Academic Education Publishing House

Industry Science and Engineering Vol. 1 No. 9, 2024

Collaborative Intrusion Detection in IoT by Integrating Blockchain and Semi-Supervised Learning

Ruozheng Wu[1], Zexiang Liu[1], Ziao Dong[1], Yanbing Liang[1,2,*]

[1]*College of Science, North China University of Science and Technology, Tangshan, Hebei, China*

[2]*Hebei Key Laboratory of Data Science and Application, North China University of Science and Technology, Tangshan, Hebei, China*

**Corresponding Author*

Abstract: With the proliferation of Internet of Things (IoT) devices, traditional intrusion detection systems (IDS) face challenges such as widespread data distribution and high labeling costs. This paper proposes a blockchain-based semi-supervised collaborative intrusion detection system (B-IDS) that integrates blockchain technology with a semi-supervised Generative Adversarial Network (SGAN) model to enhance intrusion detection performance in IoT environments. B-IDS leverages the decentralized, tamper-proof, and traceable characteristics of blockchain to achieve secure data sharing and collaborative detection. Additionally, it utilizes InterPlanetary File System (IPFS) for large-scale data storage, alleviating the data transmission bottleneck associated with blockchain. Experimental results demonstrate that B-IDS outperforms traditional models in accuracy, recall, and F1 score, effectively enhancing the system's detection accuracy and resilience against attacks. This system exhibits high adaptability and application potential, offering a new solution for intrusion detection in IoT.

Keywords: Blockchain; IoT; IDS; SGAN; Collaborative Detection; IPFS

1. Introduction

The rapid development of the Internet of Things (IoT) encompasses various types of devices, ranging from smart home appliances and industrial sensors to medical monitoring equipment. This large-scale, heterogeneous, and distributed network of devices generates significant data traffic and information exchange demands, posing new challenges for intrusion detection systems (IDS) in IoT [1]. Traditional IDS typically rely on centralized machine learning models; however, as the scale of IoT networks expands, such centralized approaches exhibit notable limitations in data scalability, cost-effectiveness, and response speed.

To address these challenges, a decentralized distributed architecture emerges as an ideal solution. As a decentralized database, blockchain has gained widespread application in various fields due to its tamper-proof, irreversible, and traceable characteristics [2]. In IoT environments, the integration of blockchain with IDS can effectively enhance the detection capabilities against network attacks and improve the stability and security of the system. Khan [3] et al. proposed a decentralized machine learning framework based on blockchain to enhance intrusion detection performance in drone swarms, facilitating intelligent decision-making among multiple drones. Alkadi [4] et al. introduced a Deep Blockchain Framework (DBF) that employs privacy-preserving blockchain and smart contracts for data migration and consensus across clouds, providing a secure distributed intrusion detection system based on privacy-oriented blockchain and smart contracts for IoT networks. Abdel-Basset [5] et al. proposed a Federated Intrusion Detection Framework (FED-IDS) based on joint deep learning, migrating the learning process from a central server to distributed vehicular edge nodes, leveraging blockchain's consensus mechanism to avoid model contamination. Additionally, the combination of blockchain technology with other distributed storage techniques in blockchain-based applications has demonstrated significant advantages. Li [6] et al. utilized InterPlanetary File System (IPFS) in a blockchain-based Collaborative Intrusion Detection System (CIDS) to facilitate the hosting and sharing of information among nodes for threat detection, malicious traffic filtering, and reduction of false positives.

Traditional IDS models often rely on feature extraction and rule matching, making them slow to react to novel and unknown attacks. Moreover, the dependence of traditional models on large amounts of labeled data complicates their training process and incurs high costs. In the context of IoT, the diversity of devices and the heterogeneity of data distribution further exacerbate the challenges of data labeling. To address these issues, the Semi-Supervised Generative Adversarial Network (SGAN) classification model, which integrates generative adversarial networks and semi-supervised learning, allows for network traffic classification tasks in the absence of extensive labeled data by utilizing data samples generated by the generator alongside a small amount of labeled data, thereby enhancing the model's generalization ability and classification accuracy [7]. Furthermore, leveraging the security features of blockchain technology, SGAN can ensure the security and privacy of data transmission, thereby enhancing the overall security of the system.

