

Machine Learning-based Intrusion Detection System Against Routing Attacks in the Internet of Things

Abdelhammid Bouazza¹, Hichem Debbi² and Hicham Lakhlef³

¹ Department of Computer Science, University of M'sila, M'sila, Algeria

² Department of Computer Science, University of M'sila, M'sila, Algeria

³ Sorbonne Universités, Université de Technologie de Compiègne CNRS, HEUDIASYC UMR 7253CS 60319; 60203 Compiègne Cedex, France

Abstract

Internet of things (IoT) applications are growing daily, as they are being used in many areas and systems, and as their uses and modes of employment increase, there are many gaps with them. Security is one of the most challenging problems in IoT. IoT is composed of a considerable number of connected devices. Therefore, mobile data traffic is significant, and routing protocols are needed. IoT has many routing protocols; the most widely used is the RPL protocol, which considers limited power and the device's capabilities. Still, it suffers from several weaknesses. The most important one is routing-based attacks which target this protocol. In this work, we aim to solve the problem of Internet of Things exposure to RPL-based attacks as a routing protocol. We built an anomaly intrusion detection system based on Machine learning and an IoT attacks dataset. This dataset, which is generated through the Cooja simulator, contains the most critical attacks and implementation of different scenarios that allowed for the extraction of essential features, in addition to new sensitive features such as nodes' power and their geographical location. Furthermore, we fix minority classes (rare attacks) by balancing the dataset. The results were satisfying because they decreased the false alert rate percentage and maximised accuracy and precision.

Keywords

Internet of things, Security, Intrusion detection system, Machine learning, RPL attacks, Cooja.

1 Introduction

1.1 Background


The Internet of Things (IoT) is a network of commonplace physical things that may connect to the Internet to communicate and collect data utilizing the abilities of the network. These things (nodes) are the digital sensors or networked equipment that can exchange this data via the worldwide Internet. New applications and services are produced due to sensors, connectivity, people, and process interactions. The "Things" in the Internet of "Things" are these electronic gadgets or sensors. Connecting to the Internet via protocols of rootage helps improve quality of life. Each node can reach other nodes and exchange routing information using RPL (Routing Protocol for Low Power and Lossy Networks). However, due to its ad-hoc and limited resource structure [1], IoT systems are very sensitive to intrusions. Attacks usually target a node connected to a large data stream's usability and energy consumption. Attack detection systems are one of the security measures and are crucial in an IoT ecosystem. RPL is a novel distance vector routing protocol

Tunisian Algerian Conference on Applied Computing (TACC 2022), December 13 - 14, Constantine, Algeria

✉ abdelhamid.bouazza@univ-msila.dz; hichem.debbi@univ-msila.dz;

hicham.lakhlef@utc.fr

© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

standardized for constrained 6LoWPAN networks enabling nodes to communicate in a mesh topology. Moreover, several attacks exist on the RPL protocol that target a node's availability, and increase dramatically its power consumption.

1.2 Research problem

The biggest obstacle to routing in the internet of things is security. IoT networks struggle because they lack proven and defined design principles like the client-server paradigm. This shortcoming makes it impossible to use a wide range of conventional security solutions in IoT networks. As a result, IoT is becoming a profitable platform for various Internet assaults as the number of IoT devices rises. These attacks may take many shapes and target various

resources on various IoT devices. For a secure IoT Environment, ongoing monitoring and analysis are required.

ML is an effective method that can be applied to cyber security.

1.3 Research objectives

This research aims to develop an ML-based IDS for detecting routing assaults in IoT. This study concentrated on certain IoT routing attacks. The Cooja simulator is used to mimic each of these threats using actual circumstances. In addition to accuracy and precision, we strive to reduce the false alarm rate as much as possible.

2. Related studies and background

2.1 Intrusion detection systems

IDS scan a computer or a network for irregularities that could be a signal of an intrusion. When they identify unusual activity, intrusion detection systems often notify an administrator [2]. The two fundamental kinds of intrusion detection systems are host-based IDS (HIDS) and network-based IDS (NIDS), with the key distinctions being the IDS's location and intended use. HIDS inspect data stored on specific hosts' computers, while NIDS can monitor the network and look for suspicious activity.

It can be a misuse or anomaly detection.

