Edge Based Lightweight Authentication Architecture Using Deep Learning for Vehicular Networks

Hyunhee Park*
Department of Information and Communication, Myongji University, South Korea
hhpark@mju.ac.kr

Abstract

When vehicles are connected to the Internet through vehicle-to-everything (V2X) systems, they are exposed to diverse attacks and threats through the network connections. Vehicle-hacking attacks in the road can significantly affect driver safety. However, it is difficult to detect hacking attacks because vehicles not only have high mobility and unreliable link conditions, but they also use broadcast-based wireless communication. To this end, V2X systems need a simple but a powerful authentication procedure on the road. Therefore, this paper proposes an edge based lightweight authentication architecture using a deep learning algorithm for road safety applications in vehicle networks. The proposed lightweight authentication architecture enables vehicles that are physically separated to form a vehicular cloud in which vehicle-to-vehicle communications can be secured. In addition, an edge-based cloud data center performs deep learning algorithms to detect car hacking attempts, and then delivers the detection results to a vehicular cloud. Extensive simulations demonstrate that the proposed authentication architecture significantly enhanced the security level. The proposed authentication architecture has 94.51 to 99.8% F1-score results depending on the number of vehicles in the intrusion detection system using control area network traffic.

Keywords: Lightweight authentication, Controller area network, Intrusion detection system, Vehicular network, Deep learning

1 Introduction

Connected vehicles can access the Internet and various other sensors. They can also exchange signals and information and interact with other vehicles or objects while sensing the surrounding physical environment [1]. In addition, autonomous cars, which can drive without human control, are under active development because they are expected to reduce transportation costs and improve convenience and safety. Because 5G wireless technology enables data streaming from cloud storage in real time, various infotainment services can be provided to connected and autonomous cars [2].

Vehicle industry has been affected by advancements in information and wireless communication. Recently, vehicles are no longer only driving machines, but they are also expected to enhance driver convenience by interfacing with telecommunication systems. According to a Gartner survey, one out of five vehicles will be connected cars equipped with wireless communication by 2020 [3–4]. As shown in Figure 1, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications are performed via direct communication. Meanwhile, between the vehicle and the base station and the roadside unit (RSU) and the base station, the traffic information can be transmitted in real time using mobile communication, as well as traffic information several kilometers ahead in real time.

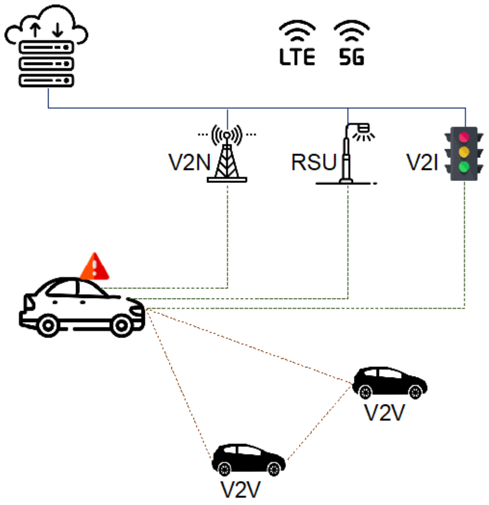


Figure 1. Cellular vehicle-to-everything

Several IT companies worldwide are developing platforms to provide infotainment services to connected cars. It is a common phenomenon for vehicles to be densely distributed in urban environments, especially when traffic congestion occurs. Therefore, vehicular cloud (VC) system, which utilizes communication resources from multiple vehicles based on cooperative V2V communication, has been actively studied [5–7]. Accordingly, vehicles with limited resources can gain access to services by using resources from other vehicles in the same VC system, where the latter vehicles can provide resources that are not currently in use, to the former vehicles. However, V2V wireless communication or infotainment services based on the Internet are exposed to various attacks and threats, such as denial-of-service (DoS) attacks to vehicular security [8]. A security attack on vehicles can lead to service failure and negatively affect driver safety. However, it is difficult for vehicles to detect attacks in real time because not only do they have high mobility and unreliable link conditions, but they also use broadcast-based wireless communication. Furthermore, vehicles in existing vehicular ad hoc networks (VANETs) can communicate only with neighboring vehicles within a fixed distance when wireless communication is available. Therefore, a secure vehicular authentication architecture is required to form

*Corresponding Author: Hyunhee Park; E-mail: hhpark@mju.ac.kr
DOI: 10.53106/160792642022012301020

cloud-based vehicular networks between vehicles that are physically separated and ensure safe communication among member vehicles. To satisfy these requirements, we propose an edge-based lightweight authentication (ELA) architecture for vehicular networks. The key contributions of this paper are as follows:

1) In the proposed ELA architecture, an edge data center for vehicle (EDC_V) publishes and distributes certificate authorities (CAs) for member vehicles of a vehicular cloud, as well as encryption keys for the safe exchange of messages among vehicles.

2) In addition, EDC_V performs deep learning to predict abnormal behaviors in vehicles. The prediction results are transferred to RSUs that manage the vehicular cloud and then quickly provided to vehicles. If a vehicle with the intent of hacking other vehicles is detected, the EDC_V stops updating certificate information for that vehicle to prevent it from being activated in the vehicular cloud.

The remainder of this paper is organized as follows: Section 2 describes the related work for vehicle authentication and security attacks in vehicular network environments, and Section 3 presents proposed edge lightweight authentication architecture. Section 4 defines the proposed Deep learning for abnormal patterns, and Section 5 provides extensive simulation parameters and simulation results. Conclusion and future work are presented in Section 6.

