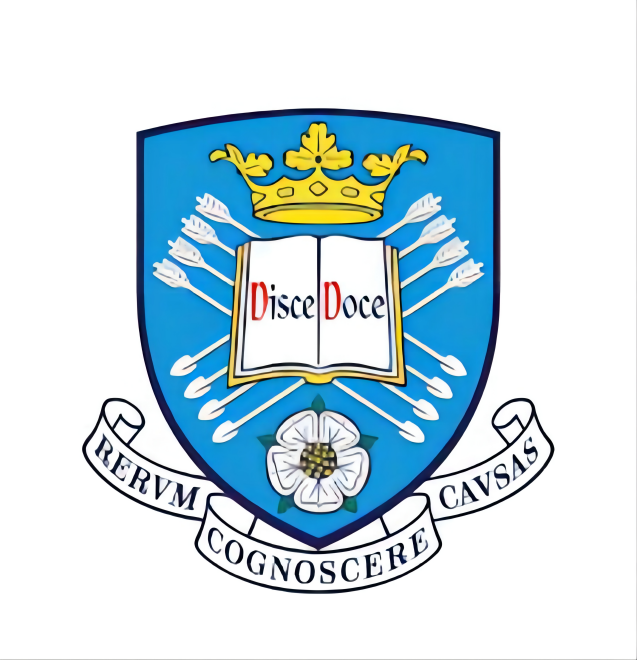
University of Sheffield

**Malicious Endpoint Detection & Response using ML**



Shangyuan Liu

*Supervisor*: Dr. Olakunle Olayinka

A report submitted in partial fulfilment of the requirements

for the degree of MSc Cybersecurity and Artificial Intelligence

*in the*

Department of Computer Science

May 11, 2022

**Chapter 1**

**Introduction**

计算机网络的安全是一个国际化的问题，每年全球因计算机网络的安全系统被破坏而造成的经济损失达上千亿美元。当商户、银行与其他商业与金融机构在电子商务热潮中纷纷进入Internet，以政府上网为标志的数字政府使国家机关与Internet互联。通过Internet 实现包括个人、企业与政府的全社会信息共享已逐步成为现实。随着网络应用范围的不断扩大，对网络的各类攻击与破坏也与日俱增。无论政府、商务，还是金融、媒体的网站都在不同程度上受到入侵与破坏。网络安全已成为国家与国防安全的重要组成部分，同时也是国家网络经济发展的关键。对入侵攻击的检测与防范、保障计算机系统、网络系统及整个信息基础设施的安全已经成为刻不容缓的重要课题。

**Chapter 2**

**Literature Review**

**2.1 Background**

As technology rapidly develops and the information age expands, the technology areas such as 5G, cloud computing, the Internet of Things (IoT) and artificial intelligence have been brought about significant innovation. The number of global IoT connections is expected to reach 14.4 billion approximately by the end of 2022 and the number of IoT devices will rise to around 27 billion in 2025 [1]. However, the endpoint environment, comprising various devices including the IoT, smartphones and laptops etc, is being exposed to massive cyber attacks such as ransomware, malware, phishing, DDoS, and botnets [2]. According to research studies, up to 90% of successful cyber attacks and 70% of successful data leakage incidents emanate from endpoint devices [3]. Moreover, the endpoint environment also has been challenged by many unknown malicious attacks. Most significantly, traditional endpoint security tools and approaches are ineffective at detecting or eliminating advanced threats. As a result, these threats can lurk on the network for months, collecting data and identifying vulnerabilities to prepare for ransomware attacks and zero-day exploit attacks.

**2.****2 Endpoint** **Detection and Response**

In order to mitigate malicious cyber threats, [Anton Chuvakin](https://blogs.gartner.com/anton-chuvakin/2013/07/26/named-endpoint-threat-detection-response/" \t "https://www.mcafee.com/enterprise/en-us/security-awareness/endpoint/_blank) suggested endpoint detection and response (EDR) [4], which is a solution to fortify endpoint security that will continuously collect endpoint data (computers, servers, mobile devices, IoT devices, etc.). And combined with automated rules-based response and analysis capabilities to analyse and detect suspicious cyber activities and threats on endpoints, and automatically respond to secure endpoint devices and IT assets [5].

In terms of the history of endpoint security, traditional anti-virus (AV) is the first generation, which primarily matches known virus databases and signature databases to detect and kill viruses. The endpoint protection platform (EPP) is the second generation. The main function of the EPP system is to fortify endpoint devices' security, including anti-virus, anti-spyware, anti-cyber-attack, anti-phishing, firewall, prevent unauthorized access [8], and also provide data loss protection and data encryption. It is a comprehensive endpoint protection product. While EDR belongs to the third generation, it relies on an accurate indicator of compromise (IOC) threat intelligence information support combined with big data analysis, artificial intelligence, analysis technologies, thus forming a complete endpoint protection system [9]. The emergence of EDR remedies the shortcomings of the traditional EPP, which is not a complete substitute for EPP and AV, but a compliment to each other. In the future, there will be a move towards a more comprehensive Extended Detection and Response (XDR) [10].

EDR technology differs from the previous static defence based on decision boundaries, rules, and strategies. It is a proactive endpoint security protection technology that monitors any security threats and malicious activities by recording endpoint and network events, combining known IOCs, using behaviour analysis and machine learning techniques to monitor any security threats and malicious activities, and automating blocking, forensics, and remediation. Therefore, EDR technology has become a significant role in the overall security protection system in cyberspace because of its sophistication and superiority against unknown threat attacks, zero-day vulnerability attacks, APT attacks, and fileless attacks.

**2.3** **Threats to** **Endpoint Devices**

Due to COVID-19 pandemic, the number of attacks detected in the first quarter of 2020 alone reached 445 million [64], and the attacks against endpoint devices can be classified into the following eight main categories.

**Ransomware**

Ransomware is a type of malware that is deployed on enterprise endpoint devices frequently. And it employs encryption algorithms to encrypt data and files that infected individuals generally are unable to decrypt, and all ransomware requires victims to pay a ransom in exchange for regaining access. Endpoint security includes advanced threat detection to identify and block ransomware before it encrypts data. [11] [12].

**Advanced Persistent Threat**

Advanced Persistent Threat (APT) is a sophisticated, persistent, and multi-stage cyber attack that includes three elements: advanced, long-term, and threatening. Advanced means that sophisticated malware and techniques are used to exploit vulnerabilities in systems. And Long-term refers to the continuous monitoring of a specific target, retaining access to it, and obtaining data from it over time. Threatening emphasises human involvement in planned attacks that target high-value organisations [13]. Successful APT attacks result in significant financial loss or political impact, and therefore APT is currently one of the most serious threats to companies and governments [13][14].

**Malware**

Malware has the ability to access the wireless and sensor network environment in a variety of methods. Once an intrusion is successful, it is more difficult to prevent than TCP/IP networks in terms of transmissibility, invisibility, and destructiveness. It can be difficult to detect and remove such malicious code.

**Phishing**

Phishing describes the use of fraudulent emails or other digital communications to trick users into revealing sensitive information. Basically, phishing attacks account for over half of all cybercrime, and phishing attacks can install malicious ransomware, Trojan horses, etc. on targeted endpoint systems [15]. Therefore, it can result in the loss of large amounts of sensitive private data from endpoint devices and can lead to identity theft, which can cause more serious security incidents [16]. Advanced Endpoint Security solutions integrate email gateways that identify and isolate malicious emails, thereby reducing the risk of users falling into the trap of phishing tactics.

**Botnet**

A botnet comprises a series of hosts infected with malware. Attackers infect a wide range of hosts on the Internet by spreading botnets through various means, forming a botnet. Furthermore, it can be used by the attacker to perform a series of malicious activities, such as distributed denial of service attacks, sending spam, stealing personal information, and performing distributed computing tasks. Such cyber-attacks can result in the crippling of the entire IoT infrastructure network or critical applications, as well as the leakage of a significant amount of confidentiality or personal privacy, and can also commit other criminal activities.

**Port Scanning**

Port scanning is defined as a client sending a corresponding request to a certain range of server ports in order to confirm which ports are available. However, the technique has been used for malicious network activities, where hackers send a set of port scan messages in order to find vulnerabilities within a specific computer port to obtain sensitive information from the user [5]. In response to such attacks, users should implement monitoring of port traffic to prevent attackers from exploiting potential vulnerabilities to gain unauthorized access to the network and commit illegal activities [17].

**Sensor-Based Threats**

Smart IoT devices are vulnerable to sensor-based threats and attacks due to the lack of proper security mechanisms to control the use of sensors by installed applications. By exploiting sensors on smart devices (e.g. accelerometers, gyroscopes, microphones, light sensors, etc.), attackers can extract information from the device, transmit malware to it or trigger malicious activity to compromise it [18].

**Telnet Attacks**

Telnet is a TCP/IP application, as well as a protocol used to connect a local computer to a remote computer. But for an attacker, Telnet is just a tool for remote login. Once an attacker has established a Telnet connection to a remote host, the intruder can use the software and hardware resources on the target host. Due to the continued increase in Tenet-based attacks in recent years, Šemić, H. and Mrdovic, S. proposed an approach for implementing a honeypot for detecting and responding Tenet attacks on IoT devices [19]. Furthermore, IoT honeypots are used to analyse malicious Tenet traffic on IoT devices running on different CPU architectures.

**2.4 Machine Learning-based Detection Algorithm**

The number, variety, speed of spread, and difficulty of detection of malware have increased, which poses a significant challenge to the security of endpoint devices [20]. Therefore, the traditional malware detection techniques that are based on features and rules do not seem to provide the most effective method to protect users from malware. According to Kaur, H., and Tiwari, R. [5], the research identified malware that uses AI algorithms to alter its signature to evade the EDR system. Therefore, research on malicious endpoint detection based on machine learning and deep learning is a popular research direction in the field of cyber security in recent years. The technology is able to provide excellent solutions in practical applications, not only improving efficiency but also reducing detection costs [21]. In this section, some common ML algorithms are compared that have been proven to detect and respond to malicious endpoints.