1. Misuse detection: In order to detect common attacks, misuse-based intrusion detection uses a database of known signatures and patterns [3].
2. Anomaly detection: Using data from regular users, an anomaly-based intrusion detection approach constructs a normal data pattern, and then compares it with current patterns online to find abnormalities [4]. In IoT-based setups, IDS algorithms based on anomalies may be utilized depending on complexity, execution time, and detection time requirements.

2.2 RPL protocol mechanism

By sending a DIO (Dodag Information Object) message to its neighbours, the root node begins the construction of a DODAG (Destination Oriented Directed Acyclic Graph), which contains node rank information to allow it to take its position in the DODAG and prevent steering loops. As a result, each node that receives a DIO message must determine whether or not it wishes to join the DODAG based on its intended use. Upon joining the DODAG, a node will have a path up to the root. After calculating its rank, it updates its neighbour table, and chooses the better father who will be used to redirect messages. DIO messages must be processed by every node in the network until all nodes are accessed. DODAG can be

joined at any time by new nodes through RPL. By using the DIS (Dodag information solicitation) message, the new node requests the DIO message from a node within the DODAG. The new node identifies its best father by receiving the DIO message following the OF (Objective function). The nodes send DIO messages periodically to keep the network stable when the node is already connected to the DODAG and then receives a new DIO message, which will be processed in three different ways:

1. Drop the DIO message according to some rules defined by RPL.
2. Process the DIO message to keep her position in the DODAG
3. Update her position by choosing new parent according to the OF, in this case the node must update parent list to avoid DODAG routing loops.

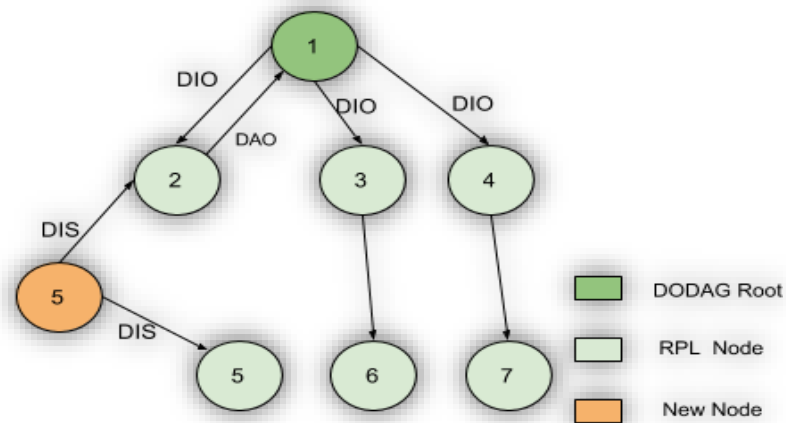


Figure 1: RPL network example (DODAG) [5].

2.3 RPL attacks

IoT applications exist in a variety of domains, including healthcare systems, smart homes, smart cities, smart energy monitoring etc. Due to this variety of applications, routing attacks pose a serious threat to IoT security [6]. RPL is a distance-based protocol. Each network node determines its routing path prior to the initialization of the RPL network. RPL is a tree-based IPv6 routing system for 6LoWPAN that produces Destination Oriented Directed Acyclic Graphs (DODAGs), commonly known as a DODAG tree. The DODAG ID for identification is assigned to each network's root node. Rank numbers and routing tables are also assigned to nodes based on their rank numbers. Nodes are ranked according to their distance from the root [7].

Depending on the type of vulnerability they seek to exploit, RPL attacks can be divided into topology, resource categories, and traffic categories. Energy and power are depleted, and memory is overwhelmed by resource-based assaults. Attacks based on topology disrupt network operations. Consequently, one or more nodes might be disconnected from the network.

In addition, these attacks pose a threat to the network's original topology. Lastly, traffic-based attackers attempt to join the network as normal nodes [5]. Attackers then use network traffic information to conduct attacks.

1. HELLO Flooding attack: A flooding attack is one type of DoS attack, where the malicious nodes send false packets in the network to wear the resources and interrupt the network's working condition. Based on the packet utilized for flooding the network [8].

2. Decreased rank attack: Other nodes are publicized lower than their original rank by malicious nodes. Due to this, several nodes choose illegitimate nodes as their preferred parents. According to WSN attacks [9], this is a sinkhole attack.
3. Version number attack: The attacks aim to increase the version number field inside the DIO messages and transmit them to its neighbours. As a result, a new DODAG construct is forced to cause data packet loss, network congestion, and node resource exhaustion due to control message overhead [10].
4. Blackhole attack : A Blackhole attack in a network would mean that one or more malicious nodes would drop all or part of the data packets being routed through it, causing disruptions in the normal flow of data through the network. A malicious node will distort routing information, present itself as the best path to the control node (called a node sink), and force data through itself [11].