2 Related work

Several studies on vehicle authentication and message encryption have been conducted to ensure the safety of V2V communications in vehicular networks [9]. The IEEE 1609 group is working on standardizing wireless access in vehicular environments (WAVE) technologies for V2V communication [10–11]. Generally, public-key-based configurations are used for vehicle authentication and message encryption in vehicular networks. Such a public-key-based configuration assumes the presence of a reliable CA, which issues and manages certificates for each vehicle.

Many studies have focused on the creation and distribution of encryption keys for encrypting and decoding messages [12–14]. Algorithms for digital signature and message encryption can be implemented through either software or hardware, but processing delays can occur based on the performance of the hardware. A public-key-based configuration broadcasts the certificate of a vehicle along with a message whenever a vehicle wishes to send a message. If the certificate of a vehicle is sent with a message, additional overhead can be incurred for message sending, and extra time is required to verify the certificate. Furthermore, broadcasting a vehicle certificate with a message may allow vehicles other than the intended recipient to receive the certificate, which can create security issues.

One of the studies suggested the introduction of a certificate revocation list (CRL) to identify abnormal or malicious vehicles with the intent of performing security attacks in a vehicular network environment [15–16]. The proposed CRL system creates a list of certificates for vehicles exhibiting abnormal behaviors or executing malicious security attacks, and then distributes the list to other vehicles. When receiving a message and the CRL list, each vehicle checks if the certificate information of the sender vehicle is on the CRL list. If the certificate information is found in the CRL list, the message from the sender vehicle is not processed. Numerous studies have examined issues related to management authority and methods for configuring and distributing CRLs in vehicular network environments [17]. However, vehicles must always maintain their current CRLs. If the size of the CRL increases, the delay time is incurred by the search operations. Further, a vehicle with limited storage space cannot maintain a large CRL. Because vehicles have high mobility within vehicular network environments and typically utilize broadcast-based wireless communication technology, it is difficult to detect security attacks in real time.

In this paper, we propose a secure vehicular authentication architecture using decentralized edge computing systems. To demonstrate, the offset ratio and time-interval-based intrusion detection system (OTIDS) dataset is used in the controller area network (CAN). Because CAN traffic is broadcast from a transmitter to the other nodes on a CAN bus, it does not contain information about the source and destination addresses for validation. Therefore, an attacker can easily inject any message that leads to system malfunctions.

3 Edge lightweight authentication

3.1 Lightweight authentication architecture

Recent developments in unknown intrusion detection, and detailed analysis methods have facilitated the detection of abnormal behaviors in edge computing environments. Specific vehicular interface standards for acquiring relevant data from a cloud server located in an external infrastructure are currently under development. Cloud-based security systems are the most prevalent approach for enhancing vehicular security. However, this approach raises concerns regarding privacy invasion because the locations and driving information of vehicles can be exposed when vehicle-to-network (V2N) communications are tracked, and big data are analyzed. Unfortunately, the accident causes of many autonomous vehicles cannot be identified because there are a variety of attack paths, numerous unknown attack techniques, and huge amounts of vehicle data. Therefore, edge computing-based analyses and responses are becoming increasingly necessary.

Vehicular cloud technology enables the resources of multiple vehicles to be combined in a cloud, allowing vehicles to access services demanding cooperative work or significant resources. In other words, vehicles with limited resources can gain access to services by using resources from other vehicles in the same vehicular cloud. Conversely, these vehicles can provide resources that are currently not in use to demanding vehicles. However, various studies on vehicular clouds [18] have focused on the clouds of vehicles in close physical proximity. In such systems, in the absence of neighboring vehicles or insufficient resources of neighboring vehicles, a vehicular cloud cannot be formed. Mershad and Artail [19] could only form vehicular clouds for vehicles located within a certain distance from an RSU. However, once a vehicular cloud is formed, no security requirements (e.g., vehicle authentication or publication and distribution of encryption keys) are necessary for communication between the member vehicles of a vehicular cloud. For these requirements, this study proposes applying the concept of edge-computing-based authentication architecture to existing vehicular networks. Because a distributed EDC_V manages all vehicles, a secure vehicular architecture for remotely located vehicles can be formed.

Figure 2 shows the architecture of the proposed ELA system. Both CA and EDC_V are located in a cloud. The CA is a trusted organization that issues certificates for vehicles. Certificates are a type of official verification for each vehicle. If the same certificate is used repeatedly while a vehicle is running, the travel path of the vehicle can be tracked. Accordingly, the proposed architecture differentiates vehicle certificates into pseudonym certificates and registration certificates. The EDC_V registers vehicles, generates and manages vehicular clouds, publishes, and distributes authentication keys and encryption keys, and performs deep learning to detect attacks. To form an optimal vehicular cloud, EDC_V tracks the mobility and resource data of the vehicles. Multiple EDC_Vs can exist according to the network size. Additionally, because EDC_V performs deep learning continuously, vehicles can detect diverse security attacks. The RSUs and vehicles are grouped into clouds. Each RSU has a wired connection to the EDC_V and facilitates communication between vehicles and the EDC_V. Wireless V2V communication is feasible within the radius of communication. V2I communication is also possible through an RSU network. In addition, vehicles can receive various services through the Internet, such as 5G wireless networks.

In the proposed ELA architecture, every vehicle should receive a certificate from the CA and register the certificate with EDC_V prior to forming a vehicular cloud. As aforementioned, certificates are distinguished as registration or long-term certificates, which are initially issued by the CA, and pseudonyms or short-term certificates for updating. Privacy protection measures for the location information of vehicles should be implemented in a V2V communication environment. The locations and travel paths of specific vehicles should not be tracked. To satisfy these requirements, pseudonyms can be used for V2V certification.

