simulation of our proposal and the standard C-ITS architecture. In fact, our proposed scheme is able to reduce the volume by 48.9%, compared to the standard C-ITS protocol. This percentage is quite interesting and will contribute to effectively reducing the consumed bandwidth. This is due to our strategy of reducing the security information by using the trust cluster, which is monitored by our VAE for vehicle behavior to detect inconsistent CAMS.

V. CONCLUSIONS

In this paper, we proposed a solution to avoid the risk of communication channel overload in C-ITS, which consists of

reducing the security information and allowing vehicles to authenticate each other using a trust cluster-based strategy. This strategy allows the vehicle to authenticate only once compared to usual standards, which use an exhaustive strategy. To maintain a high level of security between vehicles, we proposed a variational autoencoder to detect anomalies in vehicle behavior. We evaluated our proposed approach on three scenarios and have then compared it to the standard C-ITS scheme. Our approach reduced the amount of exchanged messages by 48.9%, which plays an important role in reducing the communication load. Our approach is a robust solution that provides a high level of security through unsupervised real-time detection to exploit these standards in a selective protocol. For future work, we shall evaluate the robustness of our proposal in respect to various types of cyber attacks. ”

ACKNOWLEDGMENT

This work was supported in part by EC Grant No. 2018-FR-TM-0097-S from the INEA Agency for the InDiD project. The statements made herein are solely the responsibility of the authors. S. Mao’s work is supported in part by the NSF under Grant ECCS-1923717 and CNS-2107190.

REFERENCES

[1] International Energy Agency, “How Many Cars Will Be on the Planet in the Future?” [online] Available: http://www.iea.org/aboutus/faqs/transport/ (accessed on Nov. 15, 2022)

[2] World Health Organization (WHO), “Global Status Report on Road Safety 2013,” *WHO Technical Report*, Geneva, Switzerland, 2013.

[3] X. Lin and X. Li, “Achieving efficient cooperative message authentication in vehicular ad hoc networks,” *IEEE Transactions on Vehicular Technology*, vol.62, no.7, pp.3339–3348, Sept. 2013.

[4] R. Sugumar, A. Rengarajan, and C. Jayakumar, “Trust based authentication technique for cluster based vehicular ad hoc networks (VANET),” *Springer Wireless Networks*, vol.24, pp.373–382, Feb. 2018.

[5] X. Zhu, S. Jiang, L. Wang, and H. Li, “Efficient privacy-preserving authentication for vehicular ad hoc networks,” *IEEE Transactions on Vehicular Technology*, vol.63, no.2, pp.907–919, Feb. 2013.

[6] F. Mirsadeghi, M. K. Rafsanjani, and B. B. Gupta, “A trust infrastructure based authentication method for clustered vehicular ad hoc networks,” *Springer Peer-to-Peer Networking and Applications*, vol.14, pp.2537–2553, July 2021.

[7] The European Telecommunications Standards Institute (ETSI), “E.N. 302 637-2 v1. 3.1-intelligent transport systems (its); Vehicular communications; Basic set of applications; part 2: Specification of cooperative awareness basic service,” European Standard, Sept. 2014.

[8] J. Camenisch, M. Drijvers, A. Lehmann, G. Neven, and P. Towa, “Zone encryption with anonymous authentication for V2V communication,” in *Proc. 2020 IEEE European Symposium on Security and Privacy (EuroS&P)*, Virtual Conference, Sept. 2020, pp.405–424.

[9] S. Bouakkaz and F. Semchedine, “New efficient certificateless scheme-based conditional privacy preservation authentication for applications in VANET,” *Elsevier Vehicular Communications*, vol.34, pp.100414, Apr. 2022,

[10] J. Li, Y. Liu, Z. Zhang, B. Li, H, Liu, and J. Cheng, “Efficient ID-based message authentication with enhanced privacy in wireless ad-hoc networks,” in *Proc. 2018 International Conference on Computing, Networking and Communications (ICNC)*, Maui, HI, Mar. 2018, pp.322–326.

[11] F. Liu and Q. Wang, “IBRS: An efficient identity-based batch verification scheme for VANETs based on ring signature,” in *Proc. 2019 IEEE Vehicular Networking Conference (VNC)*, Los Angeles, CA, Dec. 2019, pp.1–8.

[12] H.D. Nguyen, K.P. Tran, S. Thomassey, and M. Hamad, “Forecasting and anomaly detection approaches using LSTM and LSTM autoencoder techniques with the applications in supply chain management,” *Elsevier International Journal of Information Management*, vol.57, pp.102282, Apr. 2021.

[13] D.P. Kingma, and M. Welling, “Auto-encoding variational bayes,” arXiv:1312.6114, May 2014. [Online]. Available: https://arxiv.org/abs/1312.6114.

[14] C. Yang, X. Wang, and S. Mao, “Unsupervised detection of apnea using commodity RFID tags with a recurrent variational autoencoder,” *IEEE Access Journal*, vol.7, pp.67526–67538, May 2019.

[15] R. Boutahala, M. Ayaida, and H. Fouchal, “Reducing security overhead in the context of connected Vehicles,” in *Proc. IEEE GLOBECOM 2022*, Rio de Janeiro, Brazil, Dec. 2022, pp. 1–6.

[16] R. Boutahala, H. Fouchal, and M. Ayaida, “An efficient approach to reduce the security messages overload on C-ITS,” in *Proc. IEEE ICC 2022*, Seoul, South Korea, May 2022, pp.1500–1505.

[17] Y. Huang and W. Lee, “A cooperative intrusion detection system for ad hoc networks,” in *Proc. 1st ACM workshop on Security of Ad Hoc and Sensor Networks*, Fairfax, VA, Oct. 2003, pp.135–147.

[18] Y. Guo, W. Liao, Q. Wang, et al., “Multidimensional time series anomaly detection: A GRU-based Gaussian mixture variational autoencoder approach,” in *Proc. Asian Conference on Machine Learning*, Beijing, China, Nov. 2018, pp.97–112.

[19] D.P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, Dec. 2013, [online] Available: https://arxiv.org/abs/1312.6114.

[20] R. Riebl, H. Günther, C. Facchi, and L. Wolf, “Artery: Extending veins for VANET applications,” in *Proc. 2015 International Conference on Models and Technologies for Intelligent Transportation Systems* (MT-ITS), Budapest, Hungary, June 2015, pp.450–456.

[21] A. Varga, “The omnet++ discrete event simulation system,” in *Proc. the European Simulation Multiconference*, Prague, Czech Republic, June 2001, pp.319–324, 2001.

[22] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, “Sumo (simulation of urban mobility) - An open-source traffic simulation,” in *Proc. 4th Middle East Symposium on Simulation and Modelling*, Dubai, UAE, Sept. 2002, pp.183–187.