An Access Control Mechanism Based on Risk Prediction for the IoV

Yuanni Liu[1], Man Xiao[1], Yanyan Zhou[1], Di Zhang[2], Jianhui Zhang[3, *], Haris Gacanin[4], Jianli Pan[5]
Chongqing University of Posts and Telecommunications, Chongqing, China[1]
Zhengzhou University, Zhengzhou, China.[2]
National Digital Switching System Engineering & Technological R&D Center, China[3]
Nokia Bell Labs, Antwerp, Belgium[4]
Department of Mathematics and Computer Science, University of Missouri-St.Louis, MO, USA[5]
Email:liuyn@cqupt.edu.cn

***Abstract*—The information sharing among vehicles provides intelligent transport applications in the Internet of Vehicles (IoV), such as self-driving and traffic awareness. However, due to the openness of the wireless communication (e.g. DSRC), the integrity, confidentiality and availability of information resources are easy to be hacked by illegal access, which threatens the security of the related IoV applications. In this paper, we propose a novel Risk Prediction-Based Access Control model, named RPBAC, which assigns the access rights to a node by predicting the risk level. Considering the impact of limited training datasets on prediction accuracy, we first introduce the Generative Adversarial Network (GAN) in our risk prediction module. The GAN increases the items of training sets to train the Neural Network, which is used to predict the risk level of vehicles. In addition, focusing on the problem of pattern collapse and gradient disappearance in the traditional GAN, we develop a combined GAN based on Wasserstein distance, named WCGAN, to improve the convergence time of the training model. The simulation results show that the WCGAN has a faster convergence speed than the traditional GAN, and the datasets generated by WCGAN have a higher similarity with real datasets. Moreover, the Neural Network (NN) trained with the datasets generated by WCGAN and real datasets (NN-WCGAN) performs a faster speed of training, a higher prediction accuracy and a lower false negative rate than the Neural Network trained with the datasets generated by GAN and real datasets (NN-GAN), and the Neural Network trained with the real datasets (NN). Additionally, the RPBAC model can improve the accuracy of access control to a great extent.**

***Index Terms*—Access Control, Generative Adversarial Network, Internet of Vehicles, Risk Prediction**

I. Introduction

As an emerging paradigm, the Internet of Vehicles (IoV) supports wireless communication for information sharing between vehicles [1]–[3], improving the safety of traffic [4], [5]. However, the wireless communication faces many threats, such as replay attack [6], [7]. Therefore, it is critical to protect the safety of information resources [8].

The access control mechanism aims to reduce the unauthorized access in the IoV. Existing access control mechanisms, such as Role-Based Access Control (RBAC) [9] and Attribute-Based Access Control (ABAC) [10], are usually based on static methods. Therefore, once a node is authorized, it will not be changed. Additionally, when a node is attacked during access activities, the system is difficult to make a timely response to protect the resources. Aim at this problem, Weng et al. [11] have proposed a dynamic scheme for the SDN-based VANET, the administrator appoints qualified SDN applications to access resources by allocating a secret key dynamically. Zhang et al. [12] have proposed a Global Access Control model, and tackle the need in vehicular communication to make decisions based on dynamic information.

The risk prediction model predicts the risk level of a node, which acts as the basis for evaluating the access rights. Najada et al. [13] have validated that the prediction model based on Neural Network (NN) is better to process behavior data of vehicles than other models. Similarly, Mao et al. [14] have developed a risk prediction model based on the NN, which predicts the traffic accidents effectively. These solutions improves the traffic safety to some extent. However, they need lots of training sets to guarantee the accuracy of models. One way to overcome this problem is the Generative Adversarial Network (GAN). For example, Sun et al. [15] have leveraged the GAN to repair the parking data, which improves the performance of parking guidance system. Additionally, considering the impact of mode collapse and gradient disappearance on the performance of GAN, Arjovsky et al. [16] have replaced the Jensen-Shannon divergence with the Wasserstein-1 distance to solve the problem of gradient disappearance. Zhang et al. [17] have proposed a new architecture for the GAN, which uses two different discriminators to solve the problem of mode collapse.

