Adaptive Privacy-Preserving Framework for Network Traffic Anomaly Detection*

Yu-ran Jeon, Seung-ha Jee, Su-Kyoung Kim, and Il-Gu Lee†

Sungshin Women's University, Seoul, Korea
cseyrj@gmail.com, {220256039, 220254012, iglee}@sungshin.ac.kr

Abstract

As cyberattacks become more sophisticated and intelligent, research on anomaly detection systems that support strong privacy protection is being actively conducted. A conventional privacy-preserving anomaly-detection system receives encrypted data as input and detects anomalies with homomorphic encryption. However, there are limitations in that the latency increases significantly during the detection process, and the detection accuracy decreases. To solve this problem, we propose an adaptive privacy-preserving anomaly detection (APPAD) model that adaptively performs homomorphic operations. The APPAD model processes incoming traffic as plaintext or ciphertext depending on the sensitivity of the traffic and performs anomaly detection through homomorphic encryption only on encrypted traffic. Experimental results in various network environments show that the proposed model improved accuracy by up to 73%, reduced latency by 8.6 times, and showed negligible information leakage compared to conventional privacy-preserving anomaly detection models.

Keywords: Homomorphic encryption; machine learning; privacy-preserving anomaly detection

1 Introduction

With recent advancements in digital technology, cyberattack techniques have become increasingly sophisticated, highlighting the importance of anomaly detection systems for detecting these attacks. An anomaly detection system trains normal patterns from the given data and then detects values that deviate significantly from the data distribution, thereby identifying abnormal or potentially dangerous events [1]. This technology has been widely utilized in areas such as the detection of abnormal transactions in the financial sector [2], disease outbreak patterns in the healthcare sector [3], and internal intrusions in corporate systems [4].

However, conventional anomaly-detection systems have been designed based on the assumption that servers are trustworthy, with data collected in plaintext on servers and then used in the learning and detection processes. Although this structure is simple to implement, it has a fundamental limitation in that the data are exposed to the server [5]. In particular, sensitive information such as financial transaction histories, medical records, and location information can be leaked by attackers or misused in cyberattacks. Therefore, as the zero-trust security paradigm is gaining attention, privacy-preserving anomaly detection systems have been actively studied [6]. Homomorphic encryption is a cryptographic

* Proceedings of the 9th International Conference on Mobile Internet Security (MobiSec'25), Article No. 15, December 16-18, 2025, Sapporo , Japan. \space © The copyright of this paper remains with the author(s).
† Corresponding author

technique that allows arithmetic operations to be performed directly on the ciphertext, ensuring that the decrypted result is identical to the result obtained by applying the same operation to the plaintext [7]. Anomaly detection with homomorphic encryption allows secure learning in encrypted form without exposing sensitive information to the server as plaintext. However, the process of calculating encrypted data requires a considerably larger amount of computation than general operations, which increases latency and reduces detection accuracy because of the noise generated during the complex calculation process. Although homomorphic encryption preserves privacy, it introduces considerable latency in anomaly detection scenarios that require real-time performance.

To address this tradeoff, this study proposes the adaptive privacy-preserving anomaly detection (APPAD) model, an anomaly detection method that dynamically applies homomorphic encryption based on the privacy sensitivity of the data. The APPAD balances privacy and latency by performing homomorphic operations only when necessary, considering traffic sensitivity.

The contributions of this study are as follows:

- An anomaly detection system that adaptively performs homomorphic computation is proposed to address the privacy issues of conventional plaintext-based anomaly-detection systems, and the delay and accuracy issues of privacy-preserving anomaly-detection systems.
- The trade-off between delay and privacy issues was improved by classifying network traffic types according to their privacy sensitivity and determining anomaly detection methods based on the traffic type.
- Compared to conventional anomaly-detection models, we demonstrated efficient anomaly detection by improving accuracy by up to 73%, reducing delay by 8.6 times, and achieving low information leakage.

The remainder of the study is organized as follows. Section 2 analyzes research related to machine-learning-based anomaly detection and privacy-preserving anomaly detection. Section 3 proposes a technique that dynamically applies homomorphic encryption based on data-privacy sensitivity. Section 4 analyzes the performance evaluation results of the proposed and conventional approaches. Finally, Section 5 concludes the paper.