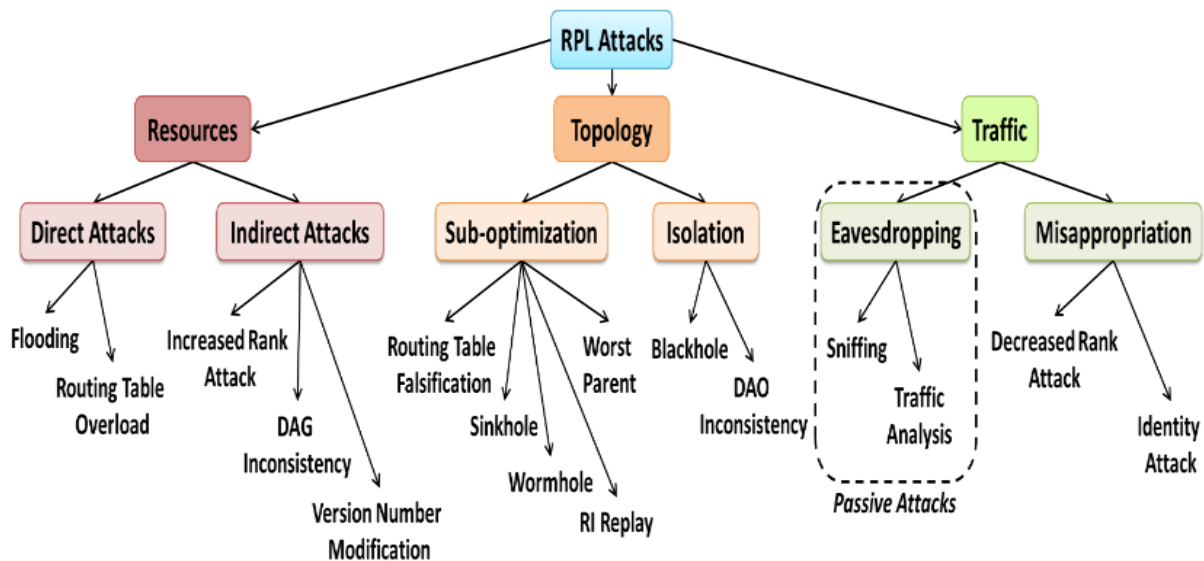


Figure 2: Taxonomy of attacks against RPL networks [5]

3. Related works

This section provides an overview of studies on detecting routing attacks in the Internet of Things using intrusion detection systems.

In [12], the authors have given proof of concept for using deep learning in IoT. First, a method for detecting routing attacks for IoT was given based on deep learning. Nonetheless, the datasets were not enough, and the existing data was very poor in terms of quality; which is considered to be the major problem in IoT.

The authors have also proposed a clearly scalable attack detection methodology based on in- deep-learning for the detection of IoT routing attacks that are a restricted category, hello-flood type, and version number modification attacks with great precision and accuracy. Furthermore, they have built a deep neural network of models formed using IRAD datasets.

In [13], the authors used Contiki-Cooja to simulate RPL attacks and four different attacks. The researchers chose four attacks to implement the experiment: a "hello flood" attack, a "DODAG Information

Solicitation" attack, an "increased version" attack, and a "reduced rank" attack. Later on, a new machine learning model was proposed based on characteristics extracted from network traffic packets, and while using the selected features. Three different classifiers were determined to be more efficient in detecting various attacks, including Naive Bayes, Random Forests, and C4.5. Lastly, their experimental results showed that they could achieve 99.33% classification accuracy using the Random-forest classifier.

In [14], the authors proposed an "IDS " intrusion detection system for smart hospitals. When doing so, they have offered an RPL attack detection system based on anomalies against an IoT network and especially the RPL using support vector machines. The authors considered Hospitals to be an interesting case study, in which many challenges can be faced, such as resiliency of services, interoperability of assets and protection of sensitive information. Throughout the case study, a set of simulation scenario took place. In the first scenario the IoT network didn't include any malicious mote, in the second scenario the IoT network had 1 malicious mote randomly placed, in the third scenario, the IoT network had two malicious motes randomly placed, and lastly on the fourth scenario, the IoT network had 4 malicious motes randomly placed.