The primary contributions of this paper are as follows:

(1) Design of a Local Semi-Supervised Intrusion Detection Model: The model is designed to evaluate and analyze public datasets, conducting data preprocessing. The structure of the semi-supervised model effectively utilizes both labeled and unlabeled data for training, adjusting parameters to improve performance.

(2) Transmission of Anomalous Traffic and Intrusion Detection Models using Blockchain and IPFS: This ensures the security and transparency of data sharing, verification, and recording processes, while alleviating the data transmission bottleneck in traditional blockchain systems, thus achieving efficient distributed data transmission and storage.

(3) Organic Integration of Blockchain Technology with IoT Intrusion Detection: A comprehensive evaluation and verification phase is designed to compare the performance differences between the local intrusion detection model and the blockchain collaborative detection system, ensuring the stability and effectiveness of the system.

2. Related work

2.1 Blockchain

With advancements in information technology, the integration of blockchain technology with intrusion detection systems (IDS) has become a hot research topic. This integration aims to enhance the security and efficiency of IDS, protecting sensitive data from unauthorized access and facilitating collaboration among nodes in a decentralized architecture. Each node can independently perform monitoring tasks and record network activities, thereby achieving a global security situational awareness [8].

The core principle of blockchain lies in generating timestamped data blocks through cryptographic hash functions, which are permanently linked to form a distributed ledger. Once verified, the data is stored in a sequential manner on the chain, where every participant in the network can observe and verify the authenticity of the data blocks through a consensus mechanism. Once a data block is verified, it is irreversibly added to the blockchain, ensuring the integrity and irreversibility of the recorded data. The structure of blockchain mainly consists of block headers and transaction bodies. The block header includes key information such as version number, hash value of the previous block, Merkle tree root hash, timestamp, and nonce value, which guarantees the security and consistency of on-chain data. The structure of the blockchain is shown in **Figure 1**.

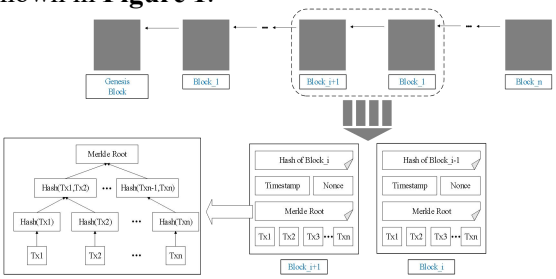

Figure 1. Structure of the Blockchain

Moreover, blockchain replaces traditional third-party intermediaries through consensus mechanisms, playing a key role in ensuring data authenticity. Common consensus algorithms include Proof of Work (PoW), Proof of Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT), which are widely applied in various blockchain systems. In intrusion detection scenarios, new threat information or detection results must be confirmed by the majority of nodes in the network before being recorded, thereby improving data reliability and preventing the dissemination of erroneous or malicious information.

2.2 IPFS

IPFS (InterPlanetary File System) [9] is a peer-to-peer (P2P) file transfer protocol and network based on distributed technology, designed to build a global distributed file system for efficient and secure data storage, retrieval, and sharing.

The core principles of IPFS include:

(1) Content Addressing: IPFS generates a unique identifier for files through content hashing, rather than relying on the traditional URL-based file location. This way, data integrity can be guaranteed by the content itself.

(2) Distributed Storage: IPFS splits files into multiple data blocks, distributed across nodes globally, thus improving data availability and redundancy.

(3) Peer-to-Peer Communication: IPFS enables data transfer through P2P communication, allowing direct data retrieval from nodes without reliance on centralized servers, thus enhancing transfer efficiency and security.

