An approach system detection intrusion for an IOT-based learning system

ADMEUR SMAIL [1*], ALAOUI SOUAD[2], HADDANI OUTMAN[3], AMJAD SOUAD[4] and ATTARIUAS HICHAM[5]

[1*] Abdelmalek Essaadi University, Tetouan, Morocco
* s.admeur@uae.ac.ma
[2]Sidi Mohamed Ben Abdellah University, Fes, Maroc
alsouad@gmail.com
[3]Abdelmalek Essaadi University, Tetouan, Morocco
Haddani.outman@gmail.com
[4]Abdelmalek Essaadi University, Tetouan, Morocco
amjad_souad@yahoo.fr
[5]Abdelmalek Essaadi University, Tetouan, Morocco
attariuas.hicham@gmail.com

Abstract:The Internet of Things (IoT) is a network of objects connected to the Internet, which enable data to be collected, shared and used. They are often low-powered devices with limited resources, making them vulnerable to a variety of attacks due to their interconnected nature and lack of network security or data leakage. So, detecting and preventing intrusions into an IoT environment has become paramount. This work creates an Intrusion Detection System (IDS) based on two Machine Learning techniques. The reduction of the dimensionality algorithm method concerning the sample selection (SS) of our system was identified by comparing the vector machine (SVM) and the multilayer perceptron (MLP). These results led us to consider SS techniques for the MLP classifier in order to fill this gap and further improve performance. Indeed, the results exceeded those of SVM. This proves the effectiveness of SS methods in increasing generalization capacity. We carried out a thorough and comprehensive study of the descriptive statistics of the data. As a result, we were able to detect dependency relationships between variables, while categorizing them. This analysis enabled us to identify the most important variables. By applying SVM to the variables selected in the previous step (descriptive statistics), we were finally able to maintain good performance while significantly reducing computational costs.

Keywords: Internet of Things, intrusion detection system, MLP, SVM, SS, Kddcup'9.

1. Introduction

Today, data security and confidentiality are one of the main concerns of the IoT, which represents the interconnection of physical objects with the Internet, enabling data to be collected, exchanged and analyzed in real time. This makes them more vulnerable to various attacks [3].

This is where IoT intrusion detection comes in. Intrusion detection systems (IDS) are designed to continuously monitor network activities and their events [4], connected objects, detect suspicious or malicious behavior, and take action to prevent or mitigate attacks to defend against intruders [5], with acceptable accuracy while minimizing energy consumption in limited resources [6]. There are different types of Intrusion Detection Systems (IDS) used to detect suspicious or malicious activity in computer networks, including signature based, anomaly based, machine learning, network

M. Khaldi et al. (eds.), *Proceedings of the E-Learning and Smart Engineering Systems (ELSES 2023)*, Atlantis Highlights in Social Sciences, Education and Humanities 14,
https://doi.org/10.2991/978-94-6463-360-3_35

malicious activity in computer networks, including signature-based, anomaly-based, machine learning, network-based,...

The aim of our work is to create an intrusion detection system (IDS) based on two techniques: feature selection and classification methods are two important concepts in the field of machine learning and data analysis. They are often used together to solve classification and prediction problems, which are used to select the most relevant and informative variables or features in order to reduce the dimensionality of data.

In this context, the database of the DARPA project, KDD-CUP 99 [19, 20], was analyzed in order to discover the various anomalies existing in the database, then to process them via learning methods.

Indeed, a great deal of effort has gone into this. First of all, cleaning and pre-processing the database. Indeed, the first difficulties began to appear with missing variables, duplicates and outliers. Secondly, a

study was devoted to the analysis of variable type, i.e., continuous or discrete, qualitative or quantitative. Finally, ordinal or nominal. This problem enabled us to analyze and categorize all these variables. Given the large number of examples and attributes, we carried out two main analyses: the first concerned dimensionality reduction. In this case, we proposed an increasing method for selecting relevant attributes via the SVM learning system. We used the cross-validation method to find the optimal number of capable attributes, to reduce computational cost and maintain classifier performance.In the second analysis, we proceeded by reducing the number of examples, using MLP example selection techniques. The aim is to focus training on the most relevant examples, capable of offering a better classification frontier.

2. Architecture IoT

The Internet of Things (IoT) is a collection of numerous objects that enable people as well as objects to interact and create intelligent environments such as transportation, agriculture, healthcare, energy, cities, etc. [1,2].