**Support Vector Machine (SVM)**

In 1992, the Support Vector Machine (SVM) was introduced by Boser et al. [22], which is a supervised learning model in classification problems and regression analysis. The basic principle of SVM is to dichotomise the training data set correctly and separate the hyperplanes at the maximum geometric interval. In Figure 2.1, is the separating hyperplane.

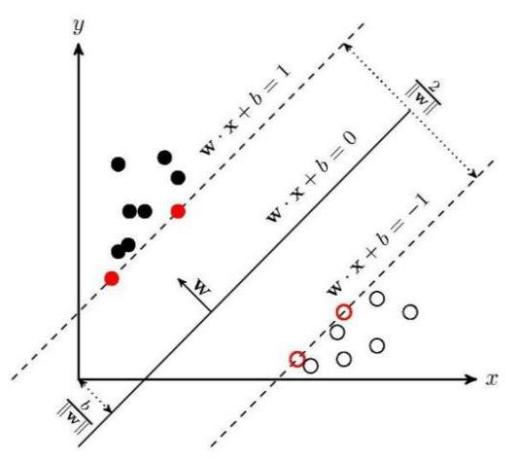


Figure 2.1 Fundamental of Support Vector Machine Architecture

The SVM algorithm was originally designed for binary classification problems [23], making SVM one of the most popular algorithms for malicious endpoint detection. Previous research by Ambwani [28] showed using multi-class SVM classifiers to identify various types of attacks, and the application of SVM in intrusion detection had a good performance. Moreover, SVM has been proven to have a higher potential compared to traditional methods. However, it does not perform well in multi-classification detection, and some questions remain to be addressed.

Furthermore, some authors have suggested [31] a semi-supervised learning mechanism for alarm filters in intrusion detection systems (IDS), which are built using the Reduced Support Vector Machine (RSVM) [30] algorithm to reduce false alarms from NCF instances. The final results show that the use of connection features with supervised and semi-supervised learning techniques successfully reduces false positives by up to 85%, while still maintaining a high attack detection rate. However, the present studies were constrained to the construction of an alert filter using partially annotated data in this approach is not satisfactory.

In order to address the quadratic programming (QP) problem that occurs in the training process of SVMs, the research of Tian et al. [29] has come up with a good solution, which is an AdaBoost.M1+ SVM algorithm based on an integrated multi-classifier and used the Sequential Minimal Optimization (SMO) algorithm to solve the optimization problem of the SVM objective function. Then executing the relevant files generates a log file that provides statistical analysis of API calls in a virtual environment. The SVM model finally achieves 95.2% malware classification accuracy after the weighted average.

The previous research of Ioannou, C. and Vassiliou, V. [32][33] demonstrated an algorithm that the C-Support Vector Machines (SVM) was used to detect network layer attacks on IoT devices. The C-SVM model was trained using benign and malicious local sensor activities and could achieve 100% accuracy in detecting specific malicious attacks. A closer look at the literature however reveals a number of gaps and shortcomings, such as the performance of the method is not satisfactory when evaluating against routing attacks in different network topologies. In addition, the investigators did not compare the experimental results with other models.

**Decision Trees (DT) -** **Random Forest (****RF)**

The decision tree is a non-parametric supervised learning algorithm for classification and regression tasks. It is a tree structure in which each internal node represents a test on an attribute, each branch represents a test output, and each leaf node represents a category as represented in Figure 2.2. Basically, there are three common DT node classifications based on ID3, C4.5, and CART, where ID3 was proposed by Quinlan J.R [24] that based on information entropy and information gain, Quinlan J. R [25] presented the C4.5 algorithm which is an improved version of ID3 based on information gain rate, and Breiman, L. [26] proposed CART which is based on the Gini coefficient.

Tin Kam Ho (1995) proposed the Random Forest algorithm, which further incorporates the selection of random attributes in the training process of decision trees, based on the construction of Bagging integration using decision trees as base learners [27]. As shown in Figure 2.3, when performing the classification task, each decision tree in the forest is decided and classified separately, and a classification result is obtained separately, then RF will take the most classified result as the final result.

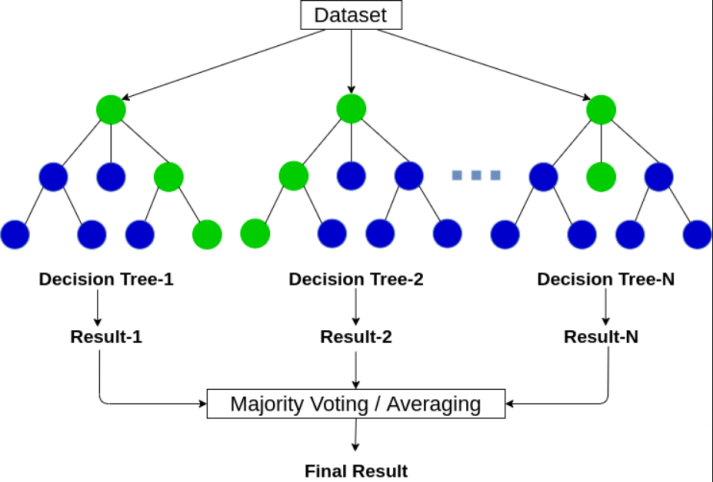


Figure 2.2 Decision Trees Architecture Figure 2.3 Random Forest Architecture

In 2019, Hasan, M. et al. [34] adopted the Distributed Smart Space Orchestration System (DS2OS) to create a virtual IoT environment to capture the required experimental data as well as used the Kaggle dataset provided by Pahl et al. [35]. And several machine learning algorithms such as DT and RF are applied to identify attacks and anomalous activities against sensors in IoT endpoints. The results of the five-fold cross-validation of the dataset demonstrated the effectiveness of the method, however, in the case of large data and other unknown problems, the method cannot ensure the performance of the classifier.

Furthermore, feature selection is an arguably important question to be addressed. In order to distinguish normal traffic from malicious traffic, it is also possible to identify different types of network attacks, Alothman, Z et al. [36] suggested experiments with multiple classifiers (J48, RF, and MLP) using Pearson’s Correlation Coefficient (PCC) and Relief-F to remove the Bot-IoT dataset some irrelevant features. The final experimental results showed that the RF classifier outperformed other classifiers, but due to data sample imbalance problems, however, the detection rate for specific attack types is not high.

Afterward, Penmatsa, R. K [37] also introduced a method that achieves 99.73% accuracy after a 90% reduction in network traffic features. The method was based on Grey Wolf Optimization (GWO) to optimize features in botnet data without compromising classification accuracy. The GWO optimisation algorithm was also evaluated on SVM, KNN, and DT models respectively, and the GWO algorithm performed best on the DT classifier compared to the other classifiers.

**Logistic regression (LGR)**

Logistic regression, also known as logit regression, is a generalised linear regression analysis model and is frequently used for dichotomous classification. LGR essentially assumes that the data follows this distribution and then uses extreme likelihood estimation to estimate the parameters. It is based on linear regression with a sigmoid activation function, which has the mathematical expression:

IMG_256

In 2019, Ongun, T. et al. [38] leveraged LGR and other machine learning algorithms to detect the CTU-13 dataset to solve the problem of detecting botnets in network traffic data. In the research, connection-level representation and aggregated traffic statistics feature representation methods were tried for feature extraction. However, the proposed research suffers from limitations due to LGR does not handle the data sample imbalance well compared to random forest and gradient boosting. In order to properly address this question, the data sampling strategies are supposed to be appreciated in data processing stage.

In addition, Machaka, P. et al. [39] investigated a semi-supervised machine learning model combining LGR and K-Means for detecting DDoS attacks in CPS-IoT in 2021. The One-over-rest training scheme and the bfgs algorithm were utilized to train the LR model for the DAPRA IDS dataset. The use of a semi-supervised machine learning model resulted in better performance, achieving 100% detection accuracy and 0 false positives.

**Neural Networks**

Neural Networks (NNs) have a structure inspired by the human brain, mimicking the way biological neurons pass signals to each other [41]. Therefore, an NN composed of node layers, containing an input layer, one or more hidden layers, and an output layer, as depicted in Figure 2.4 [40]. Each node is also known as an artificial neuron, which is connected to another node and has associated weights and thresholds. Neural network relies on large amounts of training data and continues to learn to improve its accuracy over time [40], so in general, it consistently outperforms other machine learning models.

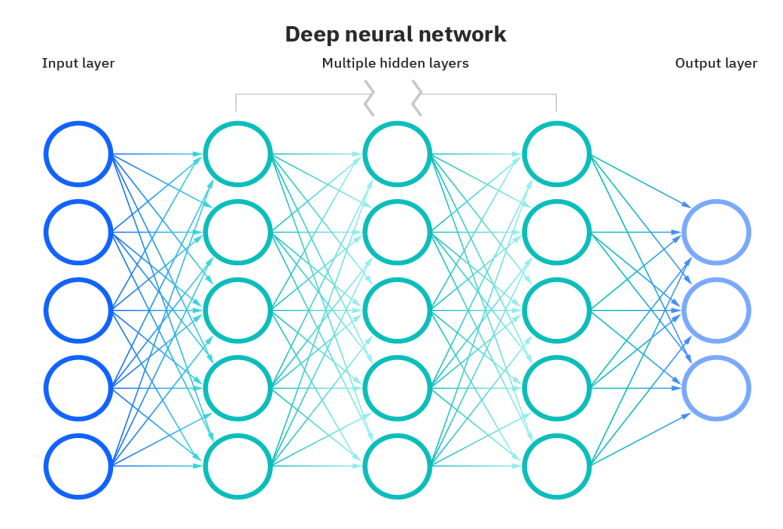


Figure 2.4 Deep Neural Networks Architecture

More formally, NN algorithms have been studied and found to be suitable for big data research and also have become one of the most popular approaches for detecting malicious endpoints and network attacks [42][43]. In 2016, a multilevel intrusion detection approach using NN was proposed by Ji, S.-Y. et al. Discrete Wavelet Transform (DWT) features were used to generate predictive models to detect anomalous behaviour and attacks on network traffic [45]. Similarly,

some authors [46][47] have driven the further development of using DWT combined with machine learning and data mining methods to construct detection models.