The Selected IDS is centralized and uses an SVM machine learning algorithm to identify abnormalities. In order to assess the precision of the proposed IDS, the researchers employed energy consumption as a metric and gathered data for monitoring power per motes in terms of radio energy, receive radio energy, radio transmission energy, and interfered INT radio energy. The observed findings indicate that, as the number of malicious nodes rises, the technique will become more efficient and precise in terms of detection accuracy.

In [15], the authors presented distributed IoT threat detection based on deep learning. They have later on evaluated the performance of classical machine learning and deep learning for detecting distributed attacks. This work performs distributed attack detection via fog computing [16]. In addition, they employed the NSL-KDD [17] dataset to identify assaults. Although this research presents a potential solution for distributed deep learning, it does not particularly address IoT threats.

In [18], the authors have used unsupervised pre-training using SAE (sparse auto encoding) and DNN classifier. An accuracy of 99.65% was reached, and the final model used was AN ID against Clone ID attack.

Comparison with Related Work. To our knowledge, we are the only ones that Added new features (Rank, geographical position), and we studied the attacks' principal to build a global data set valid for any IoT RPL routing attack that adopts data balancing.

4. Proposed model and dataset collection

There are limited datasets available and the quality of available data is poor. Using real scenarios and sensors, we produced our dataset through simulation, and we implemented the Cooja simulator. Here is a summary of how the dataset was built:

4.1 Traffic capture

We captured all the traffic that went through the IoT network with different scenarios as a PCAP file by Wireshark with the help of a ready tool in the Cooja simulator named radio messages. PCAP file is converted to a CSV file. All the simulation is divided into a window time of 1000ms, which means in each second, we have captured some packets. The algorithm used is described in Figure 3.

Raw data sets include data types, such as IP addresses, that the learning algorithm cannot comprehend, causing the model to overfit. Source and destination IPv6 addresses are transformed to node ID to circumvent this issue. For example: IPV6 address 2001:0db8:3cd4:0015:0000:d234::3eee:0011 can be shortened to 11 and the broadcast IP address ff02::1a is converted to 99.

4.2 Generate new features

All the previous steps generated a total of 13 features from 6 features at the beginning.

The transmission and reception time of each packet is calculated. The full length of each packet's delivery and reception is 1000 milliseconds. We then determined the average emission and receiving time for each node, and the number of control packets transmitted from each node (concern the control packets: DAO, DIO and DIS) is calculated in windowing size, 1000 ms. Those values impacted attack detection like Hello Flooding because, in this attack, the transmission rate should be higher. The algorithm used is shown in figure 3 below:

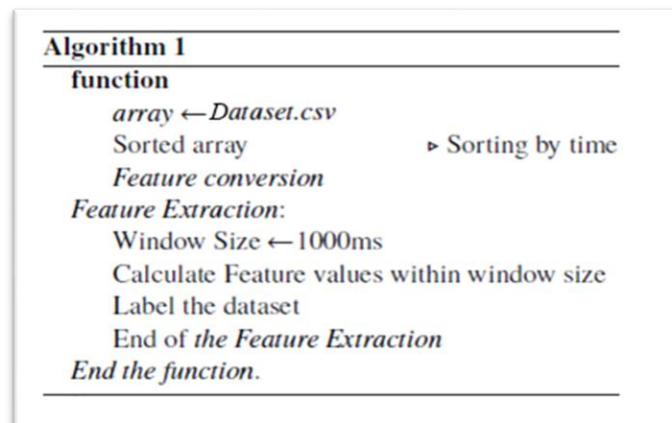


Figure 3: Features extraction algorithm

4.3 Energy Tracking

We tracked the power of the nodes without attacks, and we found that the attacks consumed the energy of the nodes greatly. Using the simulator, four properties were derived: energy (ON), emission mode (radio TX), reception mode (radio RX) and finally INT (interfered radio).

4.4 Position and rank tracking

By changing of position (X, Y) and rang (rank) of the nodes, we discovered that malicious nodes always take an important geographical position and are close to the root node to cover and influence as many nodes as possible.