It's important to note that IoT architecture can vary according to the specific needs of each deployment. Some architectures may include additional layers, or combine several layers into a single one.

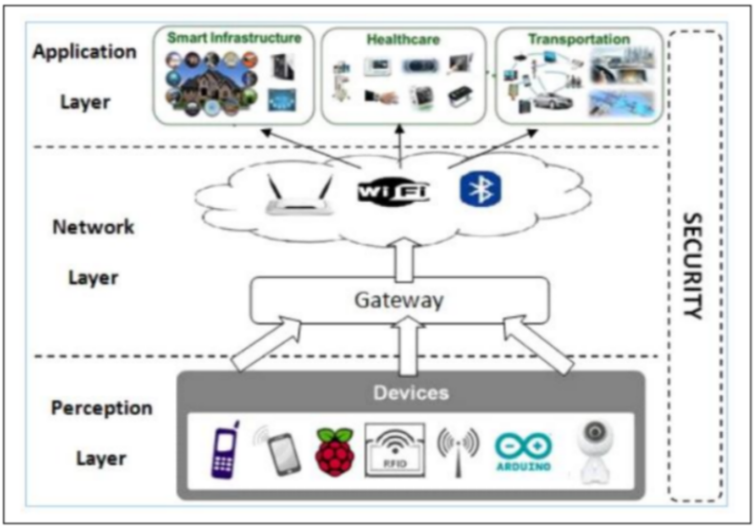

Figure 1: 3-layer IOT architecture.

Figure 1, presents the IoT architectural model composed of 3 layers that function as an overall structure that enables the connectivity, communication and management of smart objects within an IoT network, composed of interconnected layers that enable the collection, processing and exchange of data between IoT devices. It comprises :

- The perception layer consists of end devices that collect data from the physical universe. Digital applications can then analyze the collected data. As this layer remains in contact with real-world objects, it is the physical layer.
- The network/data transport layer: This layer manages the transport of data between all the layers of an IoT architecture. This layer also defines the network topology for the entire network of devices, cloud applications and databases.
- The application layer represents the specific applications that use IoT data to deliver services and functionality to end users. These can be applications in a variety of fields, such as home automation, connected health, energy management, logistics, smart agriculture, etc.

It should be noted that IoT architecture can vary according to requirements , data security and confidentiality are key considerations in IoT architecture, and appropriate security mechanisms must be put in place at each layer to protect data and devices from potential threats.

3. Internet of Things security

IoT security is an ongoing challenge, and a major concern given the growing number of connected devices and the potential risks associated with these systems, as new vulnerabilities and threats

constantly emerge. A necessary st security measure, which integrates security measures at all layers of the IoT architecture, is essential to prevent attacks and ensure user confidence in these connected systems, with manufacturers, service providers and users all playing an important role in promoting IoT security."[7]

Therefore, here are some key aspects of IoT security:

- Authentication and authorization: It's essential to adopt strong authentication and authorization mechanisms to verify the identity of devices, users and IoT services. This ensures that only authorized devices can access system data and functionality.
- Network security: IoT networks must be secure to prevent unexpected attacks
- IoT data collection and use: data must be anonymized, encrypted and stored securely.
- Physical security: IoT devices must be physically protected against unauthorized access. This may include physical locking mechanisms, protection against theft or destruction.
- Updates: IoT devices must be regularly updated with the latest security patches to address known vulnerabilities.

4. Intrusion detection system

The Intrusion Detection System (IDS) is a security tool used to monitor and detect suspicious or malicious activity on a network or computer systems, [4]. Its main objective is to identify intrusion attempts or abnormal behavior that could indicate a security breach. In general, there are two main types of intrusion detection system:

- Signature-based IDS: This type of IDS examines network traffic or system logs for patterns corresponding to pre-established signatures of known malicious behavior. Signatures are specific patterns that correspond to previously identified attacks or suspicious activity. When a match is found, the IDS generates an alert to report the suspicious activity.
- Behavior-based IDS: This type of IDS monitors network traffic or system logs for abnormal behavior that could indicate an attack in progress. Rather than relying on specific signatures, it uses algorithms and behavior patterns to establish a normal network or system profile. Any activity that deviates from the expected behavior is considered suspicious and triggers an alert.