##### Deep Neural Network (DNN) is a comprehensive learning model. Because of its powerful computational capabilities with more hidden layers [49], it can be one of the most widely used learning techniques for detecting malware and endpoints currently. Gao, M. et al. [48] combined a 4-layer DNN and Apriori algorithm to design an IDS. The Apriori association analysis algorithm was applied to compute association rules between discrete features to reduce the false alarm rate. The results showed that the application of association analysis and DNN has good results, but the structure of NN still has much space for optimization.

Nagisetty., A., & Gupta, G.P. [44] offered a system for detecting malicious network traffic in IoT backbone networks using four different deep learning models: Multi-Layer Perceptron (MLP), CNN, DNN, and Autoencoder. The proposed models were evaluated by UNSW-NB15 and NSL-KDD9. The results showed that all models outperformed UNSW-NB15 on the NSL-KDD99 dataset, and DNN had the best performance with an accuracy of 99.24%. This has been previously assessed only to a very limited extent because such datasets do not provide a good representation of the network environment of IoT endpoints.

In respect to real time detection, Lee, W. Y. et al. [50] presented a classification approach for malicious Android applications. A combined model was constructed by stacking RNN and CNN, where CNN and Gated Recurrent Unit (GRU) were used to extract features from the dataset and the RNN was utilized to learn the data features for classification. The approach decreased the training time of the RNN model and achieved a true positive rate of 97.7% with a false positive rate of 0.01.

**2.5 Research Datasets**

In the past two decades of research in the field of cybersecurity, most early studies as well as current work focus on DARPA, KDD-99 CUP, and NSL-KDD datasets [28] [31] [37] [39] [45] [48] [50], which are not only the most widely used datasets but also the benchmarks for evaluating the performance of detection algorithms and models with merits.

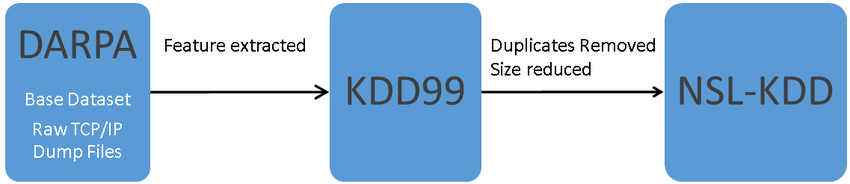


Figure 2.5 The relationship between DARPA, KDD99, and NSL-KDD datasets

Shalaginov et al. [56] proposed an ANN-based model for detecting malicious attacks in the MQTT network environment. And the proposed model was trained and evaluated in the KDD-99 and NSL-KDD datasets, but such datasets do not representative of the IoT system environment under MQTT networks. As shown in Figure 2.5, KDD99 was extracted from DARPA and NSL-KDD was generated from KDD99. However, because of a large amount of data redundancy in the datasets, these datasets have been gradually abandoned because the redundancy problem affects the accuracy of the classifiers [51]. In an age of rapid IoT technological advancement, the conventional Internet traffic datasets are not representative of the traffic found in wireless sensor networks.

The MQTT protocol is one of the most common standards for machine-to-machine communication in the IoT [52]. Therefore, datasets based on the MQTT protocol represent the characteristics of IoT networks better and have been frequently used in IDS and EDR research past years. For instance, the following studies were conducted on MQTT-based devices. In order to address the malicious endpoint problem, Khan, Muhammad Almas et al. [52] in the study used the MQTT-IoT-IDS2020 dataset [53] to construct a DNN-based IDS. The results show that DNN achieves 99% and 98% detection accuracy for binary and multi-class attacks in the feature data models of Uni-Flow and Bi-Flow.

Furthermore, Ullah, I. and Mahmoud, Q.H. [54] created a new dataset based on the MQTT-IoT-IDS2020 dataset and other datasets (MQTTset, IoT-23) and constructed a feedforward neural network (FNN) based model for detecting the malicious endpoints. The new dataset with simplified features reduces the time and computational complexity of training the model and predicting the classification while meeting the high detection accuracy.

MQTTset is another dataset focused on the MQTT protocol, which was proposed and used in IoT networks by Vaccari et al. [55] to classify legitimate traffic and network attacks in MQTT networks. The MQTTset dataset collected and created by different IoT sensors (temperature, motion sensors, etc.) was used in [57] study and the best model of machine learning was selected to detect attacks in the MQTT-IoT network. The results showed that the trained learning model could secure the IoT environment and achieve an effective detection accuracy of over 99% against attacks in MQTT networks. Idrissi et al. [58] also performed a benchmark test of the proposed algorithm based on the Generative Adversarial Network (GAN) architecture using the MQTTset dataset.

In most relevant datasets, there is not much information included about the botnet scenario. Therefore, Koroniotis, N. et al. [59] suggested a new data set, BoT-IoT, which contains both legitimate and IoT network traffic as well as various types of attacks. Some features with low relevance were removed by using techniques such as feature selection and supervised learning techniques (SVM, RNN, LSTM) were implemented to evaluate the new dataset. The evaluation results showed that both the RNN and LSTM models with an accuracy of over 99.74%. And the dataset with the 10 best features performed better than the full dataset. In 2020, a hybrid algorithm was proposed by Shafiq, M. et al. [60] for BoT-IoT attack traffic identification from several different machine learning algorithms. And proposed an algorithm to select the most effective machine learning model based on a bijective soft set approach.

The 23 sub-datasets in the IoT-23 dataset have serious data sample imbalance problems. Therefore, Oha, C.V. et al. [62] created 13 CSV files for each of the 13 labels, containing data from all 23 datasets. The data were randomachine learningy selected to generate new datasets to mitigate the underfitting and over-fitting problems associated with sample imbalance. And Abdalgawad, N. et al. [61] presented generative deep learning models such as AAE and BiGAN trained and generated using the IoT-23 dataset. The evaluation results showed that the model outperforms traditional machine learning algorithms, and the dataset can capture attacks from IoT endpoints effectively.

Liu et al. [63] proposed to use common smart home devices to create the dataset IoT-NI. After data pre-processing, the dataset was applied to five common machine learning methods, including LGR, SVM, KNN, RF and XGBoost. The accuracy, recall and F1 scores of RF were evaluated to be 100%. Although the RF method provides the highest metric score, it requires the largest amount of computation.

**2.7** **Comparison Table of Past Research Literature**

Table 2.1 summarises the results obtained by applying different machine learning algorithms on different datasets in recent years. These classifiers and algorithms were trained on the features of the proposed dataset and used to classify different network attacks. It can be seen from Table 2.1 that deep learning-based algorithms perform better than traditional machine learning methods in detecting malicious endpoints, with an accuracy of 99.9%, but with special feature engineering, some DT and RF classifiers can achieve equally high accuracy rates.

In the last two years of research datasets such as KDD-99, UNSW- NB15 and NSL-KDD are not sufficiently representative of the heterogeneous nature of current IoT networks and their performance in terms of attack diversity among IoT endpoints is fairly limited. Therefore, more and more datasets on IoT are being adopted, such as MQTT-IoT-IDS2020, MQTTset, BoT-IoT, IoT-2 and IoT-NI, which were captured by virtual or realistic IoT sensors. And in order to improve model performance, some datasets were individually designed by the researchers for a certain attack category. From the results, these datasets on IoT are able to accomplish the classification of normal and malicious IoT traffic with high accuracy.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Algorithm** | **Year** | **Dataset** | **Max ACC** | **Max F1** | **REF** |
| Muti-Class SVM Classifiers | 2003 | KDD-99 | 91.67% |  | [28] |
| AdaBoost.M1+ SVM - SMO | 2010 | VET Zoo | 95.2% |  | [29] |
| RSVM | 2010 | KDD-99 | 90.91% |  | [31] |
| C - SVM | 2019 | RMT tool | 75% |  | [32] |
| Decision Tree / Random Forest | 2019 | DS2OS | 99.4% | 0.99 | [34] |
| J48, RF and MLP | 2020 | Bot-IoT | 99.9% |  | [36] |
| GWO - DT | 2021 | KDD-99 | 99.73% |  | [37] |
| Logistic Regression | 2019 | CTU-13 | 99.8% | 0.97 | [38] |
| LGR - K-Means semi-supervised | 2021 | DAPRA IDS | 100% | 1.0 | [39] |
| MLP / CNN / DNN / Autoencoder | 2019 | NB15 / NSL-KDD | 99.24% | 0.9928 | [44] |
| NN - DWT | 2016 | NSL-KDD | 96.67% |  | [45] |
| 4-layer DNN + Apriori | 2020 | NSL-KDD |  | 0.70 | [48] |
| CNN - GRU / RNN | 2019 | NB15 / NSL-KDD | 99.24% |  | [50] |
| DNN | 2021 | MQTT-IoT-IDS2020 | 99.9% |  | [53] |
| FNN | 2022 | MQTT-IoT-IDS2020  MQTTset / IoT-23 | 99.93% | 0.993 | [54] |
| LGR / KNN / RF / SVM / DT | 2022 | MQTTset | 99.89% | 1.0 | [57] |
| GAN | 2022 | MQTTset | 99% |  | [58] |
| SVM / RNN / LSTM | 2019 | BoT-IoT | 99.97% |  | [59] |
| DT / RF / Naïve Bayes / BayesNet | 2020 | BoT-IoT | 99.99% | 1.0 | [60] |
| AAE / BiGAN | 2022 | IoT-23 |  | 0.99 | [61] |
| LR / DT / RF / XGBoost / ANN | 2022 | IoT-23 | 99.99% | 1 | [62] |
| LGR / SVM / KNN / RF / XGBoost | 2020 | IoT-NI | 100% | 100% | [63] |

Table 2.1 Comparison Table of Past Research Literature

**2.8 --- critical discussion**

**2.9 Summary**

In conclusion, this chapter provides a considerable investigation into malicious endpoint detection and response using machine learning algorithms. In order to implement the detection of malicious traffic in IoT devices, one or more classification detection algorithms are required to analyse a large amount of endpoint data. It is also important to select the appropriate feature selection and data sampling algorithms to reduce data redundancy. Therefore, in previous studies, various machine learning optimisation algorithms have been proposed as solutions that can improve the performance of classifying malicious endpoints through continuous learning. However, the previous studies cannot be considered conclusive.