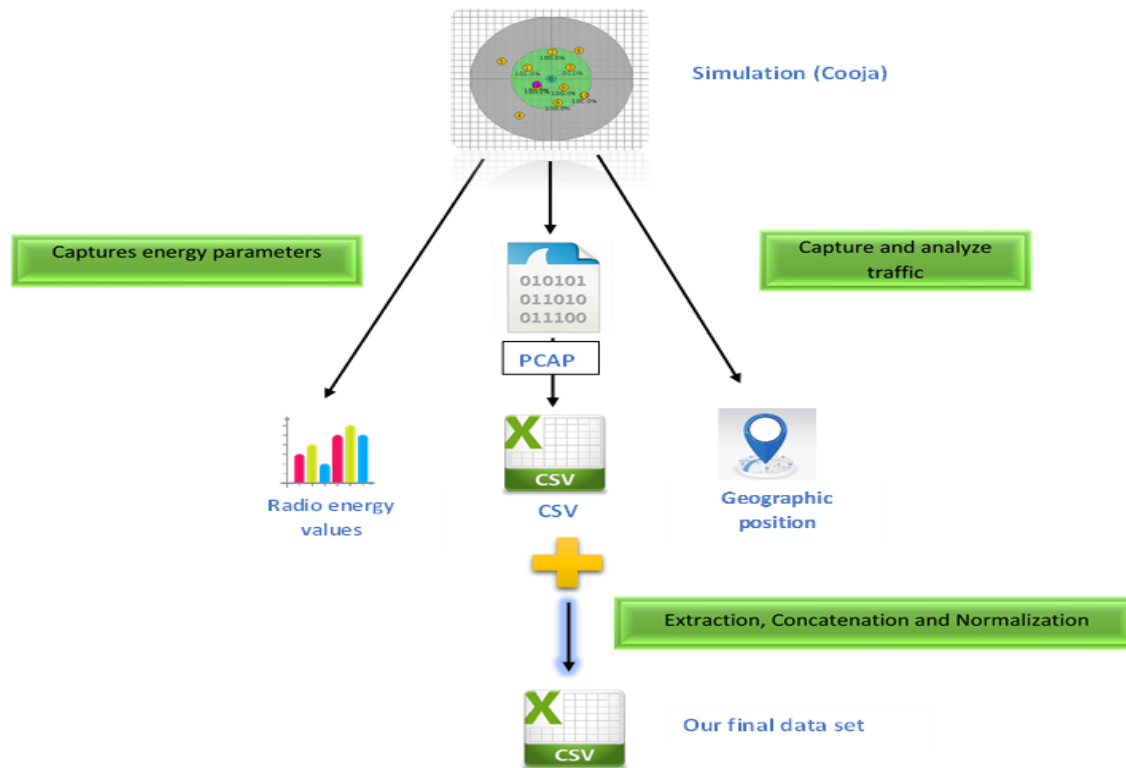


Figure 4: Different steps to build the dataset.

4.5 Dataset description

This RPL attacks dataset contains 24 features and 48024 samples. In tables 1 and 2 below, we will describe all details:

Table 1

Features description

N°	Feature name	Description	N°	Feature name	Description
1	T	Time	13	Length_rec	received Packet size
2	Src	Source	14	DIS_rec	Received DIS number
3	Dst	Dst Destination	15	DIO_rec	Received DIO number
4	Protocol	The upper layer protocol decoded	16	DAO_rec	Received DAO number
5	Dure_tr	Transmission time during a time window	17	ON_Energy	Energy
6	Moy_tr	Transmission media	18	TX	Emission energy
7	Length_tr	Transmitted Packet size	19	RX	Reception energy
8	DIS_tr	Transmitted DIS number	20	INT	Interfered radio
9	DIO_tr	Transmitted DIO number	21	Pos_x	X geographical Position in x axis
10	DAO_tr	Transmitted DAO number	22	Pos_y	Y geographical Position in y axis
11	Dure_rec	Reception time during a time window (1s)	23	Rang	Node rank in DODAG
12	Moy_rec	Reception media	24	Class	Attack Type

Table 2
Dataset information

Normal/Attack	Category	Records Number
Attack	Decreased rank	9 367
	Version Number	3 196
	Black Hole	1 493
	Hello Flooding	5 046
Normal		28922

4.6 Proposed Model

In this paper, we present a technique for discovering routing-based attacks in IoT networks based on the behaviour-based detection of intrusions provided by machine learning.

We determined, using the Contiki Cooja simulator, several network scenarios. Then, we built our dataset using important parameters to detect routing attacks in IoT networks, which is necessary to create our IDS.