IPFS provides an innovative solution for distributed, highly available, and secure data management, showing significant advantages and broad prospects in decentralized applications.

2.3 GAN

Generative Adversarial Networks (GANs) are a deep learning model proposed by Ian Goodfellow[10] et al. in 2014, consisting of a generator and a discriminator. GANs generate realistic data samples through adversarial training of these two components. The generator is a deep neural network that takes a random vector (point in latent space) as input and outputs new samples similar to real data. Its goal is to generate realistic data to "fool" the discriminator into being unable to distinguish between real and generated data.

The discriminator receives data samples as input and outputs a scalar value indicating the probability that the input sample is real data. Its task is to distinguish between samples generated by the generator and real samples, thereby enhancing its ability to identify real and fake data[11]. In adversarial training, the generator tries to produce more realistic data, while the discriminator strives to accurately identify the differences between generated and real data. An adversarial loss function is used during training, where the generator's loss is the probability that the discriminator misclassifies the generated data, while the discriminator's loss is the sum of the probabilities of correctly classifying real and generated data. This adversarial process continually optimizes both the generator and the discriminator until the generator can produce data samples that are nearly indistinguishable from real data. The structure of GANs is shown in **Figure 2**.

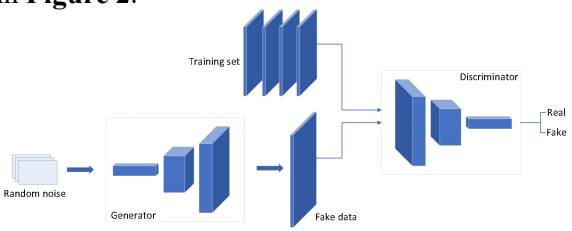

Figure 2. Structure of GAN

3. Method

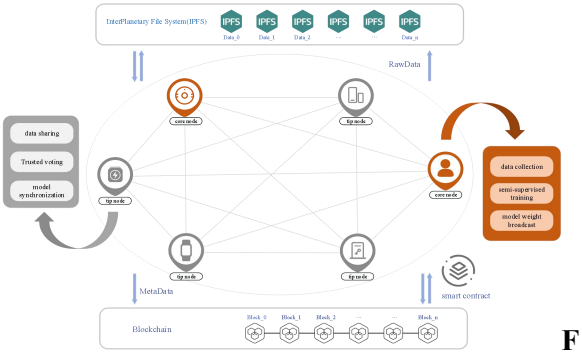

Figure 3. Blockchain-based Collaborative Intrusion Detection System

This section introduces the use of a blockchain-based collaborative intrusion detection system (B-IDS) for detecting network attacks and protecting the Internet of Things (IoT). The proposed system primarily consists of three main components: IoT nodes, blockchain with smart contracts, and IPFS. As shown in **Figure 3**, different types of IoT nodes are the basic building blocks of the system. They are identified as IoT entities within the blockchain network, distributed across various devices, including sensors, smart home devices, and industrial control systems. The blockchain, functioning as a decentralized distributed database, is responsible for storing and sharing information related to intrusion detection. Additionally, the blockchain supports the development of smart contracts, which can automatically enforce data usage policies, such as determining which nodes are permitted to send anomalous traffic alerts or whether to accept newly trained models.

3.1 IoT Nodes

In the blockchain-based collaborative intrusion detection system, IoT nodes select to register as

endpoint or core nodes based on their computational capabilities before joining the blockchain and undergo identity verification. Devices need to invoke the identity management contract to create node identities. Each node randomly generates a 32-byte binary number as its secret key (SEC-KEY) and calculates the public key (PUB-KEY) using the Elliptic Curve Digital Signature Algorithm (ECDSA). The contract assigns a Node_ID to each node for device identification. The public and private keys generated through this algorithm ensure the security of information transmission between nodes. Only authenticated nodes can participate in the collaborative intrusion detection system and engage in the consensus of the blockchain.