2 Related work

This section analyzes previous studies on machine learning-based anomaly-detection and privacy-preserving anomaly-detection technologies, as listed in Table 1. Machine learning-based anomaly detection technology receives plaintext as input and performs learning and inference, whereas privacy-preserving anomaly-detection technology receives homomorphic encrypted data as input and performs anomaly detection using homomorphic encryption.

Table 1. Comparison of previous studies

Categories	Ref.	Contribution	Limitation
Machine learning-based anomaly detection	[8]	· Two fuzzy anomaly detection techniques using unsupervised and supervised learning were proposed to address the boundary uncertainty problem.	· Information Granules (IG) technology has been applied to address the uncertainty in anomaly data detection (ADD); however, using ADD as the primary

		· It is more efficient with lower computational complexity than kernel learning-based OC-SVM and SVDD models, since it directly uses data structure information.	optimization technique is not efficient. · Data security is not considered since plaintext data is used for learning.
	[9]	· An anomaly detection technique based on system text logs for virtual network functions was proposed. · It is possible to predict abnormal signs 35 minutes before the actual failure caused by the attack occurs.	· A normal log may be misclassified as abnormal if it contains patterns that deviate from the expected behavior.
Privacy-preserving anomaly detection	[10]	· A polynomial technique optimized for Fully Homomorphic Encryption (FHE) by synthesizing multiple low-degree polynomials was proposed. · Demonstrated similar performance to the original model without separate retraining.	· Privacy-preserving machine learning (PPML) increases latency and is difficult to apply to real environments.
	[11]	· A real-time intrusion detection system that performs distributed learning on homomorphic encrypted data was proposed · The latency of homomorphic encrypted data was reduced through distributed learning.	· It reduces latency compared to the conventional centralized homomorphic encrypted data learning but has greater latency than general plaintext-based machine learning.

Ouyang et al. [8] proposed a technique to solve the anomaly data detection (ADD) uncertainty problem by utilizing fuzzy theory and particle computing in the ADD process to solve the dataset imbalance problem. The detection performance was improved by optimizing the anomaly detection boundary using the information granules (IG) technique. However, it remains unclear whether the IG technique is the most optimal approach for ADD, and the risk of information leakage from the use of plaintext data during model training has not yet been considered. Rim et al. [9] proposed a technique for detecting abnormal signs before system failure occurs by analyzing text logs generated from virtualized network functions (VNF). The proposed method vectorizes the log data and merges

duplicate data from the logs generated in the same process into a single vector. A long short-term memory (LSTM) was then trained on normal data to derive the predicted values. The error between the actual and predicted values was then calculated to classify logs with large errors as anomalies. This allows the prediction of abnormal signs in advance; however, merging similar logs into a single vector increases the error range of the data classification criterion value, making it difficult to classify normal and abnormal data in detail.

Lee et al. [10] proposed a polynomial technique that approximates ReLU and Max-pooling functions to enable HE-based data learning to solve the problems of high model latency and bootstrapping errors caused by fully homomorphic encryption (FHE). In conventional methods utilizing HE data, the network of a predefined model should either be redefined or retrained. Contrarily, the proposed technique eliminates the need for retraining or additional model design. However, the proposed model suffers from significant latency, with a maximum inference time of 4,764 s per image. Therefore, optimization is required to apply the proposed approximate deep learning model in a real-world environment. Manh et al. [11] applied HE to protect data in a blockchain environment, and proposed a privacy-preserving distributed learning (PPDiL) technique to improve model latency. The proposed technique distributes HE-applied data from a central server to multiple worker nodes and learns from each node. The central server then aggregates the learning results from these nodes to create a single optimized model to detect and block cyberattacks in the network traffic. Although the latency of conventional homomorphic encrypted data learning was improved through the distributed learning method, the proposed technique also exhibited a latency of at least 21.64 h, which is higher than that of general machine learning. Although homomorphic-encryption-based models achieve improved accuracy, they still suffer from significant latency compared with plaintext-based models. To address this issue, this study proposes an anomaly-detection technique that dynamically applies homomorphic encryption.

