A Study of Anomalous Communication Detection for IoT Devices Using Flow Logs in a Cloud Environment

Yutaro Iizawa[1], Norihiro Okui[2], Yusuke Akimoto[1], Shotaro Fukushima[1], Ayumu Kubota[2] and Takuya Yoshida[3]

[1]*ARISE Analytics Inc., 2-21-1, Shibuya, Shibuya, Tokyo, Japan*

[2]*KDDI Research, Inc., 2-1-15, Ohara, Fujimino, Saitama, Japan*

[3]*TOYOTA Motor Corporation, 1-6-1, Otemachi, Chiyoda, Tokyo, Japan*

{*yutaro.iizawa, yusuke.akimoto, shotaro.fukushima*}*@ariseanalytics.com,* {*no-okui, ay-kubota*}*@kddi.com,*

Keywords: Aomalous Communication Detection, IoT, IPFIX, VPC Flow Logs.

Abstract: Research on network-based anomaly detection has been conducted as a countermeasure against cyberattacks from IoT devices. Specifically, anomaly detection based on flow data, such as IPFIX, has garnered increasing attention to address the rising communication volume. In these studies, obtaining flow data from the communication data sent and received by IoT devices is necessary; however, obtaining these data can be difficult when the IoT system is already built in a cloud environment. In this study, we investigated an anomalous communication detection method using VPC Flow Logs, which can be obtained via AWS. VPC Flow Logs record only the number of packets and bytes in a single direction, resulting in less information than that obtained via flow data. For example, session information is divided into multiple records according to the time window. To increase the precision of anomalous communication detection using VPC Flow Logs data, we developed a methodology for the effective conversion of multiple VPC Flow Logs into bidirectional data. The efficacy of this approach was assessed by evaluating its performance on public datasets.

1 INTRODUCTION

In recent years, the proliferation of cyberattacks targeting Internet of Things (IoT) devices has become a growing concern. A notable example of this trend is the emergence of attacks that leverage IoT devices as conduits, potentially compromising the integrity of entire systems. In response to these threats, a range of research and development initiatives has been undertaken to enhance the security of IoT systems. One area of focus is anomaly detection, which involves the identification of unauthorized communications and bots, among other threats, within networks of IoT devices. With advancements in communication speed and capacity, research utilizing flow data, such as IPFIX, which is more lightweight than previously used packet data, has surged in recent years.

Concurrently, the emergence of applications built on cloud platforms such as Amazon Web Services (AWS) facilitates the utilization of not only PCAP and IPFIX but also cloud logs (e.g., VPC Flow Logs) as data related to communication. It is imperative to acknowledge that capturing PCAP and IPFIX necessitates the installation of dedicated software on the VM residing in the cloud. This software can potentially impact application performance. Consequently, leveraging cloud logs provides a substantial advantage. Nevertheless, it is crucial to recognize that the purpose of obtaining cloud logs differs from that of PCAP and IPFIX, making the application of anomalous communication detection unfeasible.

This study focuses on AWS VPC Flow Logs, a specific type of cloud log, and aims to address the various challenges associated with anomaly detection in cloud logs. The subsequent section provides a comprehensive overview of these challenges and the proposed methodology for addressing them. This study also includes quantitative verification of the efficacy of the proposed method through experiments with multiple machine learning and deep learning models.

2 RELATED WORKS

Conventional anomalous communication detection has historically relied on packet data, exemplified

Iizawa, Y., Okui, N., Akimoto, Y., Fukushima, S., Kubota, A. and Yoshida, T.
A Study of Anomalous Communication Detection for IoT Devices Using Flow Logs in a Cloud Environment.
DOI: 10.5220/0013461200003944

In *Proceedings of the 10th International Conference on Internet of Things, Big Data and Security (IoTBDS 2025)*, pages 410-415
ISBN: 978-989-758-750-4; ISSN: 2184-4976

by the PCAP format because of its ability to meticulously record network communications. This approach facilitates comprehensive analysis; however, the large volume of data generated necessitates efficient processing methods.

Consequently, in recent years, flow data, represented by IP Flow Information Export (IPFIX) (Trammell and Boschi, 2008; Claise et al., 2013), have been used. Flow data consist of information that aggregates packet data into sessions and have the following characteristics:

- The data size is small because individual packets are aggregated into sessions and recorded.
- It preserves statistical information about sessions, including source and destination IP addresses, port numbers, protocols, session start and end times, and communication volume.
- Information can be aggregated into uni-flows or bi-flows during a given session.