Although machine learning is becoming increasingly used in the field of cybersecurity, it also has some shortcomings. Because some studies have used unreliable datasets, which cannot be used to evaluate malicious endpoint detection frameworks for IoT network environments. Furthermore, some of the research objectives only distinguish between two categories of traffic data: normal and abnormal, without a detailed classification of the abnormal traffic data. Moreover, through literature review, it was found that some authors used Accuracy as a metric to evaluate the proposed model. However, it is more reasonable to use the Recall metric for malicious endpoint detection, as a higher Recall means a higher probability of the actual malicious data being predicted.

One of the tough challenges for all researchers in this domain is extracting relevant and valuable feature information from large-scale IoT traffic data. A variety of traffic features should be collected and analysed to tackle this challenge. And a novel approach is therefore needed for processing data and enabling detection of unknown attacks.

**Chapter 3**

**Requirement and Analysis**

**3.1 Project Overview**

The purpose of this research project is to build a malicious endpoint detection and response system. It is a novel, intelligent and fast proactive defence technology that follows an adaptive security architecture. Furthermore, machine learning algorithms are used to detect endpoint devices and find malicious activities, both known and unknown threats in a timely manner, and to quickly and intelligently classify and respond accordingly. In addition, the machine learning model is supposed to demonstrate excellent performance and robustness, as discussed in Section 2. It helps to maintain the accuracy of the determination when the models are confronted with small changes in the input data.

**3.2 System Requirements**

**Functional**

EDR is a technology designed and configured to secure endpoint devices that can detect and report unauthorised or anomalous phenomena in the system in a timely manner, and it is a technology used to detect violations of security policies in the network. Therefore the application of EDR systems requires the detection of intrusion attacks before they are directed at endpoint devices and the use of alarm and protection systems to expel intrusion attacks. During an intrusion attack, it is possible to reduce the damage caused by the intrusion attack.

In practice the trained model will be placed in the important network segment, monitoring the various packets in the segment constantly and characterising each packet or suspicious packet. If an abnormal endpoint is detected, the intrusion detection system can raise an alarm or even cut the network connection outright.

In addition, after an intrusion attack, it is not only necessary to be able to identify the type of attack, but also to display detailed information about the source of the attack, such as IP address and port number.

**Non-functional**

After reviewing and analysing previous research papers, several machine learning algorithms are decided to be implemented in this project to detect malicious endpoints. Each algorithm will be individually realized and tested on both binary and multi-classification problems, and the ultimate goal is to combine them to form an architecture for detecting malicious endpoints. An integrated learning approach is also required when necessary, as individual learners are prone to under-fitting or over-fitting, and in order to obtain a learner with good generalisation performance, multiple individual learners can be trained to form a strong learner through some combination of strategies. Finally, multiple algorithms will be trained and tested on real IoT endpoint data to ensure the usability and effectiveness of models in a realistic environment.

It is crucial to select a valid dataset for research in this area. In most studies supervised learning based machine learning algorithms are proposed and therefore the chosen dataset should be labelled, containing normal, abnormal and various attack categories. Furthermore, there may be cases of null or abnormal values in the dataset, so methods such as data cleaning should be exploited to mitigate the influence of such data. In addition, the data redundancy and sample imbalance issues mentioned in section 2.4 are also significant. In order to solve such problems, the use of multiple feature selection and data sampling methods will effectively alleviate the imbalance problem and thus improve the classification performance. In the machine learning modelling process, various model evaluation metrics need to be used for different problems. Therefore, appropriate evaluation criteria such as Accuracy, Precision, Recall, etc. should be proposed before designing the model to demonstrate how well the model performs in detecting malicious endpoints. The time cost of training and testing the model should also be recorded.

It is also important to consider how to build models efficiently. The use of the Python programming language and machine learning libraries (scikit-learn, Keras, TensorFlow) can help in the implementation of algorithms. This is because these machine learning development frameworks offer a wide choice of algorithms and support data pre-processing and model evaluation operations. To maintain a high level of productivity and efficiency in projects from development to deployment and then in operations and maintenance.

**3.3 Data Selection Analysis**

It is vital to choose a valid dataset for this study. A series of recent studies have indicated that the most cited intrusion detection datasets, which are KDD-99 and NSL-KDD, have been gradually replaced. Due to evolving network traffic patterns, traditional datasets cannot be applied to current network endpoint devices, so it is critical to identify a comprehensive network dataset that reflects current traffic patterns. With the proliferation of IoT devices in recent years, the large amount of data generated by endpoint devices has driven the technology development in the field of intrusion detection. Therefore, this project will focus on IoT datasets to perform research on malicious endpoint detection and response.

**MQTT-IoT-IDS2020**

MQTT-IoT-IDS2020 is the first dataset to be created using the MQTT transport protocol. There were 12 sensors, a simulated camera and other devices to simulate an MQTT-based IoT network architecture. While capturing data from these sensors which communicate with each other via the MQTT protocol, and recording five scenarios including normal operation, aggressive scan, UDP scan, Sparta, and MQTT brute-force attack. The MQTT-IoT-IDS2020 dataset contains three abstraction levels of features are Uni-Flow, Bi-Flow and Packet-Flow respectively. And Figure 3 illustrates the distribution of five scenarios in three flow features.

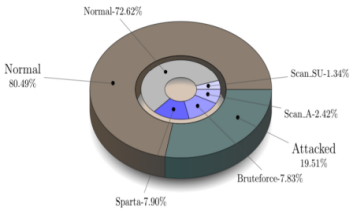
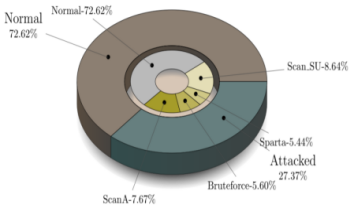
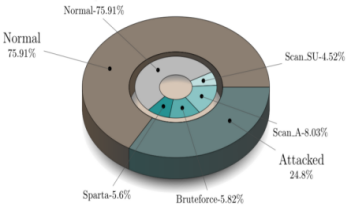


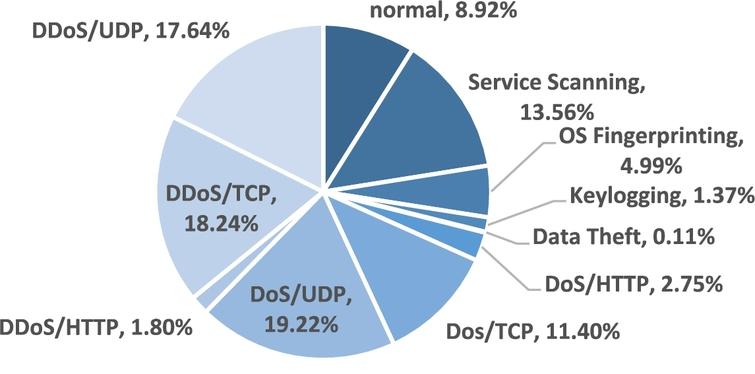
Figure 3.1Five files statistics of Uni-flow, Bi-flow and Packet-flow [52]

**MQTTset**

Similar to MQTT-IoT-IDS2020, the MQTTset dataset is composed of 8 different characteristics of MQTT sensors in a typical smart home, and the generated MQTT traffic is transformed into a PCAP file. The MQTTset dataset has six categories: Legitimate, Malformed, DoS attack, SlowITe, Bruteforce, and Flooding, so the dataset can reflect the real IoT environment.

**BoT-IoT**

The BoT-IoT dataset was built using Ubuntu VM to simulate an IoT network, and using the Node-red tool to simulate five IoT sensors to build a virtual IoT environment. And the MQTT protocol is used to transfer IoT messages to the cloud. The original dataset has 46 features and includes four categories respectively Normal, DoS, DDoS, Reconnaissance and Information Theft. However, the lack of monitoring of real IoT devices makes this dataset not really representative of IoT traffic [36] and it is clear from the Figure 3.2 that there is a significant sample imbalance problem in this dataset.



**Figure 3.2** The distribution of attack types of BoT-IoT dataset [36]

**IoT Network Intrusion (****IoT-NI)**

There were two IoT devices SKT NGU and EZVIZ Wi-Fi (C2C Mini O Plus 1080P) cameras used to create the IoT-NI dataset. The categories of the dataset are Normal, DoS, MITM, Mirai and Scan. With the exception of the Mirai botnet category, the other attack categories were constructed using packets collected when modelling network attacks using the Nmap software. However, it is clear from Figure 3.2 that the Mirai class dominates and the data sample is severely imbalanced.