Here are some common features of intrusion detection systems [6]:

- Network traffic monitoring: IDSs analyze network traffic for suspicious patterns or behavior, such as attempted port scans, DDoS attacks, intrusion attempts, etc.
- Anomaly detection : IDSs use machine-learning techniques to detect abnormal behavior that could indicate malicious activity. This can include detecting anomalies in traffic patterns, packet rates, resource utilization patterns, etc.
- Alert generation and incident response: When suspicious activity is detected, the IDS generates alerts to inform security administrators. Alerts can take the form of e-mail notifications, system messages or management console notifications.

The aim of an intrusion detection system is to improve network or system security by identifying suspicious activity as quickly as possible.

4.1.Types of intrusion detection systems

The various intrusion detection systems available can be classified according to several criteria

(see Figure 2), which are :Intrusion detection systems (IDS) can be classified into several types according to their operation, location and deployment. Figure 2 shows the detailed architecture according to these criteria

- Its place in the protected (or monitored) system
- Detection method.
- System behavior after detection.
- Data source.
- Frequency of use

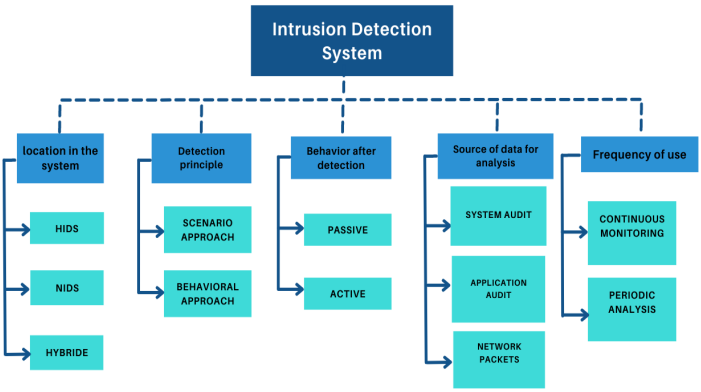

Figure 2: Classification of intrusion detection systems

Depending on their location in the computer system, as well as their data sources, intrusion detection systems can be classified into three broad categories, depending on whether or not they focus on monitoring:

- Machine activity: i.e. host-based intrusion detection system, HIDS. It ensures host security.
- Network traffic: i.e. network based on the intrusion detection system, NIDS, ensuring network security.
- A specific application on the machine: i.e. application-based IDS, also known as hybrid IDS.

4.1.The detection principle

IDSs have two different approaches [14] to detecting intrusions: the scenario approach and the behavioral approach.

The former approach defines a set of system behaviors that violate the security policy, and searches for patterns or signatures corresponding to these scenario attacks, while analyzing the collected data. However, the latter relies on defining a set of reference system behaviors. It then compares the system's activity with its normal profiles (applications), which have been previously rejected, in order to detect deviations considered as intrusions. We will briefly present these

different approaches, outlining their limitations. Each of these approaches can lead to false positives or false negatives.

2.4.1. Scenario approach (anomaly detection)

The scenario-based approach is an anomaly detection-based intrusion detection method useful for detecting unknown attacks or abnormal behavior that does not match pre-established signatures. It involves creating scenarios or models of normal behavior for systems or networks, then continuously monitoring activities to detect deviations from these scenarios. However, it is important to regularly update normal behavior scenarios to reflect changes in the system or network, in order to minimize false alarms and ensure accurate anomaly detection,[15] So far, the methods and their effect proposed are as follows:

- Expert systems: the expert system contains a set of rules describing attacks. Audit events are translated into facts. These facts have a semantic meaning for the expert system. Its inference engine then decides whether or not a listed attack has occurred. More recent tools no longer use it.
- Genetic algorithms: use genetic algorithms to search for attacks in audit trails. Each individual in the population encodes a particular subset of attacks potentially present in the audit trail. According to the audit file, the value of each individual is relative to the degree of realism of the hypothesis it encodes. The GASSATA tool is an example.
- Pattern Matching: this is today's most popular method. Signature attacks are provided, at different semantic levels depending on the tool. Various algorithms are used to locate these signatures in audit trails. However, this approach has two drawbacks: only known attacks can be detected, and the database of attack scenarios needs to be updated very frequently.