Each node is equipped with the following key components:(1)IDS Component: Responsible for monitoring network traffic and detecting anomalous traffic to identify potential intrusion behavior.(2)Blockchain Component: Facilitates the sharing of metadata between nodes and participates in the consensus process of the blockchain to achieve collective decision-making.(3) IPFS Component: Provides decentralized file storage capabilities for storing and transmitting raw data related to anomalous traffic and intrusion detection models between nodes.(4) Training Component: Core nodes with sufficient computational power are required to perform model training when updates are needed.

3.2 Data Sharing and Model Updates

Due to the need to strictly control energy consumption on the blockchain, it is essential to minimize information processing and data storage on-chain to avoid increasing the burden on participating blockchain nodes. Therefore, when sharing anomalous traffic or models between nodes, IPFS is used to store and share the raw data. The hash value of the raw data serves as a unique index and is uploaded to the blockchain along with metadata such as timestamps and version information to ensure data security and traceability. The data sharing and storage process primarily includes the following stages:

(1) The IoT node n_i organizes the raw data of abnormal traffic detected by the IDS over a specified period into a data packet P . After encrypting P with its SEC-KEY, n_i stores P in the IPFS using content addressing, where IPFS generates a unique hash value P_{hash} that serves as the unique identifier for P .

(2) As the initiating node, n_i uses its SEC-KEY to digitally sign the metadata (MetaData) containing P_{hash} , a timestamp, version information, and other data, thereby generating a data transmission certificate (Token). Subsequently, n_i sends the (MetaData, Token) to the full node N .

(3) Upon receiving the (MetaData, Token) from n_i , N first verifies the validity of the digital signature to ensure that the message originates from a legitimate initiating node. Next, N decrypts the Token using n_i 's public key and compares the parsed data with the MetaData. After confirming the integrity and accuracy of the data, N retrieves P_{hash} through the IPFS system to further validate its correspondence with the MetaData.

(4) Once confirmation is complete, N transmits the (MetaData, Token) to the data-sharing contract and returns a signed message to n_i containing the transaction address for executing the data-sharing contract. The data-sharing contract parses the information, which includes the node identification information (Storage_Node_ID field), and then waits for n_i to send information to the current transaction. The contract checks whether the signature information matches the identifiers in the (MetaData, Token). If the match is successful, the contract instructs all nodes to package the metadata into the transaction and publish it, thereby storing the MetaData on the blockchain.

When a certain node needs to read the abnormal traffic information, it can locate P in the IPFS using P_{hash} from the MetaData and then decrypt the original abnormal traffic data using n_i 's PUB_KEY. This process effectively utilizes the security features of blockchain while efficiently mitigating its energy consumption issues.

In B-IDS, when a decline in model performance or changes in the data environment are detected, the system triggers the model update process. Through the Proof of Work (PoW) consensus mechanism, a suitable training node is selected from the core node cluster. The training node executes semi-supervised training using the new

traffic data to adjust and optimize the intrusion detection model. Upon completion, the training node uploads the new model and related evaluation metrics to IPFS and executes the data-sharing process. Nodes within the blockchain network validate the model's performance and use smart contracts to determine whether to accept the new model update. If the new model is approved, it will be formally recorded on the blockchain; otherwise, the system will initiate an iterative process to select a new training node and repeat the above steps until the model is successfully updated.

3.3 Semi-Supervised IDS Model

Due to the massive scale of IoT traffic data, its complex distribution, and data imbalance, traditional supervised learning models often suffer from overfitting or underfitting issues, and it is impractical to fully annotate such vast amounts of data in reality. In this context, Semi-Supervised Generative Adversarial Networks (SGAN) provide an effective solution[12]. The SGAN model combines Generative Adversarial Networks with semi-supervised learning, enabling network traffic classification using the data samples generated by the generator alongside a limited amount of labeled data. This significantly enhances the model's generalization capability and classification accuracy.