Therefore, this paper proposes a Risk Prediction-Based Access Control (RPBAC) model to secure the information sharing between vehicles. Furthermore, focusing on the problem of pattern collapse and gradient disappearance, a combined GAN based on Wasserstein distance (WCGAN) is proposed. The main contributions of this paper are as follows:

- We propose a Risk Prediction-Based Access Control (RPBAC) model. Considering the problem of limited training sets, we introduce the GAN in the risk prediction module, the generator generates new datasets by learning the distributions of real datasets, which increases the items of training sets, improving the accuracy of the NN.
- Considering the problem of pattern collapse and gradient disappearance, we design a combined GAN based on Wasserstein distance (WCGAN). By using multiple

978-1-7281-5207-3/20/$31.00 ©2020 IEEE

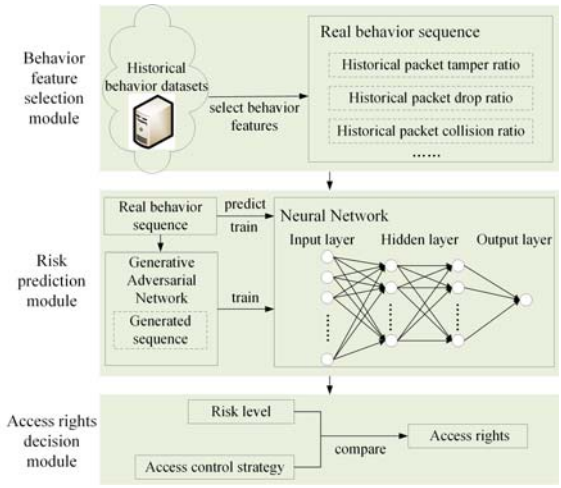


Fig. 1. The Risk Prediction-Based Access Control model.

generators, the WCGAN covers different modes.

The rest of this paper is organized as follows. Section II presents the proposed RPBAC model. Section III explains the problem in the traditional GAN, formulates the proposed method WCGAN and states the method algorithm. The performance of the RPBAC and the WCGAN are evaluated in Section IV. Section V concludes this paper.

II. The Risk Prediction-Based Access Control Model

Fig. 1 shows the proposed RPBAC model, which includes the behavior feature selection module, the risk prediction module and the access rights decision module. The detailed descriptions of these modules are as follows.

A. The Behavior Feature Selection Module

The principal purpose of behavior feature selection module is to select behavior features from the historical datasets. Behavior features, including the received signal strength, the packet transmitted amount, the packet received amount, the packet delivery ratio, the packet drop ratio, the packet capture ratio, the packet collision ratio, the packet re-transmission ratio, and the packet tamper ratio [18], are selected as a representative sequence to train the risk prediction module.

B. The Risk Prediction Module

The risk prediction module consists of a four-layer Neural Network (NN), including one input layer, two hidden layers and one output layer. The output layer has one unit, which is a sigmoid function given by (1).

$$f(x) = \frac{1}{1+e^{-x}} \tag{1}$$

The range of the function is $[0, 1]$. If the input represents a normal behavior, the output is close to 0, otherwise, the output is close to 1.

Considering the problem of limited training sets, the risk prediction module introduces the GAN. Fig. 2 shows

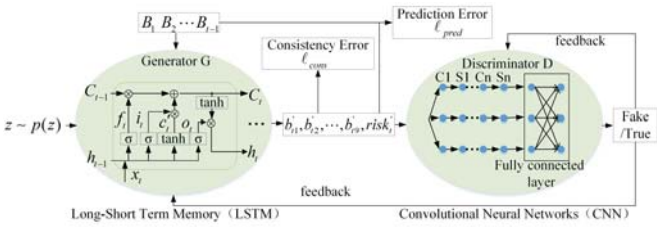


Fig. 2. The structure of Generative Adversarial Network.

the structure of GAN, the Long Short-Term Memory (LSTM) is acted as the generator G, and the Convolutional Neural Network (CNN) is acted as the discriminator D. The historical behavior is represented as $B = \{B_1, B_2, ..., B_{t-1}\}$, where B_{t-1} is a subsequence at time $t-1$, and $B_{t-1} = \{b_{t-1,1}, b_{t-1,2}, ..., b_{t-1,9}, risk_{t-1}\}$ with ten features, such as $b_{t-1,1}$ is the packet transmitted amount and $risk_{t-1}$ is the risk level. Given a sequence B, a new sequence $B_t = \{b'_{t,1}, b'_{t,2}, ..., b'_{t,9}, risk'_t\}$ can be generated by the G, where $risk'_t$ is the risk level of this new sequence.