Data imbalance refers to a disproportionate distribution of classes within a dataset. If a model is trained under an imbalanced dataset [19], it will become biased and rare attacks are a bad problem. By balancing the dataset, the effectiveness of the model will be improved.

4.7 Dataset balancing

There are 28922 normal samples and 19102 attack samples in the data set. As demonstrated in Table 1, more than sixty percent of the samples fall within normal categories. In this manner, the learning model will predict the majority classes but not the minority classes, indicating that it is biased. Various resampling methods [21] have been proposed to address this issue, including random oversampling, which randomly replicates exact samples of minority classes using techniques such as the synthetic minority oversampling technique (SMOTE), the synthetic minority oversampling technique for nominal and continuous data (SMOTE-NC), and the adaptive synthetic minority oversampling technique (ADASYN). In this study, we used the ADASYN method since it is capable of managing mixed datasets of categorical and continuous features and allows us to avoid the benefits of random oversampling and SMOTE sampling.

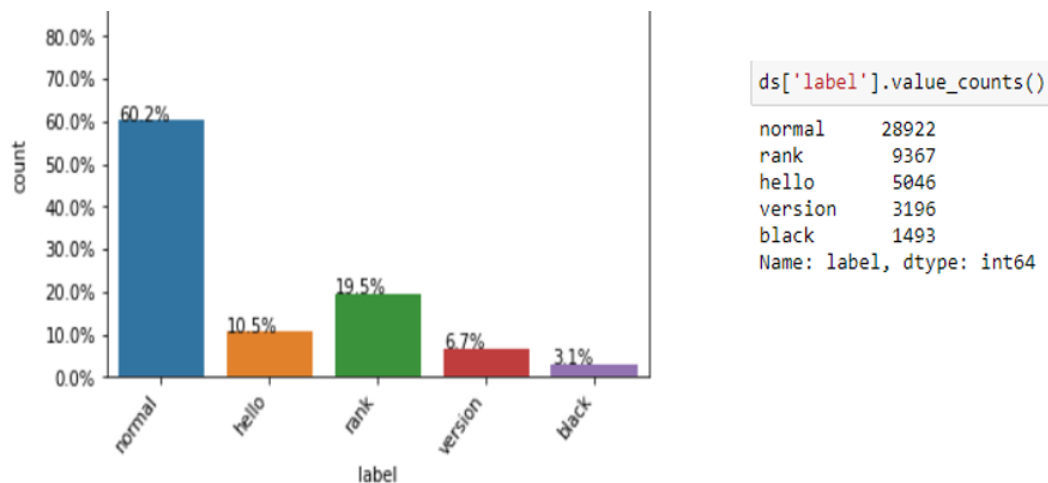


Figure 5: Dataset before balancing

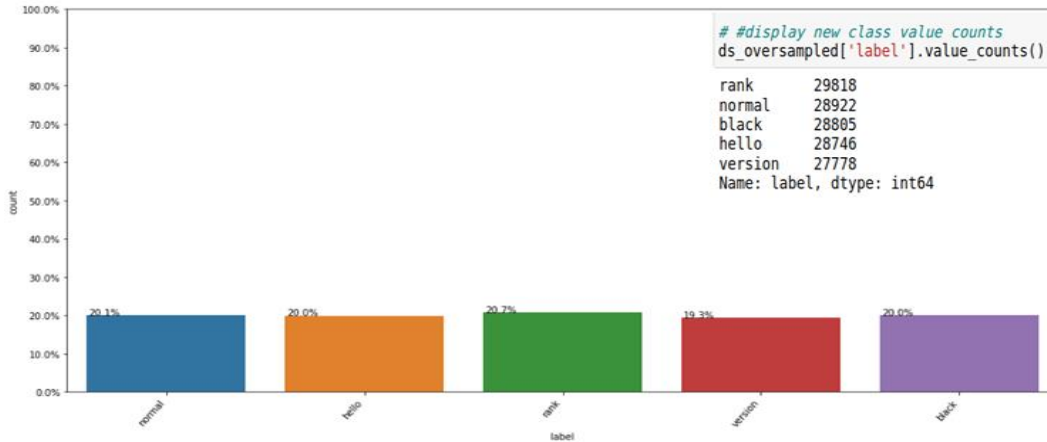


Figure 6: Dataset after balancing

4.8 Features engineering and classification

In this step, feature transformation is applied to the training set. Continuous numerical characteristics are subjected to a min-max scaler. In addition, categorical characteristics are encoded via label encoding, which substitutes each category column with a specified number. This modification is done to the validation and testing subgroups afterwards.