Unlike traditional GANs, SGAN converts the binary classification output of the discriminator network (Sigmoid) into a multi-class output (Softmax), achieving outputs for $N + 1$ categories, where N represents the labeled classes, and one class is designated for "fake data." After this conversion, the discriminator D can distinguish among [CLASS-1, CLASS-2, ..., CLASS-N, FAKE], thereby simultaneously functioning as a classifier (C), referred to as the D/C network.

The training process of SGAN is similar to that of GAN but only uses labeled data from a portion of the data batches for training. The D/C network optimizes predictions for real samples by minimizing negative log-likelihood, while the generator G improves the quality of generated samples by maximizing this likelihood. In B-IDS, the architecture of the SGAN generator consists of six transpose convolutional layers (Conv2DTranspose), with interspersed batch normalization (BatchNormalization) and LeakyReLU activation functions, ultimately using a tanh activation function to output a tensor matching the input image size. The discriminator network comprises five convolutional layers (Conv2D) combined with batch normalization, LeakyReLU, and dropout layers to prevent overfitting. Finally, a fully connected layer outputs multiple class nodes and, under the Softmax activation function, yields classification probabilities, completing the supervised learning process for labeled data. The structure of the model is illustrated in **Figure 4**.

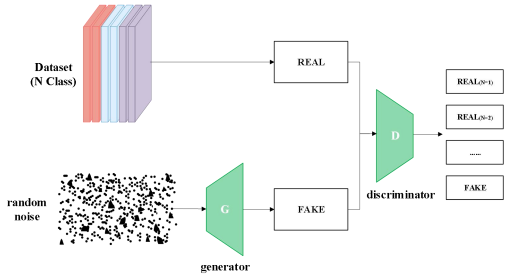

Figure 4. SGAN Network Structure

4. Experiment

4.1 Dataset Analysis and Processing

CIC-IDS2018 is a publicly available dataset designed for intrusion detection research, created by the Canadian Institute for Cybersecurity. Due to the characteristics of certain fundamental features in the dataset, such as "Timestamp," "Protocol," "PSH Flag Cnt," "Init Fwd Win Byts," "Flow Byts/s," and "Flow Pkts/s," which are indicative of network identities, these six basic features are removed from the dataset to prevent overfitting in the model.

For the processed daily datasets, standardization is necessary before conducting experiments, which ensures that the data is uniformly mapped to the range of [0, 1]. The specific expression for data normalization is given by Equation (1), Where X_{norm} represents the normalized value, X_{min} denotes the minimum value of the feature, and X_{max} indicates the maximum value of the feature.

$$X_{norm} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{1}$$

In order to input numerical text data into the SGAN algorithm, the CIC-IDS2018 dataset is preprocessed to yield a clean dataset, consisting of single-row numerical data. This numerical data is then transformed into a two-dimensional matrix format. Based on the number of features in the dataset, each traffic data instance is converted into a 28×28 matrix. Zero-padding is

employed, and the feature data is evenly distributed throughout the feature matrix at equal intervals, with each row of data corresponding to one feature matrix. **Figure 5** illustrates the structural representation of the data after two-dimensional conversion.

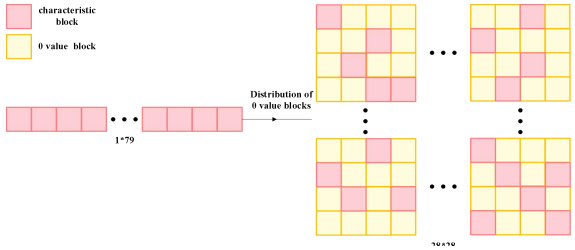

Figure 5. Data Matrix Transformation Flowchart

The dataset consists of a total of 15 categories of data collected over different dates, where one type represents normal traffic and the other 14 types represent various network attack categories. In the experiment, the training set comprising 40%, the validation set 10%, and the remaining data allocated for simulation.