The training process of the GAN can be attributed to the "minimax game" between the G and the D, where the G generates new sequence as accurately as possible to "cheat" the D, and the D discriminates whether the input is a real sequence to "beat" the G. The objective function of the GAN is given by (2).

$$\begin{aligned}\min_G \max_D V(D,G) = &E_{b\sim p_{data}(b)}[\log D(b)] \\ &+E_{z\sim p_z(z)}[\log(1-D(G(z)))]\end{aligned} \tag{2}$$

Where the D aims to maximize the expectation of $\log D(b) + \log(1 - D(G(z)))$, on the contrary, the G aims to minimize the expectation of $\log(1 - D(G(z)))$

C. The Access Rights Decision Module

This module is responsible for determining the access rights of a node, based on a predefined strategy. In this paper, we customize the access control strategy with a threshold n, $n = 0.5$. Therefore, the vehicle could access the information resource, when the predicted risk level is below n.

III. The Combined GAN based on Wasserstein Distance

In this section, we describe the problem in the traditional GAN, and state the proposed method WCGAN in terms of Wasserstein distance and the combined GAN.

A. Problem Statement

1) Gradient disappearance: The objective function of the GAN can be equivalent to optimizing the Jensen-Shannon divergence between the real distributions P_r and the generated distributions P_g. The JS divergence and the associated KL distance are expressed as (3).

$$\begin{cases} KL(P_r||P_g) = \int P_r(x)\log(\frac{P_r(x)}{P_g(x)})dx \\ JS(P_r||P_g) = \frac{1}{2}KL(P_r||\frac{P_g(x)+P_r(x)}{2}) \\ \qquad\qquad +\frac{1}{2}KL(P_g||\frac{P_g(x)+P_r(x)}{2}) \end{cases} \tag{3}$$

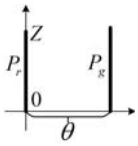


Fig. 3. The distribution of P_r and P_g.

Where P_r and P_g are continuous, when the two distributions do not overlap with each other, the Jensen-Shannon divergence tends to be a constant, which leads to the disappearance of gradient.

2) Mode collapse: In the GAN, the learning ability of a single generator is limited. The single generator just fits part distributions of real datasets, which leads to the lack of modes.

B. The Method Statement of WCGAN

1) The loss function based on Wasserstein distance: The WCGAN replaces the JS divergence with the Wasserstein distance to solve the problem of gradient disappearance. Wasserstein distance calculated in (4), aims to find the lowest cost of converting P_g into P_r.

$$W(P_r, P_g) = \inf_{\gamma \sim \prod(P_r, P_g)} E_{(x,y)\sim\gamma}[||x - y||] \tag{4}$$

Where $\prod(P_r, P_g)$ represents all possible joint between P_r and P_g, and $||x-y||$ is the distance between the real dataset y and the generated dataset x.

As shown in Fig. 3, P_r is the uniform distribution of $(0, Z)$, and P_g is the uniform distribution of (θ, Z), where θ is the vertical distance between P_r and P_g. The JS divergence and the Wasserstein distance between P_r and P_g are formulated as (5).

$$\begin{cases} JS(P_r||P_g) = \begin{cases} \log(\frac{P_g(x)}{\frac{1}{2}P_g(x)+0}) = \log 2, if\theta \neq 0 \\ 0, if\theta = 0 \end{cases} \\ W(P_r||P_g) = |\theta| \end{cases} \tag{5}$$

When θ equals to 0, the JS divergence also equals to 0. When θ is greater than 0, the JS divergence is a constant, which cannot provide an effective gradient. However, the Wasserstein distance changes with θ, providing an gradient for training.