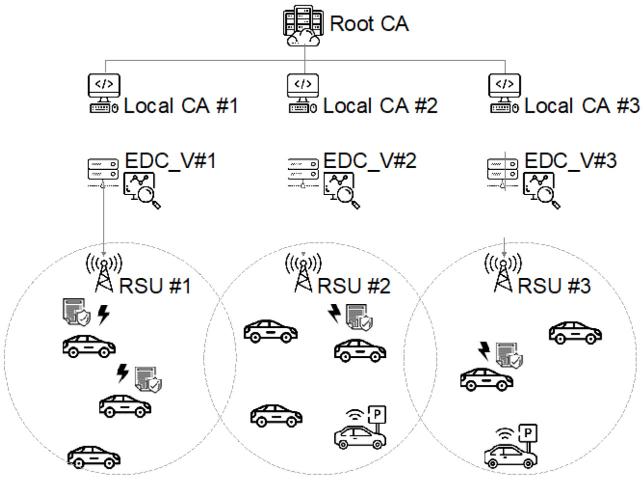


Figure 2. ELA architecture

Depending on the validity period of certificates, V2V communication utilizes two types of certificates (long-term certificates and short-term (or pseudonym) certificates). Long-term certification uses the unique ID of each vehicle, whereas short-term certification uses temporary pseudonyms allocated to each vehicle. Multiple pseudonyms can be assigned to each vehicle. The application of short-term certification using pseudonyms for V2V communication is a measure for protecting the privacy of vehicle location information. The messages sent and received between vehicles include digital signatures based on pseudonyms, which are provided by the proposed V2V communication authentication service infrastructure. This makes it possible to prevent messages and vehicle IDs from being falsified. Furthermore, because pseudonyms with short validity periods are employed, vehicle IDs does not require tracking, which prevents privacy intrusion regarding vehicle locations.

Figure 3 shows the overall process by which vehicles receive a certificate issued by the CA and register it with EDC_V. Details of this process are described below.

(1) A vehicle sends a certificate request message to the CA through an RSU network. If the vehicle is sending a certificate request to the CA for the first time, registration certification should be obtained for initial authentication. The local CA sends a request for a registration certificate to the root CA. If a pseudonym certificate is registered for updating certifications, the local CA performs registration to update the certification. The corresponding certificate request message includes unique identification information of the vehicle.

(2) The root CA creates a registration certificate by referring to the unique identification information of the vehicle. This certificate includes the certificate ID, public and private keys of the vehicle, digital signature of the CA, expiration date, time stamp, and validation results.

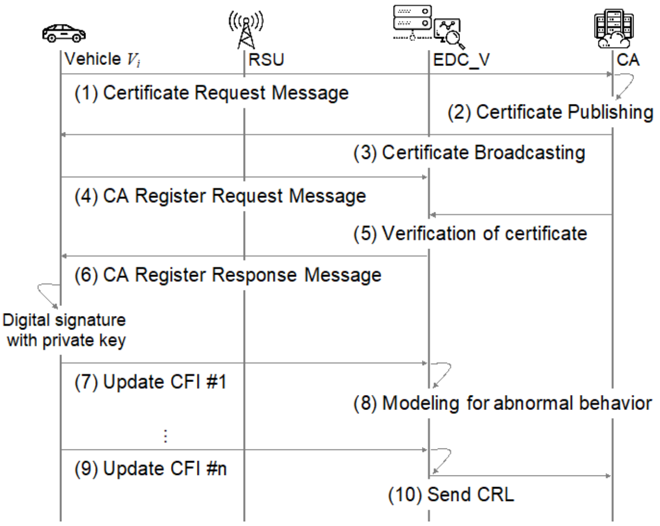

Vehicle V_i
RSU
EDC_V
CA
(1) Certificate Request Message
(2) Certificate Publishing
(3) Certificate Broadcasting
(4) CA Register Request Message
(5) Verification of certificate
(6) CA Register Response Message
Digital signature with private key
(7) Update CFI #1
(8) Modeling for abnormal behavior
(9) Update CFI #n
(10) Send CRL

Figure 3. Registration in the EDC_V

(3) After receiving a registration certificate from the root CA, the local CA sends a response message containing the registration certificate to the vehicle through the corresponding RSU network.

(4) The vehicle then registers the certificate with EDC_V to participate in a vehicular cloud. To register the certificate with EDC_V, the private key of the vehicle is used to create a digital signature in the certificate. For this, one of the most important solutions is the public key infrastructure (PKI), which is standardized by IEEE 1609.2, as the security solution for all V2X safety applications. In addition, the timed efficient stream loss-tolerant authentication (TESLA) protocol can be considered with a delayed symmetric key disclosure [20]. TESLA uses symmetric key cryptography, which is faster than using a digital signature. Another security protocol, VANET authentication using signatures and TESLA++ (VAST), is published in [21]. VAST targets different V2X

applications and provides multi-hop authentication. VAST is based on a mix of TESLA++ and elliptic curve digital signature algorithm (ECDSA) signatures. The digital signature method is beyond the scope of this study. Therefore, TESLA or VAST can be used as a digital signature method. To participate in a vehicular cloud, a vehicle broadcasts an encoded message containing both the digital signature for the message and certificate to the EDC_V.

(5) After receiving a registration request message, a public key for the CA is delivered to the EDC_V from the root CA to verify the validity of the certificate. The validity of the certificate is verified based on the public key of the CA. Moreover, EDC_V verifies the digital signature of the registration request message based on the public key of the vehicle.

(6) EDC_V sends a registration response message through the RSU network.

(7) The vehicle creates a CAN-based flow information (CFI) message and a digital signature using its private key. Furthermore, the vehicle registers the CFI by periodically sending update messages and digital signatures through an RSU network. Accordingly, EDC_V periodically acquires the latest CFI from the vehicles.