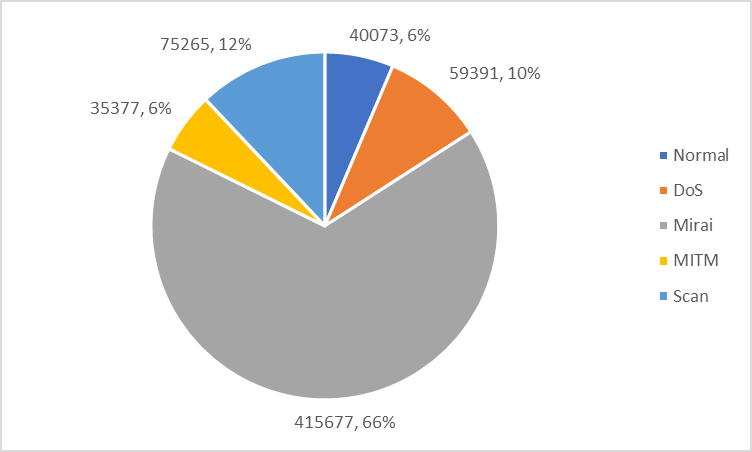


Figure 3.2 The distribution of attack types of IoT Network Intrusion dataset

**IoT-23**

Different from the BoT-IoT dataset, IoT-23's IoT traffic was collected from real hardware IoT devices. Amazon Echo devices, Philips Hue devices and Somfy door lock devices were used to capture this normal IoT packets. And malicious scenarios were created by executing malware using a Raspberry Pi again. There are 20 malicious scenarios and 3 benign scenarios in the IoT network included in the IoT-23 dataset. And Table 3.1 indicates the number of each category in the majority of labelled data.

|  |  |
| --- | --- |
| **Categories** | **Number** |
| C&C - File Download | 53 |
| C&C | 21995 |
| Benign | 30858735 |
| DDoS | 19538713 |
| C&C-Torii | 30 |
| File Download | 18 |
| Part-Of-A-Horizontal-PortScan | 213852924 |
| Attack | 9398 |
| C&C-Heart Beat | 33673 |
| C&C-Heart Beat File Download | 11 |
| C&C-Heart Beat Attack | 834 |
| C&C-Part-Of-A-Horizontal-PortScan | 888 |
| Okiru | 60990708 |
| Okiru-Attack | 3 |
| C&C-Mirai | 2 |
| Part-Of-A-Horizontal-PortScan-Attack | 5 |
| **Total** | 325307990 |

Table 3.1 The number of each category in the original dataset

**IoT-DS2**

Since an increasing number of IoT endpoints are targeted by malicious attacks. In order to tackle this challenge, a new dataset containing legitimate and malicious IoT network traffic, as well as various attack types need to be proposed. IoT-DS2 is a combination of five datasets - MQTT-IoT-IDS2020, MQTTset, BoT-IoT, IoT-NI, and IoT-23. The dataset captures complete network data containing a total of 83 features, 17 various scenarios were launched during the combination of the IoT-DS2 dataset. These scenarios are DDoS, Okiru, PortScan, Reconnaissance, Mirai, Sparta, MQQT\_bruteforce, Torii, C&C, DoS, Attack, Flood, HeartBeat, MITM ARP Spoofing, FileDownload, Theft, and Normal respectively. Moreover, this dataset includes 15 attack classes and one normal class that can be used to develop and accurately test malicious endpoint detection systems.

**3.4 Benchmark Datasets Comparison**

Valuable data contributes to training and evaluating the detection models, however there are a large number of datasets being proposed today for the detection of malicious attacks. Therefore, it is significant to review and compare the existing published datasets to find the best dataset representative for this project.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Dataset** | **Year Published** | **Features**  **Numbers** | **Categories**  **Numbers** | **IoT**  **Purpose** | **MQTT Protocol** | **Realistic Traffic** | **Labeled** |
| KDD-99 | 1999 | 41 | 5 | ✕ | ✕ | ✕ | ✓ |
| NSL\_KDD | 2009 | 43 | 5 | ✕ | ✕ | ✕ | ✓ |
| UNSW-NB15 | 2015 | 49 | 9 | ✕ | ✕ | ✓ | ✓ |
| CICIDS2017 | 2017 | 85 | 15 | ✕ | ✕ | ✓ | ✓ |
| BoT-IoT | 2018 | 45 | 5 | ✓ | ✕ | ✓ | ✓ |
| TON\_IoT | 2019 | 45 | 10 | ✓ | ✓ | ✓ | ✓ |
| IoT-NI | 2019 | 42 | 5 | ✓ | ✓ | ✓ | ✓ |
| MQTT-IoT-IDS2020 | 2020 | 5 | 5 | ✓ | ✓ | ✓ | ✓ |
| MQTTset | 2020 | 33 | 6 | ✓ | ✓ | ✓ | ✓ |
| IoT-23 | 2020 | 21 | 10 | ✓ | ✕ | ✓ | ✓ |
| IoT-DS2 | 2021 | 83 | 17 | ✓ | ✓ | ✓ | ✓ |

Table 3.3 Summary of comparison for different benchmark datasets

There are 11 dominant datasets listed and compared in a number of respects in Table 3.3. Most of the recent research has been based on IoT network datasets, which use packet-based and flow-based as well as a combination of both to detect the attacks on IoT endpoints. However, traditional datasets such as KDD-99, NSL\_KDD, UNSW-NB15 and CICIDS2017 do not contain any characteristics of IoT applications and therefore do not effectively detect the malicious endpoints. In terms of the TON\_IoT dataset, contains heterogeneous data sources and has various normal and attack events for IoT endpoints[65], but it has some limitations due to the lack of authentication and disconnection phases. Furthermore, some studies with specific purposes such as BoT-IoT and IoT-23 do not address the raw traffic data of the MQTT protocol, but rather focus more on the DNS traffic data in the loT context.

The other type of dataset is the MQTT protocol-based dataset, which is widely used to communicate data between sensors or endpoints and this type of dataset best fits the IoT application scenario [66]. Therefore, it is crucial to use real traffic datasets that not only contain basic details about the endpoint device but also include the traffic that sensors are transmitting and clearly document the type of attack so that a comprehensive evaluation of the detection algorithms for malicious endpoints can be carried out. To tackle this problem Ullah, I. and Mahmoud, Q.H. proposed to combine five datasets to form the IoT-DS2 dataset, which can capture most of the attack types and remedy the shortcomings of other datasets to a certain extent. In addition, the dataset monitors the internal behaviours of IoT endpoints accurately, therefore it was selected as a benchmark dataset for evaluating detection models. But it suffers from data imbalance, therefore, the accuracy of detection for part of the attack types is not high.

**3.5** **Composition of IoT-DS2 Dataset**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Categories** | **MQTT-IoT-2020** | **MQTTSET** | **BoT-IoT** | **IoT-23** | **IoT-NI** | **IoT-DS2** |
| 1 | DDoS | - | - | 17420085 | - | - | 500000 |
| 2 | Normal | - | - | - | 4253672 | - | 2000000 |
| 3 | Okiru | - | - | - | 12908506 | - | 500000 |
| 4 | Port Scan | - | - | - | 2000000 | - | 500000 |
| 5 | Reconnaissance | - | - | 946268 | - | - | 500000 |
| 6 | Mirai | - | - | - | - | 366971 | 366971 |
| 7 | Sparta | 1217198 | - | - | - | - | 500000 |
| 8 | MQQT Bruteforce | 2001972 | - | - | - | - | 500000 |
| 9 | Torii | - | - | - | 24492 | - | 24492 |
| 10 | C&C | - | - | - | 20612 | - | 20612 |
| 11 | DoS | - | - | - | - | 59391 | 59391 |
| 12 | Attack | - | - | - | 1699608 | - | 500000 |
| 13 | Flood | - | 77756 | - | - | - | 77756 |
| 14 | HeartBeat | - | - | - | 12648 | - | 12648 |
| 15 | MITM ARP Spoofing | - | - | - | - | 32909 | 32909 |
| 16 | File Download | - | - | - | 7707 | - | 7707 |
| 17 | Theft | - | - | 445799 | - | - | 445799 |
| 18 | Malformed | - | 3535 | - | - | - | 3535 |
| 19 | SlowITe | - | 3044 | - | - | - | 3044 |
| **Total** | | | | | | | 6554864 |

Figure 3.4 The Composition of IoT-DS2 Dataset

Tables 3.4 indicates the composition information of the dataset named IoT-DS2. The latest IoT-DS2 dataset is composed of five IDS evaluation datasets, which are MQTT-IoT-IDS2020, MQTTset, BoT-IoT, IoT-NI, and IoT-23. It contains 18 attack categories and 1 normal category, but there is a significant imbalance in the samples, so the Malformed and SlowITe categories will be removed from the dataset. Moreover, The IoT-DS2 column represents the number of instances extracted from the five datasets, and also suffers from data redundancy, so with the weights of the classes being adjusted in the final dataset to divide the number of instances in each class. This dataset is available from [67] and can be efficiently used for the development and testing of a malicious IoT endpoint detection system. Furthermore, a series of data pre-processing steps, such as data cleaning, feature transformation, feature selection, data sampling and data segmentation, are still required before training and testing the model formally. In addtion, feature selection is a vital stage in constructing a deep learning model, not only to reduce the number of features to reduce training time, but also to make the model more generalisable and reduce over-fitting.