2.4.2. Behavioral approach (abuse detection)

This approach, proposed by J.P. ANDERSON [16], and extended by D.E. DENNING [17], uses methods based on the assumption that exploitation of a system vulnerability implies abnormal use of the system. Consequently, an intrusion can be identified as a deviation from a user's usual behavior. The main function of the behavioral IDS is to detect anomalies; its evolution requires a learning phase, in which the tool will learn "normal" behavior. Consequently, each flow and its usual behavior must be declared; the IDS will trigger an alarm if an abnormal flow is detected, and will not be able to specify the criticality of the possible attack. Behavioral IDS appears much later than signature identifiers. However, they do not benefit from their maturity. As a result, using such an IDS can be tricky, as the alarms triggered will contain a large number of false alarms.This problem can be solved by generalizing flow reporting. However, in doing so, this may lead to IDS transparency with regard to IDS detection. The normal behavior of a user or application (profile) can be constructed in various ways. The intrusion detection system compares current activity with the profile. Any deviant behavior is considered intrusive. The most striking methods proposed for building profiles are the following:

- St ti ati s l ca the d oths: e pfilo i e sl calct u al e rom i vabla es coids elrte bo e rdan omdan salmp e

(e.g. a mixture model) is then used to construct the distribution of each variable and to measure, by a synthetic quantity, the rate of deviation between current and past behavior. The NIDES Tool uses ThisMethod, among others.

- Neural networks: the technique consists in teaching a neural network the normal behavior of a user [18,19]. Subsequently, once we have provided it with the current actions, it will have to decide on their normality. The HyperView tool includes a module of this type, and several working searches follow the same method. This last point remains promising. However, it is not very industrialized.

The behavioral approach makes it possible to detect previous unknown attacks and privilege abuse by legitimizing the system's users. However, as the reference behavior is never exhaustive, there is always a risk of false alarms (false positives). What's more, if attacks have been committed during the learning phase, they will be considered normal (risk of false negatives). Both approaches have their advantages and disadvantages. This is why a hybrid approach seems essential.

4.3 Effectiveness of intrusion detection systems

The effectiveness of an intrusion detection system can vary depending on several factors, such as the quality of implementation, technology selection, system coverage, and incident response capability, It is determined by the following measures[5,6] :

- Reliability: the alerts generated must be justified, and no intrusion must stop them.
- Accuracy: the intrusion detection system does not specify whether it considers legitimate user actions to be atypical or intrusive.
- Performance: The performance of an intrusion detection system is measured by the rate at which audit trails are processed. If the intrusion detection system's performance is poor, real-time detection is not possible.
- Perfection: an intrusion detection system is imperfect if it fails to detect an attack.
- Timeliness: To enable rapid reaction in the event of an attack, an intrusion detection system needs to perform and propagate its analysis quickly.
- Reactivity: An IDS must be able to detect new types of attack as quickly as possible. To achieve this, it must be constantly updated. Automatic update capabilities are therefore virtually essential.
- Ease of implementation and adaptability: an IDS must be easy to implement. More importantly, it must be adaptable to the context in which it is to evolve. There's no point in having an IDS that issues alerts in under 10 seconds, if the resources needed to react are not available to act within the same time constraints.

4.3.IDS activity diagram

The activity diagram represents a general view of the intrusion detection process and may vary depending on the architecture and specific features of the intrusion detection system

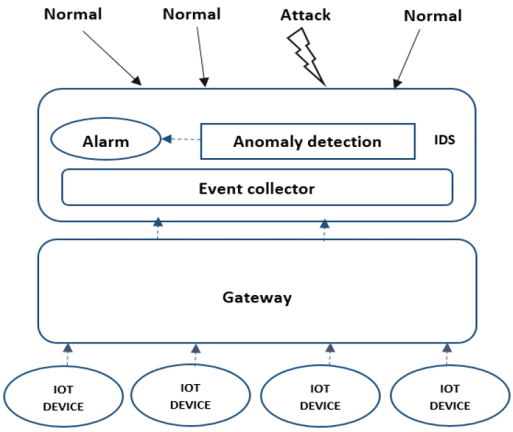

Figure 3: IDS business model

The figure is spread over 3 levels, as shown below:

- Data collection: The intrusion detection system collects data from various network objects, monitoring data, etc.
- Data pre-processing: The collected data is pre-processed to make it usable. This can include steps such as normalization, noise filtering, dimensionality reduction, etc.
- Anomaly analysis: Pre-processed data is analyzed to detect anomalies. This may involve the use of statistical techniques, behavioral models, machine learning methods, etc.
- Intrusion detection: Based on anomaly analysis, the system identifies suspicious or malicious activity that may indicate an intrusion in progress or an attempted intrusion.
- Alert generation and incident response: When an intrusion is detected, the system generates alerts to inform security administrators or incident response teams. Alerts can contain information on the nature of the intrusion, the resources involved and the urgency of the situation.