(8) EDC_V generates a set of training data by collecting periodically received CFI messages. Based on the set of training data, a data model for the normal flow of vehicles is constructed as the basis for a model for detecting abnormal flows. Vehicles with abnormal behaviors can be detected by inputting the CFI messages of vehicles, which are periodically updated, into a deep learning algorithm.

(9) If the CFI messages of vehicles that have been updated are determined to represent abnormal patterns, the information is broadcast and reported to the local CA.

(10) A local CA with information regarding vehicles with abnormal patterns requests the revocation of certificates by sending a certificate revocation list (CRL) for vehicular communication to the root CA.

3.2 CRL update for abnormal patterns

Vehicle communications typically distribute CRLs to every local CA to reflect the movements of vehicles when a vehicle certificate is revoked. In such cases, the CA has a hierarchical structure consisting of a root CA and multiple local CAs. Because this structure is complicated and covers a wide area, a significant overhead may be incurred when a CRL is sent. If several certificates of vehicles are revoked, the number of CRLs that must be distributed increases exponentially, which raises scalability issues. Additionally, because every local CA distributes CRLs, this process is very inefficient. Because a vehicle does not stay near the same local CA, CRLs must be distributed to every local CA from a networking perspective.

When a vehicle has an abnormal pattern, in this study, a local CA creates a CRL that records the revocation of the certificate for that vehicle, and then sends a request to distribute the CRL to the root CA. When the CRL is distributed, each local CA broadcasts the CRL to all vehicles within its jurisdiction through devices (e.g., RSUs) near roads. Because all vehicles are mobile, they can easily leave the area of the initial local CA. If another local CA does not have the CRL information for an abnormal vehicle, another certificate may be issued. To overcome this issue, the root CA sends a request message to all local CAs except for the local CA (i.e., referred to as local CA#1) with the CRL containing the abnormal vehicle. This message requests all remaining local CAs to check if their location information includes an abnormal vehicle. A geocoding algorithm can be used to determine the location of a vehicle using devices installed around roads [22]. In the process shown in Figure 4, the root CA determines the location of the abnormal vehicle and sends the CRL. The details of this process are discussed below.

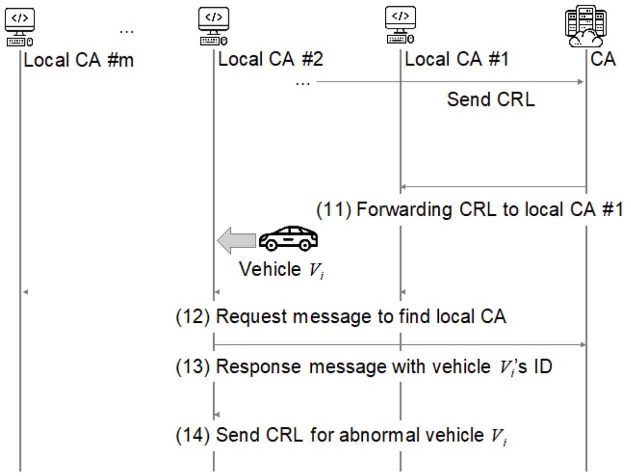

Local CA #m
...
Local CA #2
Local CA #1
CA
...
Send CRL
(11) Forwarding CRL to local CA #1
Vehicle V_i
(12) Request message to find local CA
(13) Response message with vehicle V_i's ID
(14) Send CRL for abnormal vehicle V_i

Figure 4. CRL update process

In step (10) in Figure 3, local CA#1 sends a CRL for vehicle communication to the root CA to request revocation of the certificate for a vehicle with an abnormal pattern.

(11) Because the vehicle may fall under the jurisdiction of local CA#1, the CRL containing the abnormal vehicle is sent via broadcasting.

(12) After the root CA receives the CRL following step (10) in Figure 3, it must send the CRL to the remaining local CAs because the abnormal vehicle will move. The unique identification information of the vehicle is sent to each local CA with a request message to identify the local CA corresponding to the abnormal vehicle.

(13) If the vehicle moves to another local CA (i.e., referred to as local CA#2), local CA#2 sends a response message to the root CA.

(14) After receiving a response message from the local CA#2, the root CA sends the CRL containing the abnormal vehicle to revoke its certificate.

Finally, after its certificate has been revoked, the abnormal vehicle can no longer operate in the vehicular cloud. If the certificate for a vehicle expires, it must be renewed by the CA. If the CA knows that the vehicle has an abnormal pattern, a response message is sent with the validity field set to zero.

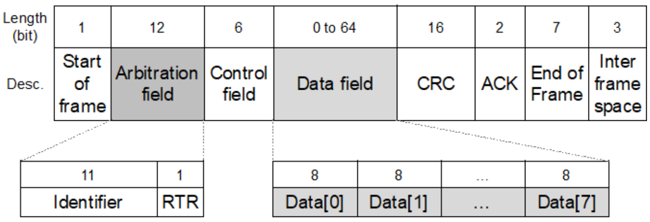
4 Deep learning for abnormal patterns

When vehicles communicate with each other through a vehicular network or use infotainment services through the Internet, they may be exposed to various security attacks. Moreover, a normal vehicle can suddenly become malicious by performing a security attack. Because vehicles largely utilize broadcast-based wireless communication, it is difficult to detect security attacks in real time. In this paper, flow information based on deep learning results is introduced to detect and classify security attacks hidden in the information flows between vehicles.

The entire process for detecting security attacks in vehicular cloud architecture can be divided into the following two main processes: collecting flow information from vehicles and implementing a deep learning algorithm in EDC_V. In the former, the member vehicles of a vehicular cloud collect the data flowing into each vehicle, process the data into packets, and periodically deliver them to the EDC_V.