**3.6 Classification Algorithms**

**Chapter 4**

**Design**

**4.1 System Architectural Design**

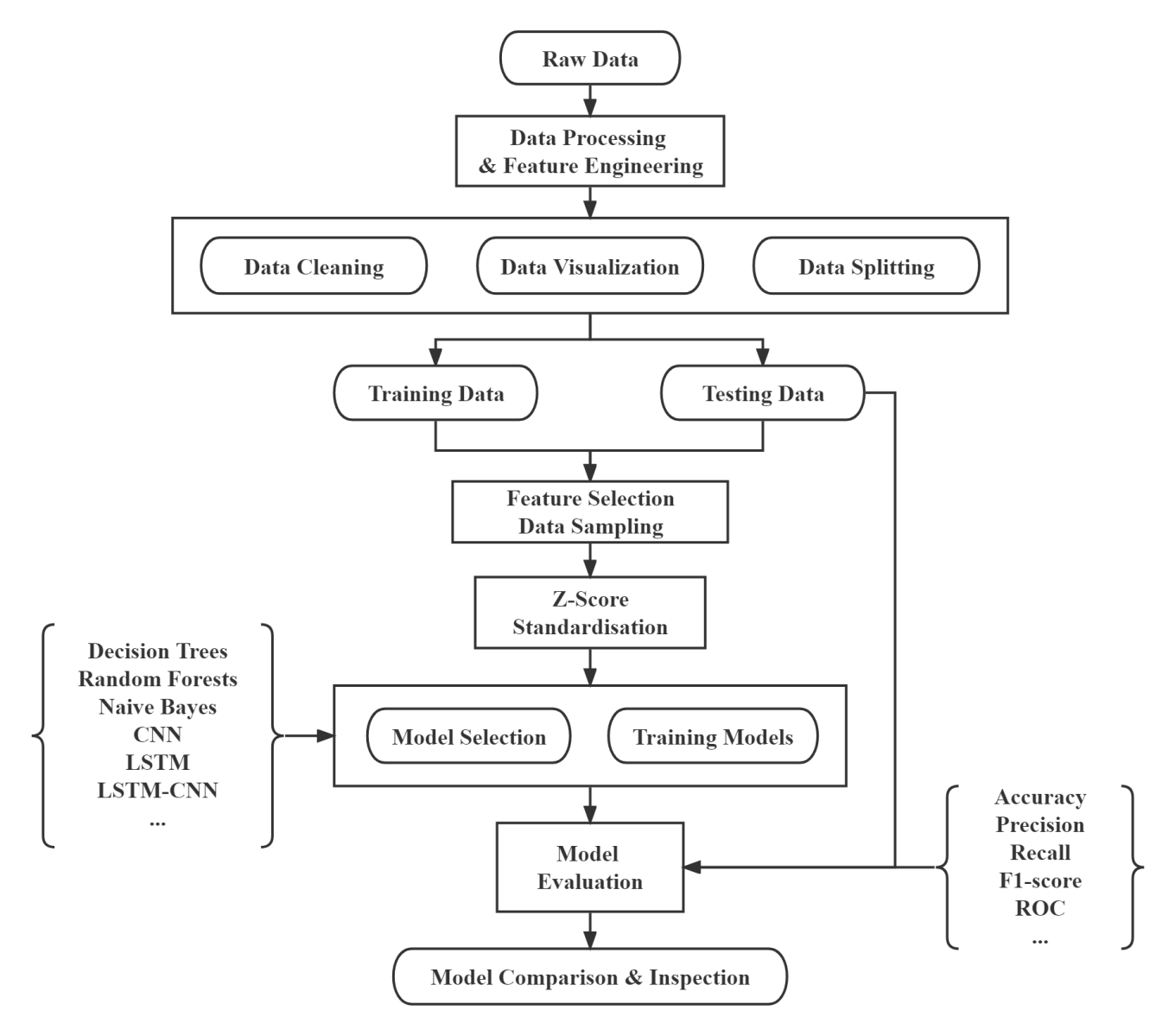
****

Figure 4.1 The structure design of malicious endpoint detection system

Figure 4.1 shows the overall framework of the system, from data processing to the creation of the ML model to the overall design of the actual prediction process. The first step in the process is to observe and analyse the collected data and perform data pre-processing operations. This involves data cleaning, data visualisation and feature replacement etc. The processed data is then randomly partitioned into a training set and a test set on a scale of 80% and 20% respectively. Feature engineering is then applied to the training set and data sampling techniques are used to address the problem of sample imbalance, and data normalisation required before training the model. Finally, the proposed models are compared with other models and a comprehensive evaluation is carried out using different evaluation metrics.

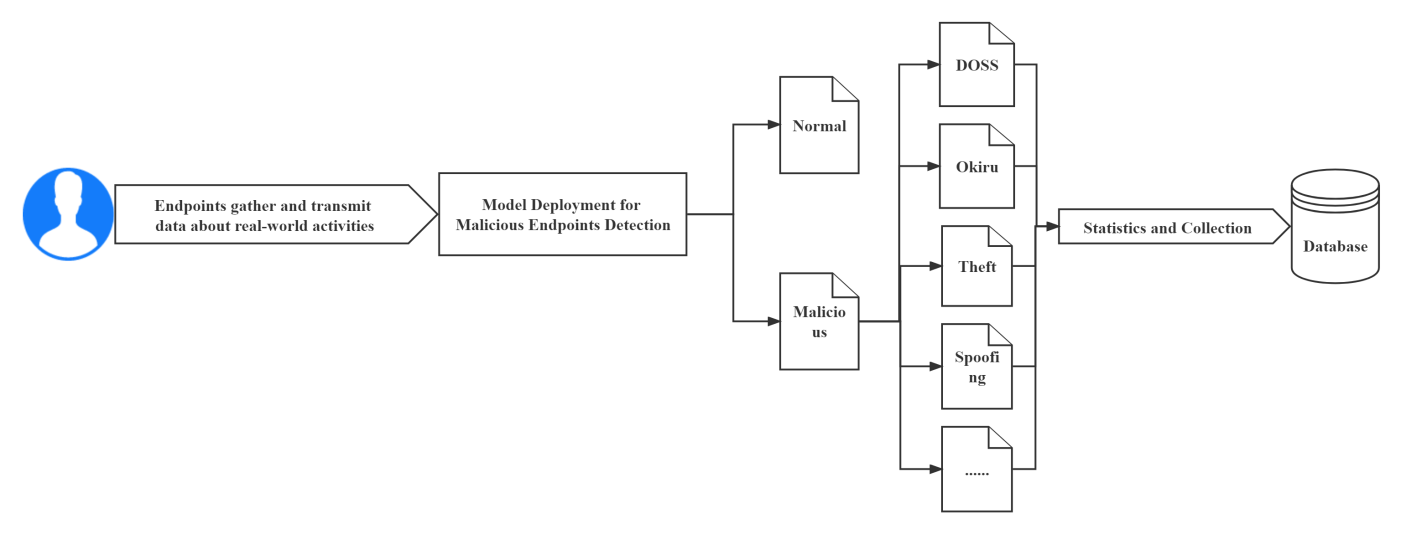
****

Figure 4.2 Real-time normal and malicious detection in endpoints

A practical application scenario of the EDR system is shown in Figure 4.2, which demonstrates the classification and prediction process of endpoint devices as it communicates and transmits data. The trained model should be deployed in each endpoint device to perform binary and multi-classification predictions for each data. In addition, it is important to respond promptly to statistics and output details of malicious endpoints and to store this information in a database to reduce the threat to sensitive data storage caused by malicious endpoints.

**4.2 Data Processing**

**4.2.1 Data Pro-processing and Cleaning**

|  |  |  |
| --- | --- | --- |
| **No** | **Features** | **Description** |
| 1 | Flow\_ID | Record the number of the transmitted data flow |
| 2 | Src\_IP | The source IP address of the host that is sending the packets |
| 3 | Src\_Port | The port number used by a program to send data to another program |
| 4 | Dst\_IP | The destination IP address of the host that is sending the packets |
| 5 | Dst\_Port | The port number used by a program to receive data from another program |
| 6 | Protocol | Textual representation of transaction protocols presents in network flow |
| 7 | Timestamp | Record the time that a traffic transaction occurred |
| 8 | Flow\_Duration | Record total duration |
| 9 | Pkts | Total count of packets in transaction |
| 10 | Max | Maximum duration of aggregated records |
| 11 | Min | Minimum duration of aggregated records |
| 12 | Mean | Average duration of aggregated records |
| 13 | Std | Standard deviation of aggregated records |
| 14 | Pkts/s | Count of packets in transaction per second |
| 15 | Byts/s | Count of bytes in transaction per second |
| 16 | Tot | Total duration of aggregated records |
| 18 | Var | Variance of aggregated records |
| 19 | Cnt | Flow state flags count seen in transactions |
| 20 | Ratio | Down or up ratio |
| 22 | Label | Class of label |
| 23 | Cat | Category of attack |

Table 4.1 The summary of utilized features

Data pre-processing is an efficient method of removing invalid data, irregular data and extremely erroneous data, so it is necessary to clean the data for a large dataset such as IoT-DS2. The IoT-DS2 dataset contains a total of 83 features, with 23 major feature types as shown in Figure 4.3. However, there are many missing and extreme values in the sample, for example, all of the features in 'Bwd\_IAT\_Mean.1' are missing values. If missing values in the data may reduce the statistical power of the research subjects or even lead to incorrect results due to biased estimation, which can have a significant impact on the modelling phase. Therefore, in order to eliminate anomalies in the original data, the first step is to find the very large, very small and missing values in the data set and interpolate them with the mean value. In terms of features with a large number of missing values, it is essential to count the percentage of missing values in each feature separately and set a threshold value and remove features that exceed that threshold.

**4.2.2 One-hot Encoding Converting**

Because the dataset contains categorical data, the categorical features need to be coded from texttypes to numeric types in order to fit and evaluate the model before splitting the dataset into a test set and a training set. As shown in Figure 4.3 the 'Label' feature contains two categories 'Normal' and 'Anomaly', while the 'Cat' feature has a total of 17 different categories represented as text types. Therefore it is important to perform One-hot encoding of the target data using the OneHotEncoder class from the Scikit-Learn library, which is the process of converting the labelled data into numerical features of the dataset for use in the neural network. In addition, the mapping of discrete features to Euclidean space by one-hot encoding solves the problem of classifiers not handling attribute data well and serves to expand the features to some extent.