Then This dataset is divided into training, validation, and testing. That contains 80% of the data for training the model, and the rest is only used to validate and test the model's performance.

Finally, we test different machine learning algorithms to classify the bidirectional flows according to the IoT environment.

We selected algorithms from among the known machine learning algorithms with initial hyperparameters to verify that a good model depends on the well of the dataset, even without being based on deep learning. They belong to four classification algorithms tested: Random Forest, Decision Tree, SVM, and Naive Bayes. The performance of the different algorithms is measured on the test set. The metrics used are accuracy, precision, and False alert rate.

5. Results and discussion:

In this section, we have evaluated the performance of the IDS classifiers.

We have focused on Three metrics Accuracy, Precision and False alert rate.

- True positive (TP): an attack data identified as an attack.
- True negative (TN): a normal data identified as normal.
- False positive (FP): a normal data identified as an attack.
- False negative (FN): an attack data identified as normal.
- Accuracy = $(TP+TN) / (TP+TN+FP+FN)$.
- Precision = $(TP) / (TP+FP)$
- False alert rate = $(FP) / (FP+TN)$

We used Intel core i7-4200M CPU @2.5Ghz*4 processor with 8 GB RAM and 500 GB Hard drive to implement the detection learning algorithms.

As for software, we used Weka 3.8.6 (Machine Learning Software in Java).

We used an 80/20 training/test split on this dataset, as illustrated in table 3.

Table 3
train, and test set

	Training	Test
Black	22 946	5 859
Rank	23 922	5 896
Version	22 199	5 579
Hello	23 027	5 719
Normal	23 161	5 761
Total	115 255	2 8814

As illustrated in Table 4, Random Forest and Decision Tree showed its high performance for the highest accuracy and a high precision rate with a low False alert rate than SVM and Naive Bayes.

Table 4
Overall performance on the test set of the different classifiers

Classifier	Accuracy	precision	False alert rate
Random Forest	0.999	0.999	0.001
Decision Tree	0.999	0.999	0.001
Naive Bayes	0.984	0.962	0.010
SVM	0,958	0,896	0.026

5.1 Comparative study with related works:

To evaluate our model's performance, we compared its performance with related works [12,13,18].

The result of this comparative study is summarized in tables 5 and 6 below:

Table 5
comparison with used dataset in each work

	Attack	Dataset	ML/DL	Features
[12]	3	Pcap file	DL	18
[13]	4	Pcap file	ML	21
[18]	1	Pcap file	DL	19
Our dataset	4	Pcap file, Energy, Position	ML	23

Table 6
comparison with related works performance

	Accuracy	Precision	False alert rate	F1-score
[12]	0.949	0.957	/	0.957
[13]	0.993	0.994	/	/
[18]	0.996	/	/	0.996
Our model	0.999	0.999	0.001	0.997

The performance of our study shows a higher accuracy of 99.99% than other related work and the highest Precision and F1-score, as illustrated in Table 6.

These promising results are mainly due to the well-balanced dataset.

6. Conclusion

In this work, we covered the most important attacks against the routing protocol in the Internet of Things and how it works. The security in IoT is more interested than in any other environment because when we talked about IoT, we talked about sensitive components and data. In this work, we built an intrusion detection system based on Machine learning. To train our model, we used a dataset of routing attacks. This dataset was built with the Cooja simulator, and it is based on recent papers. It contains four main attacks (BlackHole, Decreased Rank, Modification Version Number, Hello Flood). It also contains important features such as node position and energy. An effective and efficient Multi-classifier model was then built based on a Machine learning algorithm as a Random Forest after going through the most important steps of processing the dataset and using carefully selected parameters and hyperparameters to achieve good results. The results reported are mainly related to the accuracy, precision and low false alert rate. The final model has been evaluated and compared with recent works, and we got an excellent result, as shown above, that proved our model to be effective.