4.2 Experimental Design

The experiments were designed and implemented in a TensorFlow environment using Python. All experiments were conducted on a Windows 10 operating platform, equipped with an Intel Core i7-10850K processor, 64GB of RAM, and an NVIDIA GeForce RTX 3090 graphics card. Ganache and Truffle were utilized to simulate 10 IoT node blockchain accounts, with the nodes connecting to IPFS through the ipfshttpclient library.

To evaluate the efficiency and security of the proposed B-IDS, a local private blockchain was created using Truffle and Ganache[13], which seamlessly integrates to test the functionalities and interactions of smart contracts within the development environment. Initially, blockchain accounts were established, which were treated as IoT nodes connecting to our simulated private blockchain. The Web3 library was employed to manage these accounts, connecting nodes to the blockchain using private keys. The training hyperparameters in the experiments were set as follows: num_labeled = 0.1, z_dim = 0.1, iterations = 1000, batch_size = 32, sample_interval = 50, and learning_rate = 0.01.

4.3 Experimental Results and Analysis

Initially, the dataset was used for local model training and testing, where the SGAN network traffic classification model was compared with other traditional classification models using 10% of the labeled data. The selected dataset for the experiments was the processed CIC-IDS 2018 dataset. The accuracy, recall, and F1 scores of each model are presented in **Table 1**.

The experimental results show that the SGAN model achieved a classification accuracy of 93% using only 10% of the labeled data, significantly reducing the demand for the training dataset. The classification performance of the SGAN model improved by approximately 6% compared to the traditional C4.5 machine learning algorithm, and by about 10% compared to the 2D CNN network. Overall, these results confirm that this approach can be used to create classifiers that are more data-efficient while maintaining good accuracy.

Table 1. Performance Comparison of lassification Models

Classification Model	accuracy	recall	precision	F1-score
1D CNN	0.8503	0.85	0.85	0.85
2D CNN	0.8331	0.83	0.83	0.83
C4.5	0.8679	0.86	0.86	0.86
DCGAN(10%)	0.8962	0.89	0.89	0.89
SGAN(10%)	**0.9307**	**0.93**	**0.93**	**0.93**

Based on the data division, half of the total data was used to train the local SGAN model, while the remaining data was distributed to the nodes in a 1:1 ratio of anomalous to normal traffic to simulate data requests from IoT devices. The shared data on the blockchain and IPFS facilitated the collaborative training and credible evaluation of B-IDS, allowing for the selection of well-performing models to be updated across all nodes. The performance of the model was validated over 50 rounds using the F1 score and accuracy as reference metrics. We observed that the F1 score gradually increased, ultimately approaching 0.95. With the increase in update rounds, the accuracy of the well-performing nodes also fluctuated upwards. The trends in model accuracy and F1 score are illustrated in **Figure 6**.

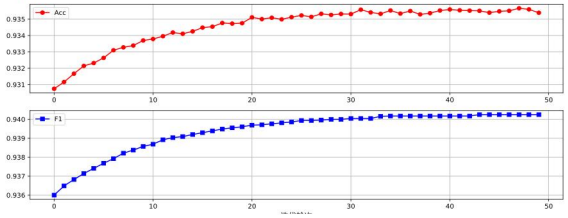

Figure 6. Changes in F1 and Accuracy with Update Rounds

The B-IDS model was compared with the local intrusion detection model and traditional intrusion detection models, with their accuracy, recall, and F1 scores summarized in **Table 2**.