In this paper, the OTIDS dataset [23] is used for abnormal pattern detection, which is an intrusion detection method, based on the analysis of the offset ratio, and time interval between request and response messages in CAN. The OTIDS dataset, which is the network attack data used for simulation results, is generated by logging CAN traffic through the OBD-II port in real vehicle. The generation of the dataset occurs by performing a message injection attack on the actual vehicle. To do this, the real vehicle used is the SOUL model of KIA brand in Korea, and three types of attack and normal data are included. The OTIDS dataset includes DoS attacks, fuzzy attacks, impersonation attacks, and attack-free states. Datasets are constructed by logging CAN traffic via the OBD-II port from a real vehicle, while message injection attacks are performed. The data attributes of CAN traffic are listed in Figure 5 [24].

1. Timestamp: recorded time (second)
2. CAN ID: identifier of CAN message in HEX
3. DLC: number of data bytes, from 0 to 8
4. DATA [0–7]: data value (byte)



Length (bit)	1	12	6	0 to 64	16	2	7	3
Desc.	Start of frame	Arbitration field	Control field	Data field	CRC	ACK	End of Frame	Inter frame space

11	1
Identifier	RTR

8	8	…	8
Data[0]	Data[1]	…	Data[7]

Figure 5. CAN message attributes

Table 1 Summary of dataset

Message type	# of messages
DoS attack	656,579
Fuzzy attack	591,990
Impersonation attack	995,472
Attack-free state (normal message)	2,369,868

Using the OTIDS dataset, we analyzed the detection results for abnormal patterns in the vehicle clouds. Three types of attacks and one attack-free state are included as output features. For the lightweight authentication architecture, the mobility vector, current EDC_V ID, ID_V_i, and OTIDS input features (e.g., timestamp, CAN ID, DLC, and Data) are added to define the CFI. We assume that a vehicle can estimate its mobility vectors using any of the methods explained in [25] with a safe circular communication region available [26].

Figure 6 shows a schematic of the operation process, including the input and output features for the deep learning algorithm. Table 1 summarizes the dataset used. In general, the data imbalance problem should be checked before applying the dataset to a deep learning algorithm. In the case of a network dataset, for example, attack data are too much or too little compared to normal data. However, the tested OTIDS dataset is collected after randomly generating three attacks and one normal traffic for 1hour and 20min; therefore, the data imbalance problem is not shown.

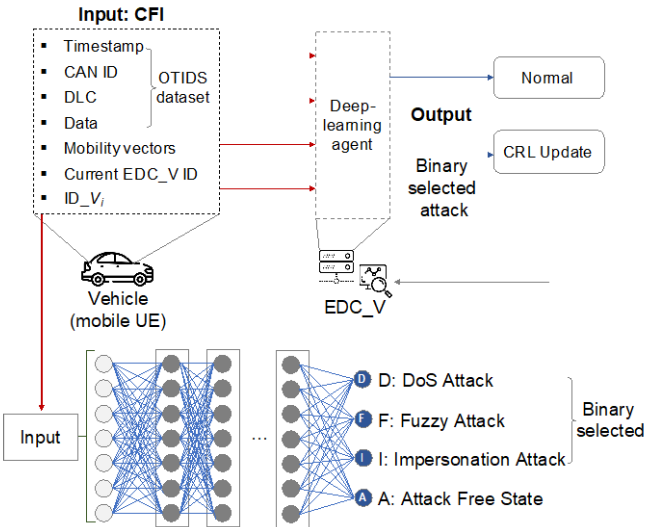


Figure 6. Deep learning architecture

5 Simulation results

This section describes the simulation results that detect security attacks, a type of abnormal pattern, using deep learning algorithms in an edge-based lightweight authentication architecture for vehicle networks. The simulation is performed based on the OTIDS dataset. The categories included in the OTIDS data set are defined as follows [23]:

1. DoS Attack: Injecting messages of '0x000' CAN ID in a short cycle
2. Fuzzy Attack: Injecting messages of spoofed random CAN ID and DATA values
3. Impersonation Attack: Injecting messages of Impersonating node, arbitration ID = 0x164
4. Attack Free State: Normal CAN messages

We create a virtual road topology using MATLAB2019b software. Table 2 lists the simulation parameters and deep-learning parameters used in the vehicle network environment. Figure 7 shows an example in which a virtual road topology is generated based on EDC_Vs; vehicles that are included and not included in the cloud. The vehicles are randomly placed in the topology. The simulation is performed while randomly changing the positions of the vehicles, and this is repeated 1000 times in each step.

In the simulation environment, we assume that EDC_V has a learning dataset (i.e., OTIDS dataset). That is, vehicles periodically send their CFI to EDC_V. EDC_V formed a vehicle cloud according to the proposed ELA architecture and received CFI, which is a training dataset from vehicle members. EDC_V then performed a deep learning algorithm to detect abnormal patterns.

Deep learning processes are implemented using Python on Ubuntu 14.0.4 LTS. Deep learning models are implemented using GPU-enabled TensorFlow4 as backend with a Keras5 higher-level framework. The GPU is NVidia GK110BGL Tesla K40, and the CPU has a configuration (32 GB RAM, 2

TB hard disk, Intel(R) Xeon(R) CPU E3-1220 v3 @ 3.10 GHz) running over a 1 Gbps Ethernet network.