The WCGAN introduces the Wasserstein distance into the loss function of discriminator. Compared with (2), the resistance loss of discriminator and the resistance loss of generator can be expressed as (6) and (7).

$$L_d^{Ad} = D(\tilde{s}_{0:t}) - D(s_{0:t}) \tag{6}$$

$$L_g^{Ad} = -L_d^{Ad} = D(s_{0:t}) - D(\tilde{s}_{0:t}) \tag{7}$$

The goal of discriminator becomes to minimize $D(\tilde{s}_{0:t}) - D(s_{0:t})$, and, the goal of generator becomes to minimize $D(s_{0:t}) - D(\tilde{s}_{0:t})$, where $\tilde{s}_{0:t}$ is a generated dataset, and $s_{0:t}$ is a real dataset.

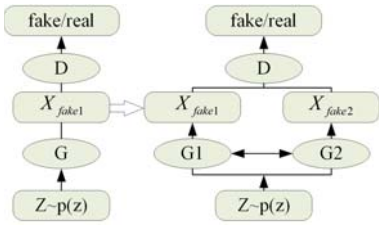


Fig. 4. The combined Generative Adversarial Network.

2) The combined GAN: In order to deal with the problem of mode collapse, the WCGAN combines multiple generators to cover multiple distributions. As shown in Fig. 4, we utilize two generators, which share an input and complement with each other to learn more modes.

Given m generators, the discriminator produces a probability distribution of soft-max over $m+1$ classes. The probability that the input belongs to real sequence is denoted by $D_{m+1}(.)$. Combined with (2), the objective function of GAN is given by (8).

$$\begin{aligned} &\min_G \max_D E_{x\sim p_{data}} \log D_{m+1}(x;\theta_d) \\ &+E_{x\sim p_z} \log(1 - D_{m+1}(G_i(z;\theta_g^i);\theta_d) \end{aligned} \tag{8}$$

Where θ_d is the parameter of discriminator, θ_g^i is the parameter of generator i, $G_i(z;\theta_g^i)$ represents the generate function of generator i, and $D_{m+1}(G_i(z;\theta_g^i);\theta_d)$ represents the probability that $G_i(z;\theta_g^i)$ is discriminated as a real dataset.

Combined with (6), aim at the output of generator i, the loss function of discriminator is calculated as (9).

$$\ell_D^i = D_{m+1}(G_i(z;\theta_g^i);\theta_d) - D_{m+1}(x;\theta_d) \tag{9}$$

In (9), the discriminator aims to maximize the probability that the real data set x is discriminated as a real data set, and minimize the probability that the generated data set $G_i(z;\theta_g^i)$ is discriminated as a real data set. Similarly, aim at the output of all generators, the loss function of the discriminator is calculated as (10).

$$L_D = \frac{1}{m}\sum_{i=1}^{m} \ell_D^i \tag{10}$$

Consequently, the gradient of the discriminator is calculated in (11).

$$\nabla_{\theta_d} L_D = \nabla_{\theta_d}[\frac{1}{m}\sum_{i=1}^{m} \ell_D^i] \tag{11}$$

Combined with (7) and (9), the goal of the generator i is the opposite of the discriminator, as calculated in (12).

$$\ell_G^i = D_{m+1}(x;\theta_d) - D_{m+1}(G_i(z;\theta_g^i);\theta_d) \tag{12}$$

Therefore, the gradient of the generator i is calculated in (13).

$$\nabla_{\theta_g^i} \ell_G^i = \nabla_{\theta_g^i}[-D_{m+1}(G_i(z;\theta_g^i);\theta_d)] \tag{13}$$

In this case, all generators can be updated in parallel. The generators form a hybrid mode caused by the objective function. When $p_d = \frac{1}{m}\sum_{i=1}^{m} p_{g_i}$, each generator represents a hybrid component to achieve global optimality.

3) The WCGAN procedure: The pseudo code of the WCGAN training is illustrated in Algorithm 1. First, we initialize the parameters θ_g and θ_d. Second, the generator i generates a dataset B_t^i at time t by learning the distributions of real datasets $B = \{B_0, B_1, \cdots, B_{t-1}\}$. Third, the θ_d is updated n_{disc} times based on the gradient of discriminator and the learning rate λ. Fourth, the m generators generate new datasets to calculate their loss function and gradient function respectively, which are used to update θ_g^i. Then, the parameter θ_g and θ_d are updated alternately, until θ_g converges. Finally, the m generators G_{θ_g} are obtained.