3 Adaptive Privacy-Preserving Anomaly Detection

This section describes the operational structure and principles of the proposed APPAD model. In the proposed APPAD model, the server collects and learns the network traffic generated when a user uses an application or a web service. It then uses real-time traffic as input for anomaly detection. When a user uses a service, traffic is generated during various behaviors such as searching, surfing, logging in, and data transmission, and some of this traffic contains sensitive data such as personal or authentication information. However, conventional anomaly-detection research assumes that anomaly detection is performed when traffic is input in plaintext, failing to consider data sensitivity. Recently, research on anomaly detection for encrypted data has been proposed; however, these privacy-preserving anomaly detection models are slow and therefore unsuitable for real-world application environments. To address the privacy and latency limitations of conventional anomaly-detection methods, this study proposes an anomaly detection model that adaptively performs homomorphic operations based on data sensitivity. The scenario for the proposed model is shown in Fig. 1. The traffic generated by the user actions is processed by the application, and the processed traffic is transmitted to the server to detect traffic anomalies.

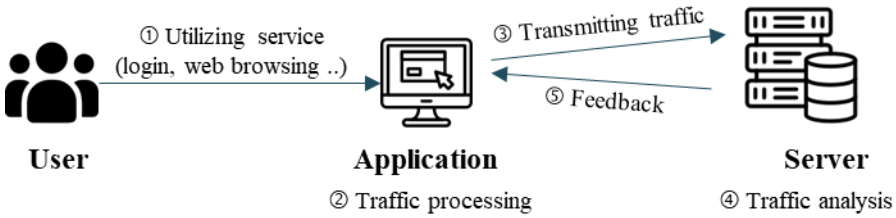

Figure 1: The scenario of the proposed model

3.1 Traffic processing

When using application services, users perform actions such as account login and web browsing, and the application collects traffic generated from the user's actions in real time. Traffic is encrypted based on privacy sensitivity. If the traffic contains private information, it is classified as privacy-sensitive traffic; if it does not contain private information, it is classified as non-sensitive traffic. APPAD performs anomaly detection using homomorphic operations when receiving privacy-sensitive traffic as input and plaintext operations when receiving non-sensitive traffic as input.

3.2 Traffic analysis

The traffic processed by the application according to predefined rules is sent to the anomaly detection server. The server verifies the traffic received from the application and determines whether homomorphic encryption should be applied to the traffic. Homomorphic computation is used to detect anomalies when homomorphic encrypted traffic is received as input. When plaintext traffic is received as input, plaintext computation is used to detect anomalies. The anomaly detection server is equipped with an anomaly detection model pretrained on plaintext traffic. When homomorphic encrypted traffic is input into an anomaly detection model, the pre-trained model outputs the encrypted results through homomorphic operations and determines whether the data are anomalous based on the decrypted results.

A logistic regression model was used as the learning and classification model for traffic anomaly detection. The logistic regression model predicts classes in the form of integer values, and performs classification based on the probability that the data belong to that class. That is, a logistic regression model can directly estimate the probability of normal or abnormal traffic, thereby enabling anomaly detection. When training the logistic regression model, the epoch was set to 10, and the threshold for determining abnormal and normal values when inferring the model was set to 0.5.

4 Evaluation

In this section, we compare the performance of the proposed and conventional models in various network environments. To simulate a network traffic collection environment, we used the Web_Auth_Anomaly_Detection dataset [12]. This dataset is synthetic data generated based on user authentication logs from web applications, and contains traffic related to normal and abnormal login events. The dataset is composed of nine features: USER ID', Timestamp', Login Status', IP Address', Device Type', Location', Session Duration', Failed Attempts', and 'Behavioral Score, which include the characteristics of login attempts.

The proposed model randomly receives plaintext and ciphertext traffic as inputs, and dynamically detects anomalies based on the type of incoming traffic. Conventional models receive only plaintext or

ciphertext traffic as input and perform plaintext- and ciphertext-based anomaly detection. For sensitive traffic, homomorphic encryption based on the TenSEAL library was applied to preserve privacy and anomaly detection was performed using homomorphic operations [13]. The TenSEAL library is a Microsoft SEAL-based Python library that can efficiently perform tasks, such as machine learning and inference, using data encrypted with the CKKS scheme.

To verify the performance of the proposed model in various network environments, we evaluated its accuracy, latency, information leakage, and efficiency based on the data sampling ratio and proportion of privacy-sensitive data. The model accuracy and latency were measured by determining the inference accuracy and inference time when new data were input into the trained model.