Table 2 Simulation parameters

Parameter	Value
Virtual road topology size	1600*m* x 1600*m*
Number of EDC_V	16
Number of vehicles	20–60
Communication range of EDC_V	300*m* x 300*m*
Topology model	Random
Number of simulation iteration	1000
Deep learning parameter	Value
Number of layers	Hidden (1–8)
Output level	4
Weight initialization	Xavier initializer
Activation function	ReLU
Cost function	Cross-entropy
Optimizer	ADAM

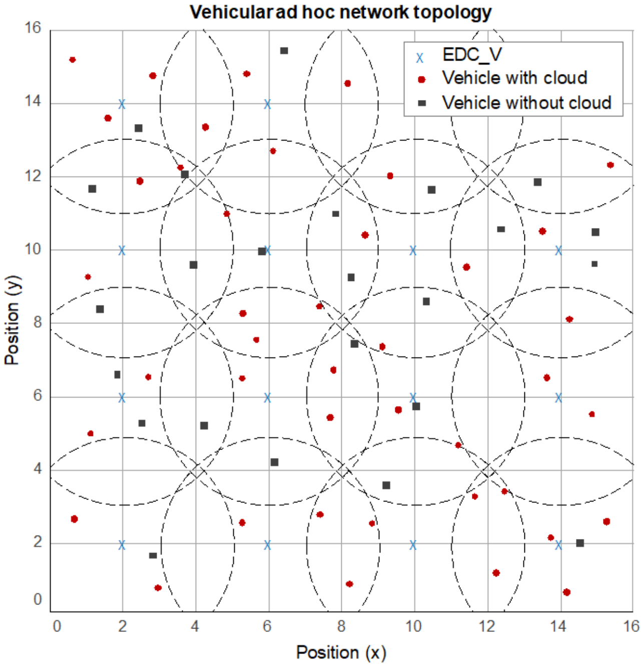


Figure 7. Virtual road topology

As the deep learning algorithm is parameterized, the performance depends on the optimal parameters. To identify the ideal parameters for the deep learning algorithm, a medium-sized architecture is used for experiments with specific hidden units, learning rate, and activation function. A medium-sized deep-learning algorithm contains three layers. One is the input layer, the second is the hidden layer or fully connected layer, and the third is the output layer. The connections between the units between the input and hidden layers and the hidden layer to the output layer are fully connected. Initially, the training and test datasets are normalized using L2 normalization. The experiment is performed for each parameter with appropriate units and for 300 epochs.

To find an optimal learning rate, three trials of experiments for 300 epochs with learning rates varying in the range [0.0001–1.0] are performed. The learning rate has a strong impact on the training speed. The peak value for the detection rate is obtained when the learning rate is 0.1. In the experimental trials, we also conducted experiments with the sigmoid and tanh activation functions for multi-class classification. When the set of experiments is carried out for 300 epochs with activation functions, the performance of the ReLU activation function is better than that of the sigmoid and tanh activation functions. In addition, all the models are trained using the ADAM optimizer with a batch size of 64 for 300 epochs to monitor the validation accuracy.

A deep learning algorithm is implemented for simulations in a vehicular ELA architecture. In general, the performance of deep learning is analyzed in terms of precision, recall, and accuracy. To derive these metrics, true positive (TP), false positive (FP), false negative (FN), and true negative (TN) rates are calculated. A TP is the correct detection of a positive outcome. An FP is a misdetection of a positive outcome that is actually negative. An FN is a misdetection of a negative outcome that is actually positive. A TN is the correct detection of a negative outcome.

Table 3 F1-score of proposed ELA architecture

	1 hidden layer	2 hidden layers	4 hidden layers	8 hidden layers
Prediction Accuracy (F1-score)	94.51	96.52	97.29	99.81

For the vehicle mobility model, we acquired data from nodes that moved randomly at speeds between 1 and 30 m/s. As a mobility model, we have prepared a scenario moving at a speed of 10–30 m/s (high-speed scenario) and a scenario moving at a speed of 1–10 m/s (low speed scenario). As shown in Table 3, the maximum prediction accuracy of the proposed method is 99.8% with the deep learning model. This means that even in the environment of a moving vehicle, abnormal patterns that are randomly generated can be predicted. In addition, it can be seen that the larger the hidden layer of the deep learning model, the greater the improvement in prediction accuracy (i.e., F1-score).

Figure 8 shows the F1-score result, which is the predicted accuracy of abnormal patterns according to the increase in the number of vehicles. Here, the F1-score is an accuracy calculated through precision and recall. Precision represents the ratio of the number of flows that are actually security attacks among the flows determined as security attacks. By contrast, the recall rate represents the rate at which the model perceives an actual security attack as a security attack. To evaluate the performance of the proposed ELA architecture, we compared the precision values of several deep learning models. Because the dataset used in this study is complex and does not have a large feature set, the results of machine learning algorithms are also compared through experiments. It analyzes whether security attacks can be predicted using support vector machine (SVM), decision tree (DT), and random forest (RF) algorithms, which are representative algorithms of machine learning algorithms. In this case, L represents a low-speed scenario, and H represents a high-speed scenario. In the case of the deep learning algorithm, this is the result of applying eight hidden layers.