In their seminal work, Tang et al. (Tang et al., 2016; Tang et al., 2018) proposed deep learning-based anomalous communication detection models for flow data and achieved high detection accuracy. In addition, Lo et al. (Lo et al., 2021) proposed EGraphSAGE, an extension of GraphSAGE, a type of graph neural network, and demonstrated excellent performance in anomaly detection using flow data. As network traffic is projected to rise in the future due to the proliferation of 5G and the surge in IoT devices, there is a growing need for research on anomalous communication detection methods using flow data. The architecture of systems incorporating IoT devices is predominantly cloud-centered, as exemplified by Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).

In cloud-based architectures, procuring communication data such as PCAP and IPFIX from operational services necessitates the installation of dedicated libraries on each server. This approach raises concerns regarding its impact on operational costs and performance. As an alternative, cloud logs provided by cloud services, such as VPC Flow Logs in AWS, can be utilized. The present study focuses on AWS VPC Flow Logs; however, these cloud logs are not intended for storing communication data such as IPFIX data, which hinders their application in anomalous communication detection, leading to the following issues:

- Records by time window: Records are output within a time window, so a single session may be split into multiple records.
- Unidirectional: Transmitted data and received data are output as separate records
- Retained information: Because only the time window is available, the temporal order between records is lost when multiple records occur within the same time window. Compared with IPFIX, the number of recorded features is limited.

In this study, we propose a methodology for detecting anomalous communications using only cloud logs, without making any specific changes to the cloud-based system architecture, by addressing the aforementioned issues.

3 PROPOSED METHOD

3.1 Conversion from VPC Flow Logs to Sessions

In this study, records of VPC Flow Logs are defined as a session if they have the same values, including source IP addresses, destination IP addresses, source port numbers, destination port numbers, and protocols, and if the time interval between consecutive records is within the predefined time window. Specifically, two cloud logs are considered to be in the same session if the time interval between the start times of consecutive cloud logs is within 600 seconds. The reason for choosing 600 seconds is that the maximum duration of a single window in AWS VPC Flow Logs is 600 seconds.

Furthermore, the aggregation interval of VPC Flow Logs records is defined as the time window mentioned above. Records identified as part of the same session are then aggregated by session according to the following procedure:

1. Sort records in chronological order based on matching signatures (combinations of source and destination IP addresses, port numbers, and protocols).
2. Determine session boundaries based on the time interval conditions mentioned above.
3. Aggregate statistics (packet count, byte count, etc.) for each session.

3.2 Conversion to Bi-Flows

VPC Flow Logs are not captured for each individual communication session; therefore, it is not possible to determine the outbound and inbound directions for a particular session. Consequently, a processing method has been implemented that integrates the outbound and inbound directions of communication concurrently with sessionization.

Table 1: Details of the datasets.

Dataset name	Split date & time	Format	Benign rate	# of features	# of training data	# of test data
MedBIoT	2019-03-07 17:14	VPC Flow Logs	92.6%	4	126,866	15,791,081
		VPC Bi-Flow	92.6%	7	17,352	2,789,571
Edge-IIoT	2021-11-24 00:00	VPC Flow Logs	97.3%	4	254,396	10,349,067
		VPC Bi-Flow	97.3%	7	85,806	7,451,794

For a given session, the following conditions must be met for it to be designated as the inbound session:

- The protocol must match.
- The source IP address and port number of one session must match the destination IP address and port number of the other session
- The destination IP address and port number of a session must match the source IP address and port number of another session

This allows for the treatment of sessions in which the source and destination IP addresses and port numbers respond as a single bidirectional communication.

To integrate unidirectional sessions into bidirectional sessions, the following procedure is implemented:

1. For each communication record, search for a "response communication" that meets the conditions listed above, and forms a pair.
2. Integrate the paired communication records into a single record.
3. Recalculate the bidirectional communication volume (transmission volume and reception volume), packet count (transmission packet count and rejection packet count), communication time, and other relevant metrics for the integrated record.
4. For records in which no pair is found, the feature values for the response session are set to 0 and then combined.