For information leakage, if traffic containing privacy-sensitive information was input into the anomaly-detection model and the model performed plaintext-based anomaly detection, the information was determined to have been leaked. Information leakage is evaluated as the amount of privacy-sensitive traffic leaked out of the total traffic. Eq. (1) presents a formula for evaluating information leakage (L_t). N represents the total traffic volume, p represents the proportion of privacy-sensitive traffic to total traffic, and l represents the size of a single traffic unit. Information leakage is calculated by multiplying the amount of traffic leaked by the size of the traffic.

$$L_t\ (bits) = N\ \times p\ \times l \tag{1}$$

The detection efficiency of the model (M_{eff}) was evaluated by comprehensively considering the anomaly detection accuracy (Acc), latency (d), and information leakage (L_t). The higher the detection accuracy and the lower the latency and information leakage, the more efficiently the model can perform anomaly detection. The efficiency of the model was calculated using Eq. (2), L_t is assumed to be nonzero.

$$M_{eff} = \frac{Acc}{d\ \times L_t} \tag{2}$$

Fig. 2 and 3 show graphs measuring the accuracy and latency according to the data sampling ratio. Non-privacy-preserving anomaly detection (NPPAD) is a conventional model that performs plaintext-based anomaly detection, whereas privacy-preserving anomaly detection (PPAD) is a conventional model that performs ciphertext-based anomaly detection. The proportion of privacy-sensitive data (p) in the dataset was set at 0.5.

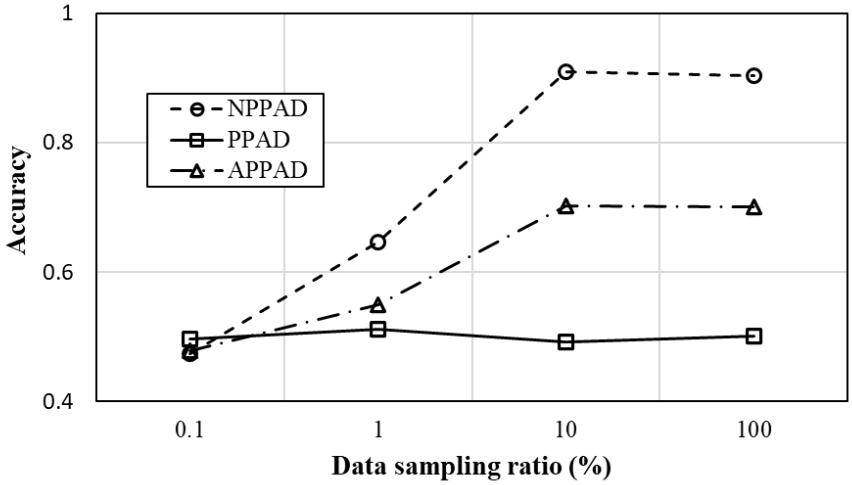

Figure 2: Accuracy vs. data sampling ratio

As the data sampling ratio increased, the amount of training data increased, thereby improving the accuracy of the NPPAD and APPAD models. However, the PPAD model, which performs anomaly detection with homomorphic encryption, maintains low accuracy regardless of the data volume. The proposed APPAD model maintained 70.2% accuracy even in data-sparse environments because it performs homomorphic computations only on privacy-sensitive traffic. While APPAD exhibits lower accuracy than NPPAD, it improves the accuracy by 42% compared to the PPAD model.