Algorithm 1 The training algorithm of the WCGAN.

Input: learning rate $\lambda = 0.1$, parameters θ_g and θ_d, the number of generators $m = 2$, the number of iterations $n_{disc} = 5$, historical behavior datasets of nodes, random distribution $z \sim p(z)$.

Output: the m generators G_{θ_g}.

Initialization θ_d and θ_g
while θ_g has not converged **do**
  for $t = 1$ to n_{disc} **do**
    for $i = 1$ to m **do**
      Sample behavior datasets $B = \{B_0, B_1, \cdots, B_{t-1}\}$, set random distribution $z \sim p(z)$
      Calculate $B_t^i \leftarrow G_i(B, z)$
      Calculate $\ell_D^i \leftarrow D_{m+1}(G_i(z; \theta_g^i); \theta_d) - D_{m+1}(x; \theta_d)$
    end for
    Update the parameters $\theta_d \leftarrow \theta_d + \lambda \nabla_{\theta_d} \frac{1}{m} \sum_{i=0}^{m} \ell_D^i$
  end for
  for $i = 1$ to m **do**
    Sample behavior datasets $B = \{B_0, B_1, \cdots, B_{t-1}\}$, set random distribution $z \sim p(z)$
    Calculate $B_t^i \leftarrow G_i(B, z)$
    Calculate $\ell_G^i \leftarrow D_{m+1}(x; \theta_d) - D_{m+1}(G_i(z; \theta_g^i); \theta_d)$
  end for
  Update the parameters $\theta_g^i \leftarrow \theta_g^i + \lambda \nabla_{\theta_g^i} \ell_G^i$
end while
Return to the m generators G_{θ_g}

IV. EVALUATION

In this section, we mainly conduct related experiments to evaluate the performance of the proposed method WCGAN and the proposed RPBAC model.

A. Experimental Setup

The related experiments are conducted in the TensorFlow 1.12.0, running on a PC with 8G memory and Intel Core i5 3.3GHz, the simulation process is written in Python. The experimental datasets are from an intrusion detection project conducted by MIT Lincoln Laboratory [19]. The datasets are divided into two parts: 5 million items of communication behavior for training, 2 million items of communication behavior for testing. We extract 0.2% random samples in the training sets, including 5000 items of normal records and 5000 items of abnormal records. The learning rate is λ, $\lambda = 0.1$.

B. The Performance Evaluation of the method WCGAN

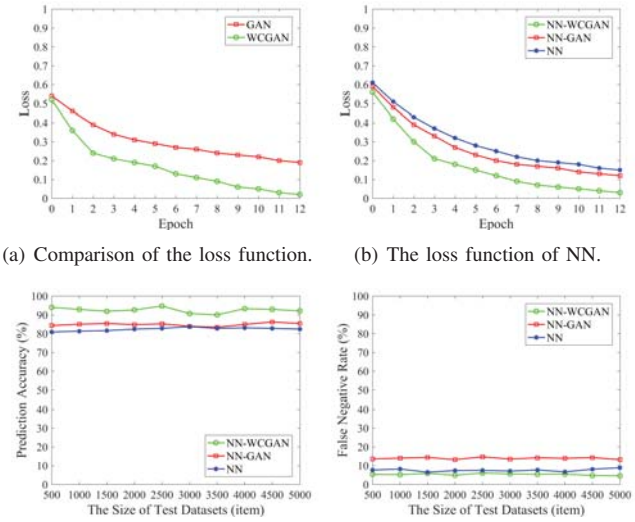
In order to validate the effectiveness of the proposed WCGAN, we first use the real training datasets to train the GAN and the WCGAN. Additionally, the WCGAN and the GAN are both used to generate 5000 items of normal records and 5000 items of abnormal records. Therefore, we use 20000 items of records (including the datasets generated by the WCGAN and the real training datasets), 20000 items of records (including the datasets generated by the GAN and the real training datasets), and the real training datasets to train the NN respectively. Then, we compare the loss function, the prediction accuracy and the false negative rate among the NN-WCGAN, the NN-GAN and the NN.