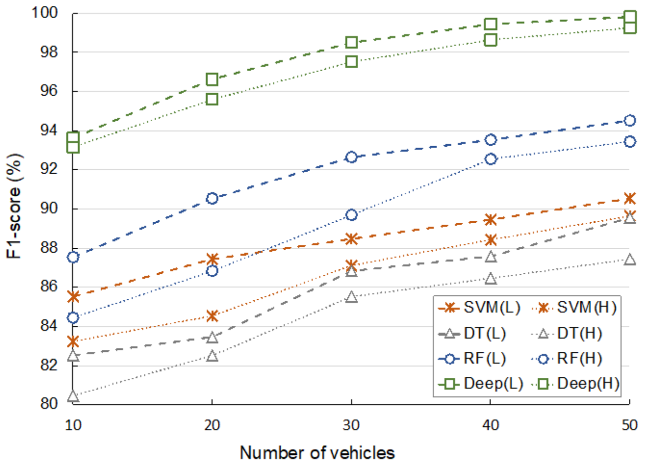


Figure 8. F1-score results based on number of vehicles

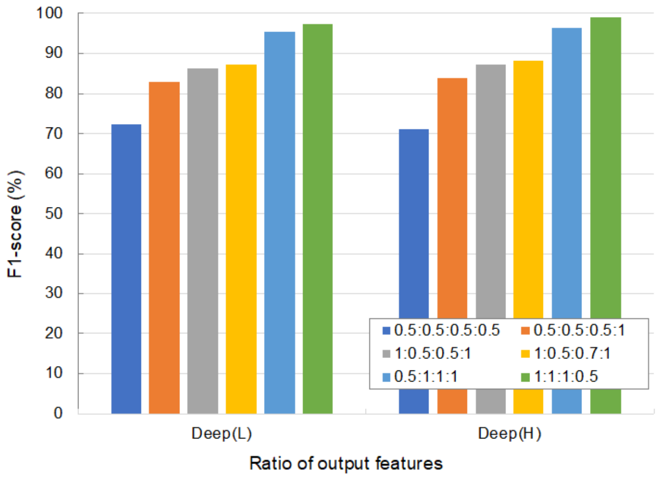


Figure 9. F1-score results based on dataset ratio

Figure 9 shows the detection results obtained by changing the attack rate of the dataset. We defined a weight factor to adjust the attack rate. In other words, the weight factor is applied to each feature of the output with a value of 0–1. For example, we can reduce the ratio by putting the ratio of attacks less than 1 (e.g., DoS attack feature: weight factor of 0.7, fuzzy attack feature: weight factor of 0.5, impersonation attack feature: weight factor of 0.5), and weight of only the attack-free state to 1. If the weight factor is applied, the number of attack datasets is reduced. In this case, because there are many combinations of each ratio, only a few cases showing a large difference in performance are typically selected, and the results are shown in Figure 9. To clarify the legend, the output feature is specified as A:B:C:D, where A represents a DoS attack, B represents a fuzzy attack, C represents an impersonation attack, and D represents an attack-free state.

In the case of 0.5:0.5:0.5:0.5, where the ratio of the dataset is reduced by half as a whole, the F1-score, which is the prediction accuracy, is about 72%. Therefore, it can be seen that the prediction accuracy increases only when the ratio of the dataset is sufficient. In addition, it can be seen that in the case of the dataset with only the ratio of the attack-free state lowered to 0.5, the F1-score is still around 98% and is high. It can be seen that even in the case of the original dataset, because the number of attack-free states is large, it did not significantly affect the prediction accuracy. By contrast, in the case of 0.5:0.5:0.5:1, where the ratio of attack data is lowered overall, it can be seen that the F1-score is degraded due to data imbalance. Even in the original dataset, the percentage of attack data is low, which seems to be because the percentage is further reduced by half. In the case of 1:0.5:0.5:1, the ratio of DoS attack and attack-free state is set to 1, and the remaining two attacks are reduced by a ratio of 0.5, and an accuracy of approximately 88% is achieved. Therefore, this indicates the importance of the ratio and the number of datasets.

6 Conclusion

This paper proposed an edge-based lightweight authentication architecture for vehicular networks using a cloud-based vehicular security system. In this paper, the proposed communication security architecture utilized the edge-based EDC_V to create and distribute authentication keys for verifying the member vehicles in a vehicular cloud, and private keys for the safe exchange of messages between member vehicles. Moreover, an edge-based cloud data center performed deep learning models to detect car hacking attempts, and then delivered the detection results to a vehicular cloud. Extensive simulations demonstrated that the proposed authentication architecture significantly enhanced the security level. The proposed authentication architecture has high detection results, depending on the number of vehicles in the intrusion detection system using CAN traffic. In future work, we are considering a scenario for various attacks that can be applied to the vehicle in the actual road environment.

Acknowledgement

This work was supported by 2020 Research Fund of Myongji University and Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT)(No. 2021-0-00990, Research on Advanced Core Technologies for WLAN based on eXplainable AI).

References

[1] A. Lamssaggad, N. Benamar, A. S. Hafid, M. Msahli, A Survey on the Current Security Landscape of Intelligent Transportation Systems, *IEEE Access*, Vol. 9, pp. 9180-9208, 2021.

[2] R. W. L. Coutinho, A. Boukerche, Modeling and Analysis of a Shared Edge Caching System for Connected Cars and Industrial IoT-Based Applications, *IEEE Transactions on Industrial Informatics*, Vol. 16, No. 3, pp. 2003-2012, March, 2020.

[3] A. Velosa, E. Perkins, H. LeHong, J. F. Hines, R. M. Satish, *Predicts 2015: The Internet of Things*, Gartner report ID: G00269692, December, 2014.

[4] L. D. Xu, W. He, S. Li, Internet of things in industries: A survey, *IEEE Transactions on Industrial Informatics*, Vol. 10, No. 4, pp. 2233-2243, November, 2014.

[5] S. Olariu, I. Khalil, M. Abuelela, Taking VANET to the clouds, *International Journal of Pervasive Computing and Communications*, Vol. 7, No. 1, pp. 7-21, April, 2011.

[6] R. Yu, X. Huang, J. Kang, J. Ding, S. Maharjan, S. Gjessing, Y. Zhang, Cooperative Resource Management in Cloud-Enabled Vehicular Networks, *IEEE Transactions on Industrial Electronics*, Vol. 62, No. 12, pp. 7938-7951, December, 2015.