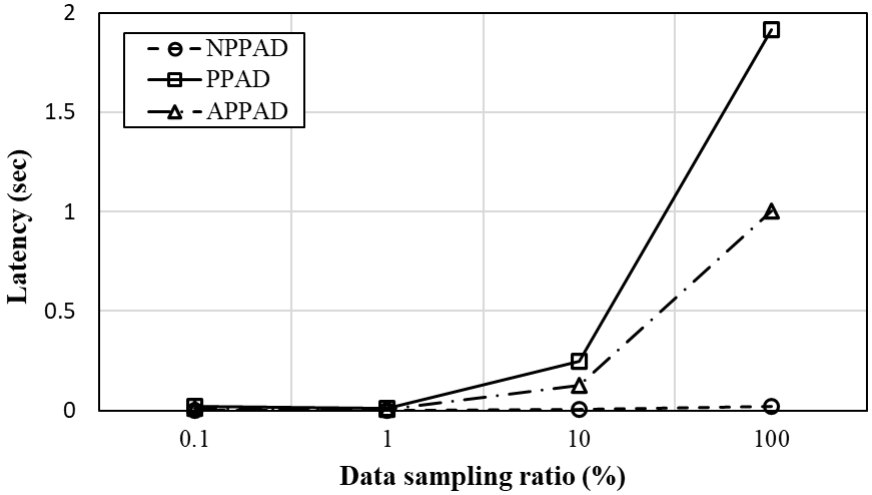

Figure 3: Latency vs. data sampling ratio

As the data sampling ratio increases, more traffic is input, slowing down traffic processing. PPAD, which performs anomaly detection with homomorphic encryption, experiences exponentially increasing latency as the data volume increases. In environments with excessive traffic, this incurs a latency 90.8% higher than that of the proposed APPAD model. That is, the proposed APPAD model demonstrates that adaptively performing homomorphic computation can significantly improve accuracy compared to PPAD and reduce latency compared to NPPAD.

We evaluated the proposed model in an environment where diverse types of traffic coexist, allowing us to better approximate real-world network conditions. Fig. 4-7 presents the results of evaluating the accuracy, latency, information leakage, and efficiency as the proportion of privacy-sensitive data increased.

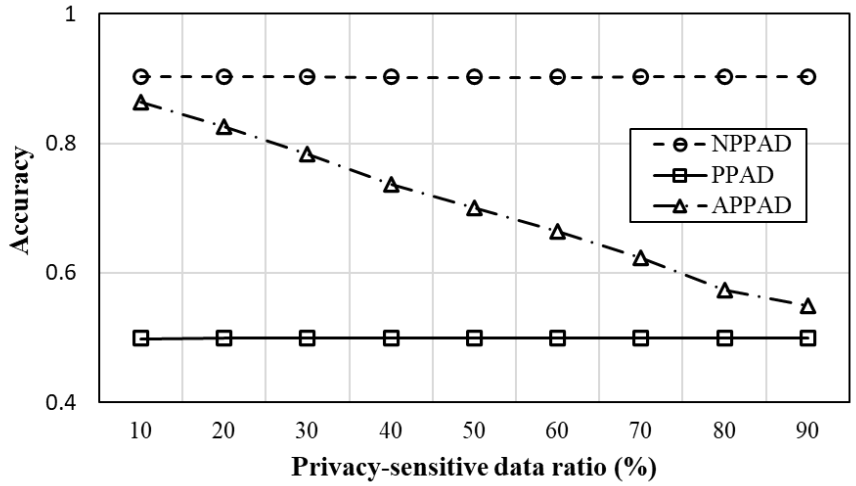

Figure 4: Accuracy vs. privacy-sensitive data ratio

As conventional models process traffic regardless of the data type, NPPAD maintains 90% accuracy, whereas PPAD maintains 50% accuracy. The APPAD processes traffic considering its privacy sensitivity. As the proportion of privacy-sensitive data increases, the number of homomorphic operations increases, leading to a decrease in accuracy.

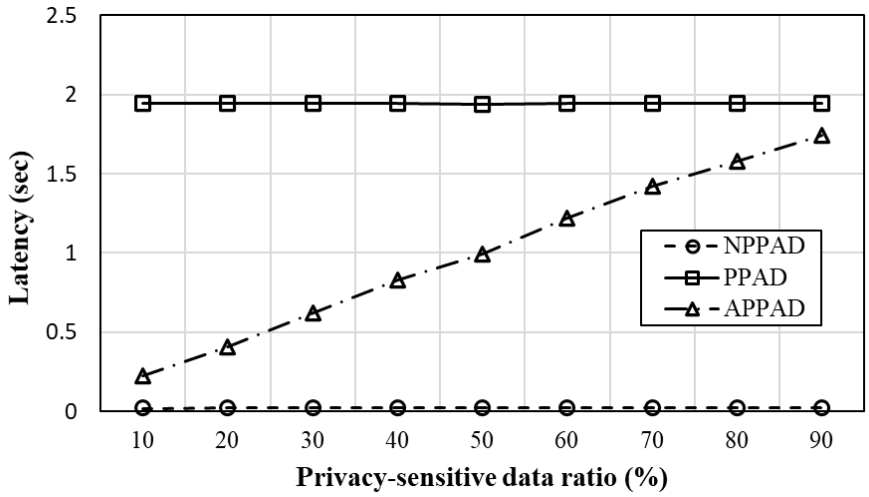

Figure 5: Latency vs. privacy-sensitive data ratio

The latency was the highest in PPAD, followed by APPAD and NPPAD. The PPAD model always performs anomaly detection with homomorphic encryption regardless of the data type, resulting in significant latency. Contrarily, the proposed APPAD model performs a homomorphic operation only when traffic containing privacy-sensitive information is the input. Therefore, it exhibits a latency similar to that of NPPAD when the proportion of privacy-sensitive data is low.