**4.2.3 Data Splitting**

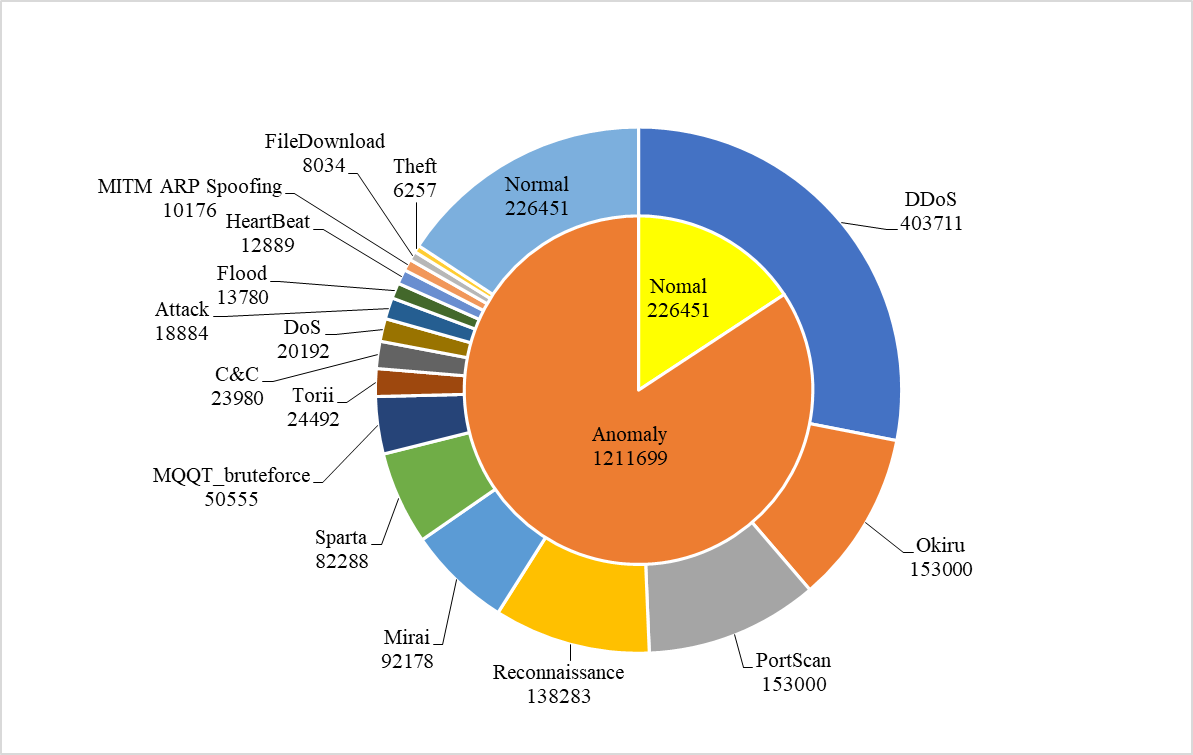


Figure 4.3 The number of different categories in IoT-DS2 dataset

**4.2.3 Features Selection**

**4.2.4 Data Sampling**

**4.2.5 Data Split**

**4.2.6 Standardization**

**4.3**

**4.2 Evaluation Methods**

**IoT-DS2 Dataset size，labels categories ...**

The number of rows in raw dataset 1438157

The number of columns in raw dataset 86

Anomaly 1211706

Normal 226451

DDoS 403711

Normal 226451

Okiru 153000

PortScan 153000

Reconnaissance 138283

Mirai 92185

Sparta 82288

MQQT\_bruteforce 50555

Torii 24492

C&C 23980

DoS 20192

Attack 18884

Flood 13780

HeartBeat 12889

MITM ARP

Spoofing 10176

FileDownload 8034

Theft 6257

Bwd\_IAT\_Mean.1 No values

查询各个特征中缺失值的百分比

设置阈值 缺失值超过70%则删除

特征转换

Anomaly - 1

Normal -0

['Normal', 'DDoS', 'PortScan', 'Okiru', 'Reconnaissance', 'Mirai', 'Sparta', 'MQQT\_bruteforce', 'Torii', 'C&C', 'DoS', 'Attack', 'Flood', 'HeartBeat', 'MITM ARP Spoofing', 'FileDownload', 'Theft']

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16]

modin.pandas 方法

**Chapter 5**

**Implementation and Testing**

**References**

1. Sinha, S. (2022). *State of IoT 2021: Number of Connected IoT Devices Growing 9% to 12.3 Billion globally, Cellular IoT Now Surpassing 2 Billion*. [online] IoT Analytics. Available at: <https://iot-analytics.com/number-connected-iot-devices/.>
2. Kim, S., Hwang, C. and Lee, T. (2020) ‘Anomaly based unknown intrusion detection in endpoint environments’, *Electronics (Basel)*, 9(6), pp. 1–21. doi:10.3390/electronics9061022.
3. Kirsch, C. (2016). *IDC: 70% of Successful Breaches Originate on the Endpoint | Rapid7 Blog*. [online] Rapid7. Available at: <https://www.rapid7.com/blog/post/2016/03/31/idc-says-70-of-successful-breaches-originate-on-the-endpoint/> [Accessed 8 Jun. 2022].
4. Chuvakin, A. (2013). *Named: Endpoint Threat Detection & Response.* [online] Availableat: <https://blogs.gartner.com/anton-chuvakin/2013/07/26/named-endpoint-threat-detection-response/> [Accessed 8 Jun. 2022].
5. Kaur, H. and Tiwari, R. (2021) ‘Endpoint detection and response using machine learning’, *Journal of Physics: Conference Series*, 2062(1), p. 12013. doi:10.1088/1742-6596/2062/1/012013.
6. Kumar, N., Kasbekar, G.S. and Manjunath, D. (2022) ‘Application of Data Collected by Endpoint Detection and Response Systems for Implementation of a Network Security System based on Zero Trust Principles and the EigenTrust Algorithm’.
7. Juma, N., Huang, X. and Tripunitara, M.V. (2020). Forensic Analysis in Access Control: Foundations and a Case-Study from Practice. *CCS*. doi:10.1145/3372297.3417860.
8. Cahill, D., & Poller, J. (2017). ‘An Adaptive and Layered Approach to Endpoint Security’, *The Enterprise Strategy Group*. [online] Available at: <https://www.bitdefender.co.th/media/wysiwyg/gravityzone/elite-security/ESG-White-Paper-Bitdefender-Jun-2017.pdf> [Accessed 9 Jun. 2022].
9. Badhwar, R. (2021). OEM and Third-Party Sourced Application and Services Risk.*TheCISO’s Next Frontier. Springer*, pp.335–344. doi:10.1007/978-3-030-75354-2\_41.
10. Yaffe, D. and Hendler, D. (2021) ‘Early Detection of In-Memory Malicious Activity Based on Run-Time Environmental Features’, *in Cyber Security Cryptography and Machine Learning. Cham: Springer International Publishing*, pp. 397–404. doi:10.1007/978-3-030-78086-9\_29.
11. Lee, S.-J., Shim, H.-Y., Lee, Y.-R., Park, T.-R., Park, S.-H. and Lee, I.-G. (2022). *Study on Systematic Ransomware Detection Techniques*. IEEE Xplore. doi:10.23919/ICACT53585.2022.9728909.
12. Scott, J. (2017). *Signature Based Malware Detection is Dead*. Institute for Critical Infrastructure Technology. Available at: <https://informationsecurity.report/Resources/Whitepapers/920fbb41-8dc9-4053-bd01-72f961db24d9_ICIT-Analysis-Signature-Based-Malware-Detection-is-Dead.pdf>.
13. Rakes, T.R., Deane, J.K. and Paul Rees, L. (2012) ‘IT security planning under uncertainty for high-impact events’, *Omega (Oxford)*, 40(1), pp. 79–88. doi:10.1016/j.omega.2011.03.008.
14. Ghafir, I. et al. (2018) ‘Detection of advanced persistent threat using machine-learningcorrelation analysis’*, Future generation computer systems*, 89, pp. 349–359. doi:10.1016/j.future.2018.06.055.
15. Aleroud, A. and Zhou, L. (2017) ‘Phishing environments, techniques, and countermeasures: A survey’, *Computers & security*, 68, pp. 160–196. doi:10.1016/j.cose.2017.04.006.
16. Jain, A.K. and Gupta, B.B. (2022) ‘A survey of phishing attack techniques, defence mechanisms and open research challenges’, *Enterprise information systems*, 16(4), pp.527–565. doi:10.1080/17517575.2021.1896786.
17. Kumar, M.S., Ben-Othman, J., Srinivasagan, K.G. and Krishnan, G.U. (2019). *Artificial Intelligence Managed Network Defense System against Port Scanning Outbreaks*. IEE Xplore. doi:10.1109/ViTECoN.2019.8899380.
18. Sikder, A.K. et al. (2021) ‘A Survey on Sensor-Based Threats and Attacks to Smart Devices and Applications’, *IEEE Communications surveys and tutorials*, 23(2), pp. 1125–1159. doi:10.1109/COMST.2021.3064507.
19. Šemić, H. and Mrdovic, S. (2017). *IoT honeypot: A multi-component solution for handling manual and Mirai-based attacks*. IEEE Xplore. doi:10.1109/℡FOR.2017.8249458.
20. Gandotra, E., Bansal, D. and Sofat, S. (2014) ‘Malware Analysis and Classification: A Survey’, *Journal of information security*, 5(2), pp. 56–64. doi:10.4236/jis.2014.52006.
21. Yang Xin et al. (2018) ‘Machine Learning and Deep Learning Methods for Cybersecurity’, *IEEE access*, 6, pp. 35365–35381. doi:10.1109/ACCESS.2018.2836950.
22. Boser, B.E., Guyon, I.M. and Vapnik, V.N. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the fifth annual workshop on Computational learning theory - COLT ’92*. Association for Computing Machinery, 144-152. doi:10.1145/130385.130401.
23. Gonzalez-Abril, L. et al. (2017) ‘Handling binary classification problems with a priority class by using Support Vector Machines’, *Applied soft computing*, 61, pp. 661–669 doi:10.1016/j.asoc.2017.08.023.
24. Quinlan, J.R. (1986) ‘Induction of decision trees’, *Machine learning*, 1(1), pp. 81–106 doi:10.1023/A:1022643204877.
25. Quinlan, J.R. et al. (1993) *C4.5 : programs for machine learning*. San Mateo, California: Morgan Kaufmann Publishers.
26. Breiman, L. (1984) *Classification and regression trees.* Boca Raton, Florida: Chapman & Hall : CRC.
27. Tin Kam Ho (1995). *Random decision forests*. [online] IEEE Xplore. doi:10.1109/ICDAR.1995.598994.
28. Ambwani, T. (2003) ‘Multi class support vector machine implementation to intrusion detection’, in *Proceedings of the International Joint Conference on Neural Networks, 2003.* IEEE, pp. 2300–2305 vol.3. doi:10.1109/IJCNN.2003.1223770.
29. Ronghua Tian et al. (2010) ‘Differentiating malware from cleanware using behavioural analysis’, in *2010 5th International Conference on Malicious and Unwanted Software.* IEEE, pp. 23–30. doi:10.1109/MALWARE.2010.5665796.
30. Lee, Y.-J. and Mangasarian, O.L. (2001). RSVM: Reduced Support Vector Machines. *Proceedings of the 2001 SIAM International Conference on Data Mining*. pp.1-17doi:10.1137/1.9781611972719.13.
31. Chiu, C.-Y. et al. (2010) ‘Semi-supervised Learning for False Alarm Reduction’, in *Advances in Data Mining. Applications and Theoretical Aspects.* Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 595–605. doi:10.1007/978-3-642-14400-4\_46.