3.3 Anomaly Detection Models

To confirm the accuracy of the anomaly detection method using cloud logs, we conducted an experiment using a dataset in which open data were converted to the VPC Flow Logs format. We used a library specifically developed for the this conversion to the VPC Flow Logs format.

We performed sessionization and bidirectionalization as preprocessing on the dataset and then evaluated the accuracy of a model commonly used in anomalous communication detection.

4 EXPERIMENT

In this section, we evaluate the proposed method using communication data for the anomalous communication detection task. First, we introduce the dataset, and then explain the models. Next, we consider the experimental results.

4.1 Datasets

We used MedBIoT (Guerra-Manzanares et al., 2020) and Edge-IIoT (Ferrag et al., 2022) as the datasets for verification. These datasets contain both benign and anomalous communications. For each dataset, we created VPC Flow Logs format data using a tool that converts PCAP to VPC Flow Logs format. Next, the VPC Flow Logs format is converted into session units using the proposed method and then converted into bidirectional data (hereafter referred to as VPC Bi-Flow). The details of the dataset are shown in Table 1. The dataset is sorted in chronological order, and the split time is set so that the training data consist solely of benign data, which is divided into training and test datasets.

The features used as inputs to the model were constructed as follows. First, the features used across all data formats include the number of buckets, traffic, the amount of communication per packet, and communication time. Note that the communication in VPC Flow Logs includes both outbound and bound routes. Next, the bidirectional features available in VPC Bi-Flow consist of the number of packets, the amount of communication, and the amount of communication per packet in the inbound communication.

4.2 Experimental Specifications

In this section, we discuss the models and evaluation metrics used to evaluate the dataset described in Section 4.1.

4.2.1 Models

We used AutoEncoder (Aggarwal, 2013) and DeepSVDD (Ruff et al., 2018), Isolation Forest, KNN, and LOF. The models were developed using

Table 2: Experimental results for MedB-IoT.

Algorithm	ROC-AUC		PR-AUC		F1 score	
	VPC Flow Logs	VPC Bi-Flow	VPC Flow Logs	VPC Bi-Flow	VPC Flow Logs	VPC Bi-Flow
AutoEncoder	0.121	0.947	0.040	0.781	0.138	0.927
DeepSVDD	0.361	0.943	0.057	0.736	0.148	0.928
IForest	0.460	0.285	0.065	0.202	0.185	0.481
KNN	0.885	0.183	0.578	0.185	0.759	0.489
LOF	0.820	0.807	0.389	0.605	0.666	0.810

Table 3: Experimental results for Edge-IIoT.

Algorithm	ROC-AUC		PR-AUC		F1 score	
	VPC Flow Logs	VPC Bi-Flow	VPC Flow Logs	VPC Bi-Flow	VPC Flow Logs	VPC Bi-Flow
AutoEncoder	0.441	0.362	0.109	0.105	0.169	0.165
DeepSVDD	0.677	0.495	0.259	0.441	0.352	0.604
IForest	0.759	0.206	0.170	0.021	0.241	0.038
KNN	0.499	0.841	0.160	0.133	0.239	0.158
LOF	0.392	0.752	0.178	0.044	0.279	0.108

PyOD (Zhao et al., 2019) to implement each model, and the default values for the hyperparameters were used.

4.2.2 Metrics

The evaluation metrics used are the area under the ROC curve (ROC-AUC), the area under the precision-recall curve (PR-AUC), and the maximum F1 score. For all these metrics, higher values indicate better performance.

4.3 Results and Discussion

4.3.1 MedB-IoT

The experimental results are presented in Table 2. With the exception of KNN, there was a general improvement in each accuracy metric. However, for KNN, each accuracy metric decreased.

The histogram of the anomaly score for the test data in MedBIoT's VPC Bi-Flow is shown in Figure 1∼Figure 5 .

The experimental results demonstrate a notable enhancement in the accuracy of the DeepSVDD model. As illustrated in Figure 2, the histograms of benign data (blue) and anomalous data (orange) are separated by a threshold of Anomaly Score $=$ 0.05. A subsequent examination of the data with Anomaly Score $\geq$ 0.05, which had a high concentration of anomalous data, revealed approximately 770,000 records with a response communication volume and packet count of 0. This finding suggests that the presence of response communication can be used as an indicator of potential anomalies. Furthermore, a trend toward an Anomaly Score $\geq$ 0.05 was identified in approximately 80,000 benign data records with no response communication. In the contrast, there were many records with Anomaly Score $<$ 0.05 for anomalous data with response communication. Also, all benign data with Anomaly Score $<$ 0.05 had response communication. This indicates that the ability to assess anomalyities for bidirectional communication is insufficient.