Fig. 5 shows the experimental results related to the WCGAN. Fig. 5 (a) shows the comparison of the loss function between the WCGAN and the GAN. After 12 iterations, the loss of the WCGAN approaches to 0, which indicates that the datasets generated by WCGAN are almost same as the real datasets, and the proposed WCGAN converges faster than the GAN. Additionally, Fig. 5 (b) shows the loss function of the NN in different training sets, after 12 iterations, the loss of NN-WCGAN is about 0.03, the NN-GAN is about 0.12 and the NN is about 0.15, which indicates that the NN-WCGAN converges faster. Similarly, Fig. 5 (c) shows the comparison of the prediction accuracy, compared with the NN, the NN-WCGAN and the NN-GAN have a higher accuracy. However due to multiple types of abnormal records in the training sets, the GAN cannot fit the distributions of abnormal records by using a single generator. Therefore, the false negative rate of NN-GAN is higher than the NN, as is shown in Fig. 5 (d). The WCGAN utilizes two generators to learn multiple distributions, increasing the similarity between the generated datasets ad the real datasets. Therefore, the false negative rate of the NN-WCGAN is similar to the NN. As a result, the datasets generated by the WCGAN could be used to train the NN, improving the performance of the NN.

C. The Performance Evaluation of RPBAC Model

The datasets of the experiments related to RPBAC model consist of different requests. Each request includes three attributes: the subject (the requesting node), the object (the accessed node), and the type of request. The subject is selected randomly. The selected nodes generate 10 rounds of requests, $R_1, R_2, \cdots, R_{10}$. The requests in each round come from 50 nodes. Each node initiates i requests, where $\mathrm{i} \in \{1, 2, ..., 10\}$. The type of request is composed of normal and malicious request alternately. We compare the number of correct access and the average response time among the RPBAC model, the RBAC model and the ABAC model to evaluate the effectiveness of the proposed RPBAC model.

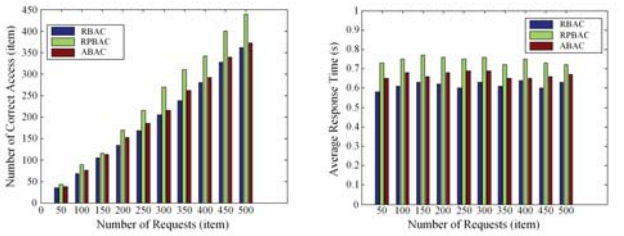
Fig. 6 shows the experimental results related to the RPBAC model, where Fig. 6 (a) shows the comparison of the number of correct control among the three models, the control accuracy



(a) Comparison of the loss function. (b) The loss function of NN.

(c) The prediction accuracy of NN. (d) The false negative rate of NN.

Fig. 5. The experimental results related to the WCGAN.



(a) The number of correct access. (b) The average response time.

Fig. 6. The experimental results related to the RPBAC model.

of the RBAC model is about 72%, the ABAC model is about 75%, and the RPBAC model is about 87%. Compared with the traditional static mechanisms, the proposed RPBAC adjusts the access rights dynamically by predicting the current risk level of vehicles, which improves the accuracy of access control. However, the RPBAC also needs more time to calculate the risk level, as shown in Fig. 6 (b). Fortunately, the increased time is relatively small to acceptance.

V. CONCLUSION

This paper proposes a novel access control model named RPBAC. By controlling the access rights of vehicles, the RPBAC model protects the safety of information sharing in the IoV. Moreover, we introduce the GAN in the risk prediction module to solve the problem of limited training sets. We also analyze the problem of pattern collapse and gradient disappearance in the traditional GAN, and develop an improved GAN, named WCGAN. To this end, compared with the traditional GAN, the experimental results show that our WCGAN performs a faster convergence speed. Moreover, the proposed WCGAN improves the prediction accuracy and reduces the false negative rate of the NN by increasing the items of training sets. Additionally, the proposed RPBAC model improves the accuracy of access control significantly. Future research includes exploring the WCGAN and the RPBAC model in a real-world deployment.

REFERENCES

[1] Chen, Shanzhi, et al. "Vehicle-to-everything (V2X) services supported by LTE-based systems and 5G." IEEE Communications Standards Magazine, vol.1, no. 2, pp. 70-76, 2017.