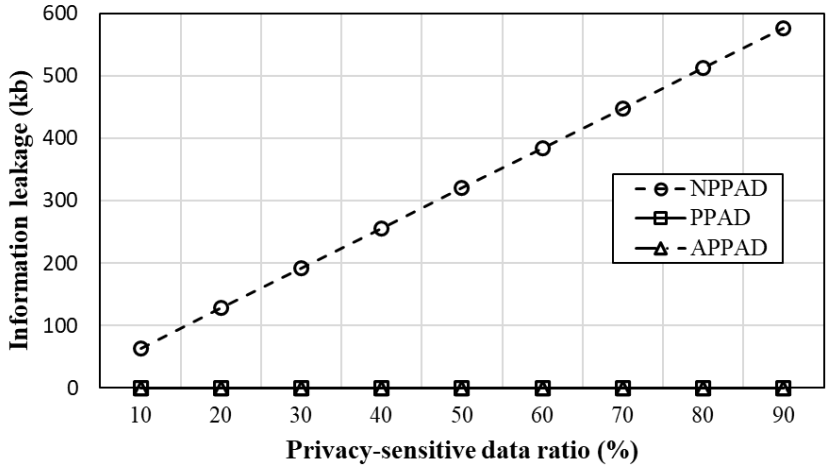

Figure 6: Information leakage vs. privacy-sensitive data ratio

In terms of information leakage, NPPAD showed increased information leakage as the proportion of privacy-sensitive data increased, while PPAD and APPAD showed almost no information leakage. The proposed APPAD and PPAD models homomorphically process privacy-sensitive traffic, thereby enabling privacy preservation. Experimental results demonstrate that the proposed APPAD model significantly reduces information leakage compared to the NPPAD model, while improving latency and accuracy compared to the PPAD model.

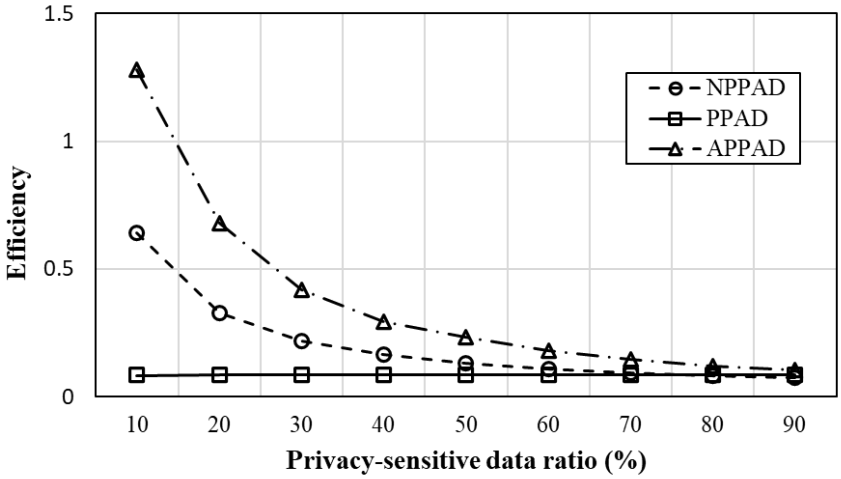

Figure 7: Efficiency vs. privacy-sensitive data ratio

The detection efficiency of the models was the highest in the order of APPAD, NPPAD, and PPAD. As APPAD performs anomaly detection through homomorphic encryption, its computational complexity increases as the privacy-sensitive data ratio increases, resulting in a slight decrease in model efficiency. Nevertheless, it demonstrated an average efficiency improvement of 88% over the NPPAD, demonstrating efficient privacy preservation. Conversely, the PPAD model, which performs homomorphic operations in bulk without considering traffic types, exhibited a significantly lower

detection efficiency. The experimental results demonstrate that APPAD effectively balances delay and privacy protection and can be efficiently applied in environments with mixed traffic types.