7. References

- [1] S. Cakir, S. Toklu, and N. Yalcin, "Rpl attack detection and prevention in the internet of things networks using a gru based deep learning," *IEEE Access*, vol. 8, pp. 183678–183689, 2020, doi: 10.1109/ACCESS.2020.3029191.
- [2] A. Abdelkader, D. Youcef, and A. Hadjali, "On the Use of Belief Functions to Improve High Performance Intrusion Detection System," in *Proceedings - 12th International Conference on Signal Image Technology and Internet-Based Systems, SITIS 2016*, Apr. 2017, pp. 266–270. doi: 10.1109/SITIS.2016.50.
- [3] W. Bul'ajoul, A. James, and M. Pannu, "Improving network intrusion detection system performance through quality of service configuration and parallel technology," *Journal of Computer and System Sciences*, vol. 81, no. 6, 2015, doi: 10.1016/j.jcss.2014.12.012.
- [4] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, 2014, doi: 10.1109/SURV.2013.052213.00046.
- [5] A. Mayzaud, R. Badonnel, and I. Chrisment, "A taxonomy of attacks in RPL-based internet of things," *International Journal of Network Security*, vol. 18, no. 3, 2016.

- [6] L. Wallgren, S. Raza, and T. Voigt, "Routing attacks and countermeasures in the RPL-based internet of things," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013, doi: 10.1155/2013/794326.
- [7] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys and Tutorials*, vol. 17, no. 4, 2015, doi: 10.1109/COMST.2015.2444095.
- [8] T. Aditya Sai Srinivas and S. S. Manivannan, "Prevention of Hello Flood Attack in IoT using combination of Deep Learning with Improved Rider Optimization Algorithm," *Computer Communications*, vol. 163, 2020, doi: 10.1016/j.comcom.2020.03.031.
- [9] A. Raoof, A. Matrawy, and C. H. Lung, "Routing Attacks and Mitigation Methods for RPL-Based Internet of Things," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 2, 2019, doi: 10.1109/COMST.2018.2885894.
- [10] A. Mayzaud, A. Sehgal, R. Badonnel, I. Chrisment, and J. Schönwälder, "A study of RPL DODAG version attacks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, vol. 8508 LNCS. doi: 10.1007/978-3-662-43862-6_12.
- [11] K. Chugh, L. Aboubaker, and J. Loo, "Case study of a black hole attack on LoWPAN-RPL," in *Proc. of the Sixth International Conference on Emerging Security Information, Systems and Technologies (SECURWARE)*, Rome, Italy (August 2012), 2012, vol. 7, pp. 157–162.
- [12] F. Y. Yavuz, D. Ünal, and E. Gül, "Deep learning for detection of routing attacks in the internet of things," *International Journal of Computational Intelligence Systems*, vol. 12, no. 1, 2018, doi: 10.2991/ijcis.2018.25905181.
- [13] M. Sharma, H. Elmiligi, F. Gebali, and A. Verma, "Simulating Attacks for RPL and Generating Multi-class Dataset for Supervised Machine Learning," 2019. doi: 10.1109/IEMCON.2019.8936142.
- [14] A. M. Said, A. Yahyaoui, F. Yaakoubi, and T. Abdellatif, "Machine Learning Based Rank Attack Detection for Smart Hospital Infrastructure," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020, vol. 12157 LNCS. doi: 10.1007/978-3-030-51517-1_3.
- [15] A. A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for Internet of Things," *Future Generation Computer Systems*, vol. 82, 2018, doi: 10.1016/j.future.2017.08.043.
- [16] A. Alrawais, A. Alhothaily, C. Hu, and X. Cheng, "Fog Computing for the Internet of Things: Security and Privacy Issues," *IEEE Internet Computing*, vol. 21, no. 2, 2017, doi: 10.1109/MIC.2017.37.
- [17] M. Tavallaei, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," 2009. doi: 10.1109/CISDA.2009.5356528.
- [18] C. D. Morales-Molina et al., "A dense neural network approach for detecting clone id attacks on the rpl protocol of the iot," *Sensors*, vol. 21, no. 9, May 2021, doi: 10.3390/s21093173.
- [19] A. Derhab, A. Aldweesh, A. Z. Emam, and F. A. Khan, "Intrusion Detection System for Internet of Things Based on Temporal Convolution Neural Network and Efficient Feature Engineering," *Wireless Communications and Mobile Computing*, vol. 2020, 2020, doi: 10.1155/2020/6689134.
- [20] S. Ruder, "An Overview Optimization Gradients," arXiv preprint arXiv:1609.04747, 2017.
- [21] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Handling imbalanced datasets : A review," *Science*, vol. 30, no. 1, 2006.