[7] E. Lee, E. -K. Lee, M. Gerla, S. Y. Oh, Vehicular Cloud Networking: Architecture and Design Principles, *IEEE Communications Magazine*, Vol. 52, No. 2, pp. 148-155, February, 2014.

[8] M. Azees, P. Vijayakumar, L. J. Deborah, Comprehensive Survey on Security Services in Vehicular Ad-hoc Networks, *IET Intelligent Transport Systems*, Vol. 10, No. 6, pp. 379-388, August, 2016.

[9] F. Qu, Z. Wu, F.-Y. Wang, W. Cho, A Security and Privacy Review of VANETs, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, No. 6, pp. 2985-2996, December, 2015.

[10] IEEE 802.11p/D3.0 Draft Amendment for Wireless Access in Vehicular Environments (WAVE), July 2007.

[11] Y. L. Morgan, Notes on DSRC & WAVE Standards Suite: Its Architecture, Design, and Characteristics, *IEEE Communications Surveys & Tutorials*, Vol. 12, No. 4, pp. 504-518, Fourth Quarter, 2010.

[12] M. N. Mejri, J. Ben-Othman, M. Hamdi, Survey on VANET Security Challenges and Possible Cryptographic Solutions, *Vehicular Communications*, Vol. 1, No. 2, pp. 53-66, April, 2014.

[13] H. Qiu, M. Qiu, R. Lu, Secure V2X Communication Network based on Intelligent PKI and Edge Computing, *IEEE Network*, Vol. 34, No. 2, pp. 172-178, March/April, 2020.

[14] L. Bariah, D. Shehada, E. Salahat, C. Y. Yeun, Recent Advances in VANET Security: A Survey, *IEEE Vehicular Technology Conference Fall 2015*, Boston, MA, USA, 2015, pp. 1-7.

[15] A. Wasef, X. Shen, EDR: Efficient Decentralized Revocation Protocol for Vehicular Ad Hoc Networks, *IEEE Transactions on Vehicular Technology*, Vol. 58, No. 9, pp. 5214-5224, November, 2009.

[16] Q. Wang, D. Gao, D. Chen, Certificate Revocation Schemes in Vehicular Networks: A Survey, *IEEE Access*, Vol. 8, pp. 26223-26234, 2020.

[17] J. J. Haas, Y.-C. Hu, K. P. Laberteaux, Efficient Certificate Revocation List Organization and Distribution, *IEEE Journal on Selected Areas in Communications*, Vol. 29, No. 3, pp. 595-604, March, 2011.

[18] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, H. Zedan, A comprehensive survey on Vehicular Ad Hoc Network, *Journal of Network and Computer Applications*, Vol. 37, pp. 380-392, January, 2014.

[19] K. Mershad, H. Artail, Finding a STAR in a Vehicular Cloud, *IEEE Intelligent Transportation Systems Magazine*, Vol. 5, No. 2, pp. 55-68, Summer, 2013.

[20] A. Perrig, J. D. Tygar, TESLA broadcast authentication, in: *Secure Broadcast Communication*, Boston, USA, Springer, 2003, pp. 29-53.

[21] A. Studer, F. Bai, B. Bellur, A. Perrig, Flexible, extensible, and efficient VANET authentication, *Journal of Communications and Networks*, Vol. 11, No. 6, pp. 574-588, December, 2009.

[22] P. A. Zandbergen, Geocoding quality and implications for spatial analysis, *Geography Compass*, Vol. 3, No. 2, pp. 647-680, March, 2009.

[23] H. Lee, S. Jeong, H. Kim, OTIDS: A Novel Intrusion Detection System for In-vehicle Network by using Remote Frame, *Annual Conference on Privacy, Security and Trust*, Calgary, Canada, 2017, pp. 57-66.

[24] A. Derhab, M. Belaoued, I. Mohiuddin, F. Kurniawan, M. K. Khan, Histogram-Based Intrusion Detection and Filtering Framework for Secure and Safe In-Vehicle Networks, *IEEE Transactions on Intelligent Transportation Systems*, Early access, pp. 1-14, June, 2021.

[25] S. Milnelli, P. Izadpanah, S. Razavi, Evaluation of connected vehicle impact on mobility and mode choice, *Journal of Traffic and Transportation Engineering*, Vol. 2, No. 5, pp. 301-312, October, 2015.

[26] S. Wang, C. Zhang, A Dynamic Interval Based Circular Safe Region Algorithm for Continuous Queries on Moving Objects, *International Journal of Communications, Network and System Sciences*, Vol. 4, No. 5, pp. 313-322, May, 2011.

Biography

Hyunhee Park is an associate professor with the department of Information and Communication, Myongji university, South Korea. Dr. Park received the Ph.D. degree from the School of Electronics and Computer Engineering, Korea University, South Korea, in August 2011. From September 2011 to February 2013, Dr. Park was a Research Professor with the Information Technology Center, Korea University. From January 2013 to November 2014, Dr. Park was a Postdoctoral Researcher with the INRIA Research Center FRANCE, where she works in DIONYSOS Research Group. Dr. Park was a Postdoctoral Researcher with Telecom Bretagne FRANCE. From November 2014 to February 2017, Dr. Park was a Senior Researcher with LG Electronics for Wi-Fi standardization. From March 2017 to February 2020, Dr. Park was an Assistant Professor with the Department of Computer Software, Korean Bible University. Since 2020, Dr. Park has been an Assistant Professor with the Department of Information and Communication Engineering, Myongji University, South Korea. Dr. Park is currently a Supervisor of Data Analysis and Networking (DAN) Laboratory. Her research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular ad hoc and machine learning/deep learning algorithms.