5 Conclusion

Anomaly detection systems are key technologies for the early identification of security threats across networks and services. Anomaly detection techniques with homomorphic encryption can preserve data privacy; however, they have limitations such as significantly increased computational delay and decreased detection accuracy. Therefore, this study proposes an anomaly detection method that dynamically applies homomorphic encryption based on the privacy sensitivity of data.

To verify the performance of the proposed model, its latency, accuracy, information leakage, and efficiency were evaluated and compared with those of a conventional model. The results show that the proposed model reduces latency by 8.6 times compared to conventional privacy-preserving anomaly detection models, while maintaining an accuracy of 86%, albeit slightly lower than that of plaintext-based models. Furthermore, the proposed model exhibits similar information leakage to privacy-preserving anomaly detection models and improves the detection efficiency by 43% compared to conventional models. However, this study does not fully specify the mechanism for real-time classification of sensitive and non-sensitive traffic. Future work will explore more effective traffic-classification strategies and validate the proposed approach across a broader range of datasets.

Acknowledgements

This work was supported by the Ministry of Trade, Industry and Energy (MOTIE) under Training Industrial Security Specialist for High-Tech Industry [grant number RS-2024-00415520] supervised by the Korea Institute for Advancement of Technology (KIAT), Ministry of Science and ICT (MSIT) under the ICAN (ICT Challenge and Advanced Network of HRD) program [grant number IITP-2022-RS-2022-00156310] and National Research Foundation of Korea (NRF) grant [RS-2025-00518150], and the Information Security Core Technology Development program [grant number RS-2024-00437252] supervised by the Institute of Information & Communication Technology Planning & Evaluation (IITP).

References

[1] Ruff, L., Kauffmann, J. R., Vandermeulen, R. A., Montavon, G., Samek, W., Kloft, M., ... & Müller, K. R. (2021). A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, *109*(5), 756-795.

[2] Motie, S., & Raahemi, B. (2024). Financial fraud detection using graph neural networks: A systematic review. *Expert Systems with Applications*, *240*, 122156.

[3] Knights, J., Heidary, Z., & Cochran, J. M. (2020). Detection of behavioral anomalies in medication adherence patterns among patients with serious mental illness engaged with a digital medicine system. *JMIR Mental Health*, *7*(9), e21378.

[4] Li, J., Tong, X., Liu, J., & Cheng, L. (2023). An efficient federated learning system for network intrusion detection. *IEEE Systems Journal*, *17*(2), 2455-2464.

[5] Wingarz, T., See, A., Gondesen, F., & Fischer, M. (2024, September). Privacy-preserving Network Anomaly Detection on Homomorphically Encrypted Data. In *2024 IEEE Conference on Communications and Network Security (CNS)* (pp. 1-9). IEEE.

[6] Syed, N. F., Shah, S. W., Shaghaghi, A., Anwar, A., Baig, Z., & Doss, R. (2022). Zero trust architecture (ZTA): A comprehensive survey. *IEEE Access*, *10*, 57143-57179.

[7] Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. *ACM Computing Surveys (Csur)*, *51*(4), 1-35.

[8] Ouyang, T., & Zhang, X. (2023). Fuzzy rule-based anomaly detectors construction via information granulation. *Information Sciences*, *622*, 985-998.

[9] Rim, D. N., Heo, D., Lee, C., Nam, S., Yoo, J. H., Hong, J. W. K., & Choi, H. (2024). Anomaly detection based on system text logs of virtual network functions. *Big Data Research*, *38*, 100485.

[10] Lee, J., Lee, E., Lee, J. W., Kim, Y., Kim, Y. S., & No, J. S. (2023). Precise approximation of convolutional neural networks for homomorphically encrypted data. *IEEE Access*, *11*, 62062-62076.

[11] Manh, B. D., Nguyen, C. H., Hoang, D. T., Nguyen, D. N., Zeng, M., & Pham, Q. V. (2025). Privacy-Preserving cyberattack detection in Blockchain-Based IoT systems using AI and homomorphic encryption. *IEEE Internet of Things Journal*.

[12] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, *1*(2018), 108-116.

[13] Benaissa, A., Retiat, B., Cebere, B., & Belfedhal, A. E. (2021). Tenseal: A library for encrypted tensor operations using homomorphic encryption. *arXiv preprint arXiv:2104.03152*.