Table 2. Comparison of Proposed Model and Related Research Model Performance

Classification Model	accuracy	recall	precision	F1-score
SGAN(10%)	0.9307	0.93	0.93	0.93
B-IDS(10%)	0.9351	0.93	0.94	0.93

The experimental results demonstrate that the IoT intrusion detection model combined with blockchain can enhance the model's performance in accuracy, recall, and F1 score through decentralized data storage and verification, immutable data records, the application of smart contracts, as well as data sharing and collaboration mechanisms. The distributed nature of blockchain and its consensus mechanism enable the model to comprehensively collect and verify feedback information from nodes, thereby improving the accuracy and robustness of intrusion detection. Furthermore, the proposed blockchain-based IoT intrusion detection model, B-IDS, exhibits the capability to quickly adapt to new tasks, making it suitable for various network conditions and access methods. This characteristic holds significant practical value for network intrusion detection.

5. Conclusion

The blockchain-based IoT intrusion detection approach addresses the issues of data security and trustworthiness, as well as the challenges of data sharing and collaboration encountered in traditional IoT intrusion detection. The combination of blockchain and IPFS effectively resolves the data transmission bottleneck that blockchain faces in practical applications. The proposed B-IDS model leverages the decentralized data storage and immutable data record features of blockchain technology to ensure the security and reliability of data generated by IoT devices during storage and transmission. Additionally, by utilizing a semi-supervised classification model as the foundational classification model, the labeling cost is significantly reduced. Moreover, the data sharing and collaboration mechanisms within the blockchain network facilitate a better understanding of the overall operational status and environment of the IoT system, thus allowing for more effective detection of anomalous behavior.

References

[1] Shen X, Lu Y, Zhang Y, et al. An Innovative Data Integrity Verification Scheme in the Internet of Things assisted information exchange in transportation systems[J]. Cluster Computing, 2022, 25(3): 1791-1803.

[2] Ali S, Li Q, Yousafzai A. Blockchain and federated learning-based intrusion detection approaches for edge-enabled industrial IoT networks: A survey[J]. Ad Hoc Networks, 2024, 152: 103320.

[3] Khan A A, Khan M M, Khan K M, et al. A blockchain-based decentralized machine learning framework for collaborative intrusion detection within UAVs[J]. Computer Networks, 2021, 196: 108217.

[4] Alkadi O, Moustafa N, Turnbull B, et al. A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks[J]. IEEE Internet of Things Journal, 2021: 9463-9472.

[5] Abdel-Basset M, Moustafa N, Hawash H, et al. Federated intrusion detection in blockchain-based smart transportation systems[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 23(3): 2523-2537.

[6] Li W, Stidsen C, Adam T. A blockchain-assisted security management framework for collaborative intrusion detection in smart cities[J]. Computers and Electrical Engineering, 2023, 111: 108884.

[7] Sajun A R, Zualkernan I. Survey on implementations of generative adversarial networks for semi-supervised learning[J]. Applied Sciences, 2022, 12(3): 1718.

[8] Li W, Wang Y, Li J, et al. Toward a blockchain-based framework for challenge-based collaborative intrusion detection[J]. International Journal of Information Security, 2021, 20: 127-139.

[9] Kumar S, Bharti A K, Amin R. Decentralized secure storage of medical records using Blockchain and IPFS: A comparative analysis with future directions[J]. Security and Privacy, 2021, 4(5): e162.

[10] Aggarwal A, Mittal M, Battineni G. Generative adversarial network: An overview of theory and applications[J]. International Journal of Information Management Data Insights, 2021, 1(1): 100004.

[11] Kuntalp M, Düzyel O. A new method for GAN-based data augmentation for classes with distinct clusters[J]. Expert Systems with Applications, 2024, 235: 121199.

[12] Aldhaheri S, Alhuzali A. SGAN-IDS: self-attention-based generative adversarial network against intrusion detection systems[J]. Sensors, 2023, 23(18): 7796.
[13] Verma R, Dhanda N, Nagar V. Application of truffle suite in a blockchain environment[C]//Proceedings of Third International Conference on Computing, Communications, and Cyber-Security: IC4S 2021. Singapore: Springer Nature Singapore, 2022: 693-702.