1. Ioannou, C. and Vassiliou, V. (2019). Classifying Security Attacks in IoT Networks Using Supervised Learning. *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pp. 652-658. doi:10.1109/dcoss.2019.00118.
2. Ioannou, C. and Vassiliou, V. (2021) ‘Network attack classification in iot using support vector machines’, *Journal of sensor and actuator networks*, 10(3), p. 58. doi:10.3390/jsan10030058.
3. Hasan, M. et al. (2019) ‘Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches’, *INTERNET OF THINGS*, 7, p. 100059. doi:10.1016/j.iot.2019.100059.
4. M. Pahl and F. Aubet. (2018) ‘All Eyes on You: Distributed Multi-Dimensional IoT Microservice Anomaly Detection’, *2018 14th International Conference on Network and Service Management (CNSM)*, pp. 72-80.
5. Alothman, Z., Alkasassbeh, M. and Al-Haj Baddar, S. (2020) ‘An efficient approach to detect IoT botnet attacks using machine learning’, *Journal of high speed networks*, 26(3), pp. 241–254. doi:10.3233/JHS-200641.
6. Varma P, R.K., Mallidi, S.K.R. and Muni, R.R. (2021). A Wrapper based Feature Selection using Grey Wolf Optimization for Botnet Attack Detection. *International Journal of Sensors, Wireless Communications and Control*, 11(9). pp .951-956. doi:10.2174/2210327911666210120124340.
7. Ongun, T. *et al.* (2019) ‘On Designing Machine Learning Models for Malicious Networ k Traffic Classification’.
8. Machaka, P., Ajayi, O., Maluleke, H., Kahenga, F., Bagula, A. and Kyamakya, K. (2021). Modelling DDoS Attacks in IoT Networks using Machine Learning. *arXiv:2112.05477 [cs]*.
9. Abiodun, O.I. *et al.* (2018) ‘State-of-the-art in artificial neural network applications: A survey’, *Heliyon*, 4(11), pp. e00938–e00938. doi:10.1016/j.heliyon.2018.e00938.
10. Huang, T.-J. (2017) ‘Imitating the Brain with Neurocomputer A ＂New＂ Way Towards Artificial General Intelligence’, *International journal of automation and computing*, 14(5), pp. 520–531. doi:10.1007/s11633-017-1082-y.
11. Xin, Y. et al. (2018) ‘Machine Learning and Deep Learning Methods for Cybersecuri y’, *IEEE access*, 6, pp. 35365–35381. doi:10.1109/ACCESS.2018.2836950.
12. Gang Liu, Fen Hu and Wei Chen (2010) ‘A neural network ensemble based method for detecting computer virus’, in *2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering.* IEEE, pp. 391–393. doi:10.1109/CMCE.2010.5610520.
13. Nagisetty., A., & Gupta, G.P. (2019). Framework for Detection of Malicious Activities in IoT Networks using Keras Deep Learning Library. *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, 633-637.
14. Ji, S.-Y. et al. (2016) ‘A multi-level intrusion detection method for abnormal network behaviors’, *Journal of network and computer applications*, 62, pp. 9–17. doi:10.1016/j.jnca.2015.12.004.
15. Donghong, S. et al. (2014) ‘Analysis of Network Security Data Using Wavelet Transforms’, *Journal of algorithms & computational technology*, 8(1), pp. 59–70. doi:10.1260/1748-3018.8.1.59.
16. Li, H. and Ding, X. (2017) ‘Research on artificial neural network intrusion detection photochemistry based on the improved wavelet analysis and transformation’, in *AIP Conference Proceedings. Melville: American Institute of Physics*. doi:10.1063/1.4977374.

1. Gao, M. et al. (2020) ‘Malicious network traffic detection based on deep neural networks and association analysis’, *Sensors (Basel, Switzerland)*, 20(5), p. 1452. doi:10.3390/s20051452.
2. Cheng, Y. et al. (2018) ‘Model Compression and Acceleration for Deep Neural Networks: The Principles, Progress, and Challenges’, *IEEE signal processing magazine*, 35(1), pp. 126–136. doi:10.1109/MSP.2017.2765695.
3. Lee, W. Y., Saxe, J., & Harang, R. (2019). SeqDroid: Obfuscated Android malware detection using stacked convolutional and recurrent neural networks. In *Deep learning applications for cyber security*  pp. 197-210.
4. Tavallaee, M. *et al.* (2009) ‘A detailed analysis of the KDD CUP 99 data set’, in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*. IEEE, pp. 1–6. doi:10.1109/CISDA.2009.5356528.
5. Khan, Muhammad Almas et al. (2021) ‘A deep learning-based intrusion detection system for mqtt enabled iot’, *Sensors (Basel, Switzerland)*, 21(21), p. 7016. doi:10.3390/s21217016.
6. Hindy, H. et al. (2020) ‘Machine Learning Based IoT Intrusion Detection System: An MQTT Case Study (MQTT-IoT-IDS2020 Dataset)’.
7. Ullah, I. and Mahmoud, Q.H. (2022) ‘An Anomaly Detection Model for IoT Networks based on Flow and Flag Features using a Feed-Forward Neural Network’, in *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, pp. 363–368. doi:10.1109/CCNC49033.2022.9700597.
8. Vaccari, I. et al. (2020) ‘MQTTset, a New Dataset for Machine Learning Techniques on MQTT’, *Sensors (Basel, Switzerland)*, 20(22), p. 6578. doi:10.3390/s20226578.
9. Shalaginov, Andrii, Semeniuta, Oleksandr and Alazab, Mamoun (2019), MEmachine learning: Resource-Aware MQTT-Based Machine Learning for Network Attacks Detection on IoT Edge Devices. *Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing Companion - UCC ’19 Companion.* pp. 123-128.
10. Hariprasad Siddharthan, T. Deepa and Prabhu Chandhar (2022) ‘SENMQTT-SET: An Intelligent Intrusion Detection in IoT-MQTT Networks Using Ensemble Multi Cascade Features’, *IEEE access*, 10, pp. 33095–33110. doi:10.1109/ACCESS.2022.3161566.
11. Idrissi, Idriss, Azizi, Mostafa and Moussaoui, Omar (2022), An Unsupervised Generative Adversarial Network Based-Host Intrusion Detection System for Internet of Things Devices. *Indonesian Journal of Electrical Engineering and Computer Science*, 25(2): 1140.
12. Koroniotis, N. et al. (2019) ‘Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset’, *Future generation computer systems*, 100, pp. 779–796. doi:10.1016/j.future.2019.05.041.
13. Shafiq, M. et al. (2020) ‘Selection of effective machine learning algorithm and Bot-IoT attacks traffic identification for internet of things in smart city’, *Future generation computer systems*, 107, pp. 433–442. doi:10.1016/j.future.2020.02.017.
14. Abdalgawad, N. et al. (2022) ‘Generative Deep Learning to Detect Cyberattacks for the IoT-23 Dataset’, *IEEE access*, 10, pp. 6430–6441. doi:10.1109/ACCESS.2021.3140015.
15. Oha, C.V. et al. (2022) ‘Machine Learning Models for Malicious Traffic Detection in IoT Networks /IoT-23 Dataset’, in *Machine Learning for Networking.* Cham: Springer International Publishing, pp. 69–84. doi:10.1007/978-3-030-98978-1\_5.
16. Liu, Zhipeng, Thapa, Niraj, Shaver, Addison, Roy, Kaushik, Yuan, Xiaohong and Khorsandroo, Sajad (2020), Anomaly Detection on IoT Network Intrusion Using Machine Learning. *IEEE Xplore*: pp. 1–5.
17. Labs, Arkose (2020), Arkose Labs Reveals 20% Spike in Fraud as Digital Behavior Shifts during COVID-19. *GlobeNewswire News Room*. available at https://www.globenewswire.com/news-release/2020/04/28/2023282/0/en/Arkose-Labs-Reveals-20-Spike-in-Fraud-as-Digital-Behavior-Shifts-During-COVID-19.htmachine learning [14 July 2022].
18. Abdullah Alsaedi et al. (2020) ‘TON\_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems’, *IEEE access*, 8, pp. 165130–165150. doi:10.1109/ACCESS.2020.3022862.
19. Khan, Muhammad Almas et al. (2021) A Deep Learning-Based Intrusion Detection System for MQTT Enabled IoT, *Sensors (Basel, Switzerland)*, 21(21), p. 7016. doi:10.3390/s21217016.
20. Ullah and Q. H. Mahmoud (2021), Ontario Tech University. *sites.google.com*. available at <https://sites.google.com/view/iotdataset1>