[2] J. Hu, S. Chen, Z. Li, Y. Li, J. Fang, B. Li, Y. Shi. "Link level performance comparison between LTE V2X and DSRC." Journal of Communications and Information Networks, pp. 101-112, 2017.

[3] S. Chen, J. Hu, Y. Shi, and L. Zhao, "LTE-V: A TD-LTE-Based V2X Solution for Future Vehicular Network," IEEE Internet of Things Journal, vol. 3, no. 6, pp. 9971005, 2016.

[4] J. Wang, C. Jiang, Z. Han, Y. Ren and L. Hanzo, "Internet of Vehicles: Sensing-Aided Transportation Information Collection and Diffusion," IEEE Transactions on Vehicular Technology, vol. 67, no. 5, pp. 3813-3825, May 2018.

[5] X. Cheng, R. Zhang, S. Chen, J. Li, L. Yang and H. Zhang, "5G enabled vehicular communications and networking," in China Communications, vol. 15, no. 7, pp. 3-6, July 2018.

[6] N. Sharma, N. Chauhan and N. Chand, "Security challenges in Internet of Vehicles (IoV) environment," First International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, pp. 203-207, 2018.

[7] D. Zhang, Y. Liu, L. Dai, A. Bashir, A. Nallanathan and B. Shim, "Performance Analysis of FD-NOMA-based Decentralized V2X Systems," IEEE Trans. Commun., vol. 67, no. 7, pp. 5024-5036, July 2019.

[8] Liqiang Qiao, Yan Shi, Shanzhi Chen and Wei Gao, "Modeling and Analysis of Safety Messages Propagation in Platoon-Based Vehicular Cyber-Physical Systems." Wireless Communications and Mobile Computing, vol. 2018, 2018.

[9] M. Ghafoorian, D. Abbasinezhad-Mood and H. Shakeri, "A Thorough Trust and Reputation Based RBAC Model for Secure Data Storage in the Cloud," in IEEE Transactions on Parallel and Distributed Systems, vol. 30, no. 4, pp. 778-788, 1 April 2019.

[10] Daniel Servos and Sylvia L. Osborn. "Current Research and Open Problems in Attribute-Based Access Control," ACM Computing. Surveys, vol. 49, no. 4, Article 65, January 2017.

[11] J. Weng, J. Weng, Y. Zhang, W. Luo and W. Lan, "BENBI: Scalable and Dynamic Access Control on the Northbound Interface of SDN-Based VANET," in IEEE Transactions on Vehicular Technology, vol. 68, no. 1, pp. 822-831, Jan. 2019.

[12] Y. Zhang, F. Qiao, I. Alsmadi and Q. Li, "Interactive based Access Control Framework for Connected Vehicle Communication," IEEE 14th International Conference on Control and Automation (ICCA), Anchorage, AK, pp. 393-398, 2018.

[13] H. A. Najada, I. Mahgoub and I. Mohammed, "Highway Cluster Density and Average Speed Prediction in Vehicular Ad Hoc Networks (VANETs)," IEEE Symposium Series on Computational Intelligence (SSCI), Bangalore, India, pp. 96-103, 2018.

[14] H. Zhao, T. Mao, H. Yu, M. K. Zhang and H. Zhu, "A Driving Risk Prediction Algorithm Based on PCA -BP Neural Network in Vehicular Communication," 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, pp. 164-169, 2018.

[15] Y. Sun, L. Peng, H. Li and M. Sun, "Exploration on Spatiotemporal Data Repairing of Parking Lots Based on Recurrent GANs," 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, pp. 467-472, 2018.

[16] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. arXiv preprint arXiv:1701.07875, 2017.

[17] Z. Zhang, M. Li and J. Yu, "D2PGGAN: Two Discriminators Used in Progressive Growing of GANS," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, United Kingdom, pp. 3177-3181, 2019.

[18] Grover J., Prajapati N.K., Laxmi V., Gaur M.S, "Machine learning approach for multiple misbehavior detection in VANET." International Conference on Advances in Computing and Communications Springer, Berlin, Heidelberg, 2011.

[19] MIT Lincoln Laboratory. KDD Cup 1999 Data, Available in http://kdd.ics.uci.edu/databases/kddcup99/.