4.3. Experiments and results

4.3.2. Database

As previously mentioned, the implementation of the behavioral approach adopted in the principle of intrusion detection always includes a learning phase during which the IDS will "discover" the "normal" functioning of the monitored elements. In fact, the learning phase requires a database that is both sound and comprehensive comparing to the expected behavior of users in the real environment. Since 1999, KDD-Cup 99 [21] has been the most widely used data set for evaluating anomaly detection methods.

KDD CUP 99 is the dataset used in the third international Knowledge Discovery and Data Mining (KDD) tool competition [22]. This latter was held in conjunction with KDD 99, the fifth

data integration stage. This means, gathering all network connections to build the database.

4 .3.3. Assessment parameters

The detection and identification of attacks or non-attack behavior can be generalized a follows[6]:

- True positive (TP): the number of attacks detected when it is in fact an attack.
- True negative (TN): the normal number detected when it is in fact normal.
- False positive (FP): the number of attacks detected when it is in fact normal, namely false alarms
- False negatives (FN): the number of the normal detected when it is actually an attack, namely the attacks that can be detected by the intrusion detection system.
- Accuracy: can be defined as follows:

$$\text{Precision} = [(T\,P + T\,N) / (T\,P + T\,N + F\,P + F\,N)] * 100\%$$

- The false alarm rate is defined as follows:

$$\text{False alarm rate} = [F\,P / (T\,N + F\,P)] * 100\%$$

Attributes	SVM	
	Precision (%)	*The rate of false alarms*
41 attributs	92,82	8,02
27 attributs	92,80	7,89

Table 1: SVM results with the most important attributes

Different within BD	Precision	*The rate of false alarms*
10%	39.72	42.96
20%	81.69	18.43
30%	81.96	18.20
40%	82.10	18.09
50%	92.40	8.37
60%	92.93	7.74
70%	92.87	7.78
80%	92.87	7.68
90%	92.69	7.60

Table 2: Precision and false alarm rate for the SVM model with different samples of the database.

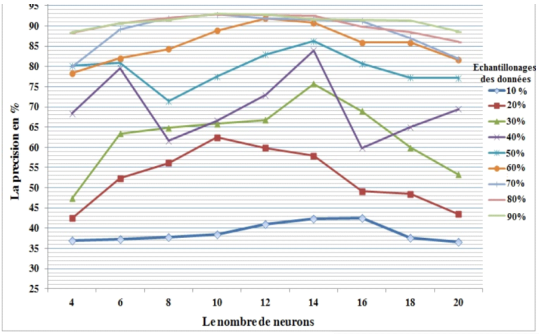

Figure 4: Curve representing the results of the MLP model for the different samples compared to the neurons

5. Conclusion

When measuring the performance of an IDS, the false positive rate and the false negative rate are used to summarize the different detection accuracy characteristics. False positives can be defined as alarms that are triggered from legitimate activities. False negatives are attacks that are not detected by the system. An IDS is more accurate if it detects more attacks and gives fewer false alarms.

An important aspect of this database is that it suffers from the problem of imbalances between classes on the one hand, and a large dispersion of the variables on the other hand. In [22], a new method to treat the imbalance data problem has been proposed.

A new method based on SS approach is used to create a balance between classes. The main advantage of our method is its ability to focus on learning the most important entities, which positively contribute to performance improvement. In the performed experiments, we will show how applying SS on the majority class can give us a better performance, avoiding the selection of non-critical samples. Unlike random sub-sampling, this method allows us to keep all important features in the data set. Furthermore, when using dimensionality reduction techniques, we achieve a significant reduction in computation time while keeping the same performance. This is achieved by selecting the most important features. We will also show that the choice of a good classifier is not enough to obtain good performances. If that is the case, it is necessary to prepare the data by going through all the preprocessing, cleaning and coding steps. These preliminary steps to the presentation of the data of the learning system contribute favorably to the improvement of the performances.