These trends were also observed in AutoEncoder (Figure 1) and LOF (Figure 5), suggesting that the evaluation metrics for VPC Bi-Flow resulted in relatively high results. In contrast, with IForest (Figure 3) and KNN (Figure 4), the histograms of benign data and anomalous data did not separate well, regardless of whether there was response communication. Although the training data for VPC Bi-Flow included approximately $3,000$ out of the approximately $17,000$ data points without response communication, the model, which achieved the high accuracy, learned to classify data without response communication as anomalies relatively well.

4.3.2 Edge-IIoT

The experimental results are presented in Table 3. There was little improvement in the accuracy of each indicator.

We will focus on the DeepSVDD model, which has relatively high accuracy, and discuss it. Figure 6 shows the histogram of the Anomaly Score for VPC Bi-Flow. When we examined the anomalous data with Anomaly Score $\geq$ 0.005, we found that it only included the records with response communication. In addition, anomalous data with Anomaly Score $\leq$ 0.005 were included data without response communication. With respect to the benign data in the test data, approximately 6 out of every 10 cases had no response communication, while in the training data, approximately $2,000$ out of approximately $85,000$ cases included response communication. This suggests that

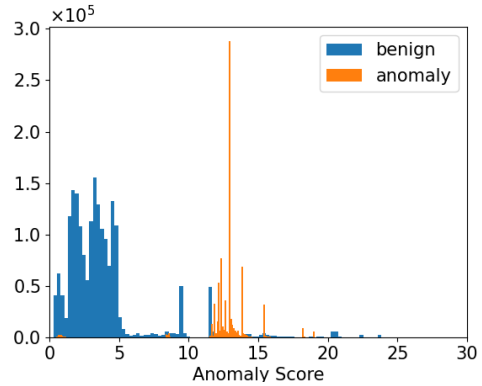

Figure 1: MedB-IoT: Histogram of Anomaly Scores for VPC Bi-Flow using AutoEncoder.

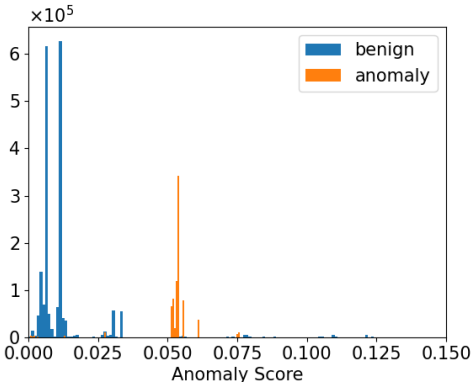

Figure 2: MedB-IoT: Histogram of Anomaly Scores for VPC Bi-Flow using DeepSVDD.

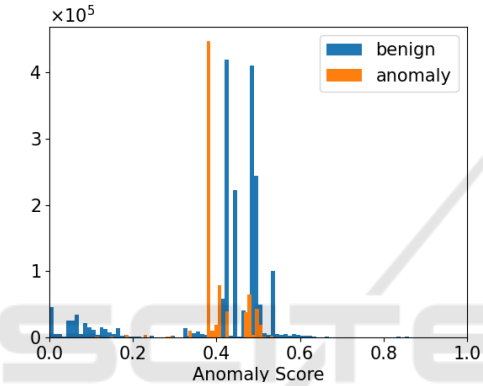

Figure 3: MedB-IoT: Histogram of Anomaly Scores for VPC Bi-Flow using IForest.

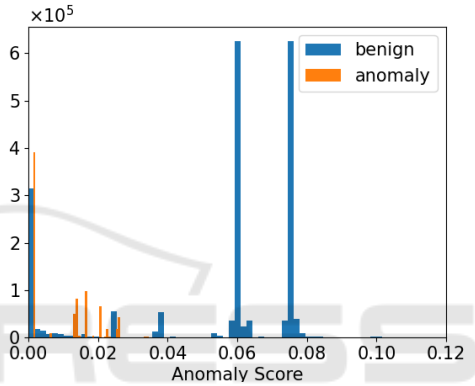

Figure 4: MedB-IoT: Histogram of Anomaly Scores for VPC Bi-Flow using KNN.