References

1. Pallavi Sethi, Smruti R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications", Journal of Electrical and Computer Engineering, vol. 2017, Article ID 9324035, 25 pages, 2017. https://doi.org/10.1155/2017/9324035

2. Theo Lynn, Patricia Takako Endo, Andrea Maria N. C. Ribeiro, Gibson B. N. Barbosa, Pierangelo Rosati, The Internet of Things : Definitions, Key Concepts, and Reference Architectures, The Cloud-to-Thing Continuum, 2020, ISBN : 978-3-030-41109-1,

3. Imad Saleh, Internet of Things (IoT): Concepts, Issues, Challenges and Perspectives, Université Paris 8, Laboratoire Paragraphe, Université Paris 8, Publié le 26 février 2018, revue Internet des

4. Elrawy, M., Awad, A. & Hamed, H. Intrusion detection systems for IoT-based smart environments: a survey. *J Cloud Comp* **7,** 21 (2018). https://doi.org/10.1186/s13677-018-0123-6

5. Chao Liang ,Bharanidharan Shanmugam,Sami Azam ,Asif Karim ,Ashraful Islam ,Mazdak Zamani ,Sanaz Kavianpour andNorbik Bashah Idris, Intrusion Detection System for the Internet of Things Based on Blockchain and Multi-Agent Systems, Electronics 2020, 9(7),1120; https://doi.org/10.3390/electronics9071120.

6 .F. Meunier. Detection d'intrusions : notions avanceés de nids axes sur le logiciel mun hunt (resource technologies). watch4net, août 2002.

7. W. Stallings. Network security essentials : applications and standards. Prentice Hall, 4 edition, 2011.

8. J. Briaut. Formalisation et garantie de propriétés de sécurité système : application à la détection d'intrusions. 13 décembre 2007

9. S. Axelsson. Intrusion detection systems : A taxonomy and survey. Technical Report, Dept. of Computer Engineering, Chalmers University of technology, Sweden, pages 99 -15, 2000.

10.Z. Baniasadi, A. Sanei, and M. R. Omidi. A fuzzy description logic model for intrusion detection systems. 5th International Symposium on Telecommunications (IST'2010), pages 552 556, 4-6 decembre 2010.

11. N. Wattanapongsakorn, S. Srakaew, and C. Charnsripinyo. A practical network-based intrusion detection and prevention system. 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pages 209 214, 25-27 juin 2012

12. R. Heady, G. Luger, A. Maccabe, and M. Servilla. The architecture of a network level intrusion detection system. Technical Report CS90- 20, University of New Mexico,Department of Computer Science, août 1990

13 . S. Axelsson. Intrusion detection systems : A taxonomy and survey. Technical Report, Dept. of Computer Engineering, Chalmers University of technology, Sweden, pages 9915, 2000.

14. V. Pathak and V. S. Ananthanarayana. A novel multi-threaded kmeans clustering approach for intrusion detection. 3rd International Conference on Software Engineering and Service Science (ICSESS), pages 57 760, 22-24 juin 2012.

15. Alfred Basta, Donald W. House, et al. “Intrusion Détection Systems : Concepts, Methods, and Technologies". 3e édition, 2021

16 .J. P. Anderson. Computer security threat monitoring and surveillance. Technical report, James P. Anderson Company, Fort Washington, Pennsylvania, avril 1980.

17.D. E. Denning. An intrusion detection models. IEEE, transaction on software engineering, 13(2) :222232, 1987.

18. E. W. T. Ferreira, G. A. Carrijo, R. Oliveira, and N. V. S. Araujo. Intrusion detection system with wavelet and neural artical network approach for networks computers. IEEE Latin America Transactions), 9(5) :832 837, septembre 2.

19.C. Han, Y. Lv, D. Yang, and Y. Hao. An intrusion detection system based on neural network. International Conference on Mechatronic Science, Electric Engineering and Computer (MEC), pages 2018 2021, 19-22 août 2011.

20 . A. Patel, M. H. Soni, S. R. Patel. "Deep Learning-Based Intrusion Detection System for IoT Networks". IEEE Internet of Things Journal, 2021

21. Kdd cup 1999 ensemble de données. http ://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, consulté le 21 juillet 2012.

22. I. Chaïri, S. Alaoui, and A. Lyhyaoui. Learning from imbalanced data using methods of sample selection. In proceding of International Conference on Multimedia Computing and Systems, Tanger -Maroc, 10-12 mai 2012.