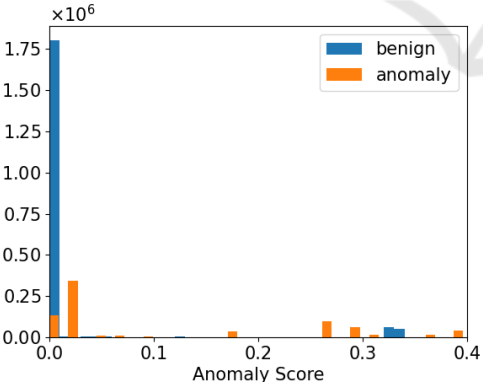

Figure 5: MedB-IoT: Histogram of Anomaly Scores for VPC Bi-Flow using LOF.

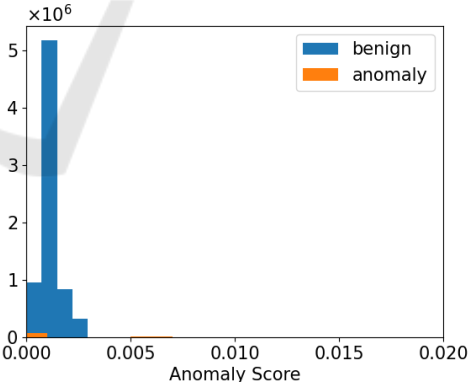

Figure 6: Edge-IIoT: Histogram of Anomaly Scores for VPC Bi-Flow using DeepSVDD.

data drift may have occurred in the benign communication between the training and test datasets due to the presence or absence of response communication. Consequently, the anomaly score of the anomalous data did not become relatively large.

5 CONCLUSION

In this study, we examined the issues associated with detecting anomalous communication from IoT devices using cloud logs, particularly VPC Flow Logs, and pro- posed a data conversion method to address

them. In the proposed method, fragmented session information in cloud logs is aggregated, and the transmitted and retrieved data are integrated to convert it into a data for- mat that can be applied to an anomalous communication detection model.

In the accuracy evaluation using public data, the method improved the accuracy of detecting anomalies without response communication; however, there was little improvement in datasets that contained many benign communications without response communication.

Future issues include the insufficient detection accuracy for anomalies related to outbound communication, and the challenge of detecting these anomalies while ensuring that benign communication without response communication is classified as benign. One approach would be to use the proposed method to segment communication into sessions, divide the data into send/receive and send- only data, and perform learning and inference for each category.

REFERENCES

Aggarwal, C. C. (2013). Outlier analysis. In *Springer: New York*.

Claise, B., Trammell, B., and Aitken, P. (2013). Specification of the ip flow information export (ipfix) protocol for the exchange of flow information. Technical report.

Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., and Janicke, H. (2022). Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications: Centralized and federated learning.

Guerra-Manzanares, A., Medina-Galindo, J., Bahsi, H., and Nõmm, S. (2020). Medbiot: Generation of an iot botnet dataset in a medium-sized iot network. In *International Conference on Information Systems Security and Privacy*.

Lo, W. W., Layeghy, S., Sarhan, M., Gallagher, M. R., and Portmann, M. (2021). E-graphsage: A graph neural network based intrusion detection system for iot. *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, pages 1–9.

Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., and Kloft, M. (2018). Deep one-class classification. In *International conference on machine learning*, pages 4393–4402. PMLR.

Tang, T. A., Mhamdi, L., McLernon, D. C., Zaidi, S. A. R., and Ghogho, M. (2016). Deep learning approach for network intrusion detection in software defined networking. *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pages 258–263.

Tang, T. A., Mhamdi, L., McLernon, D. C., Zaidi, S. A. R., and Ghogho, M. (2018). Deep recurrent neural network for intrusion detection in sdn-based networks. *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, pages 202–206.

Trammell, B. and Boschi, E. (2008). Bidirectional flow export using ip flow information export (ipfix). Technical report.

Zhao, Y., Nasrullah, Z., and Li, Z. (2019). Pyod: A python toolbox for scalable outlier detection. *Journal of Machine Learning Research*, 20(96):1–7.