# ELNIDS: Ensemble Learning based Network Intrusion Detection System for RPL based Internet of Things

Abhishek Verma [1] and Virender Ranga [2]

[1]National Institute of Technology Kurukshetra
[2]Affiliation not available

October 30, 2023



**Abstract**

Internet of Things is realized by a large number of heterogeneous smart devices which sense, collect and share data with each other over the internet in order to control the physical world. Due to open nature, global connectivity and resource constrained nature of smart devices and wireless networks the Internet of Things is susceptible to various routing attacks. In this paper, we purpose an architecture of Ensemble Learning based Network Intrusion Detection System named ELNIDS for detecting routing attacks against IPv6 Routing Protocol for Low-Power and Lossy Networks. We implement four different ensemble based machine learning classifiers including Boosted Trees, Bagged Trees, Subspace Discriminant and RUSBoosted Trees. To evaluate proposed intrusion detection model we have used RPL-NIDDS17 dataset which contains packet traces of Sinkhole, Blackhole, Sybil, Clone ID, Selective Forwarding, Hello Flooding and Local Repair attacks. Simulation results show the effectiveness of the proposed architecture. We observe that ensemble of Boosted Trees achieve the highest Accuracy of 94.5% while Subspace Discriminant method achieves the lowest Accuracy of 77.8% among classifier validation methods. Similarly, an ensemble of RUSBoosted Trees achieves the highest Area under ROC value of 0.98 while lowest Area under ROC value of 0.87 is achieved by an ensemble of Subspace Discriminant among all classifier validation methods. All the implemented classifiers show acceptable performance results.

# ELNIDS: Ensemble Learning based Network Intrusion Detection System for RPL based Internet of Things

Abhishek Verma
*Research Scholar*
*Department of Computer Engineering*
*National Institute of Technology*
Kurukshetra, India
abhishek_6170034@nitkkr.ac.in

Virender Ranga
*Assistant Professor*
*Department of Computer Engineering*
*National Institute of Technology*
Kurukshetra, India
virender.ranga@nitkkr.ac.in

***Abstract*—Internet of Things is realized by a large number of heterogeneous smart devices which sense, collect and share data with each other over the internet in order to control the physical world. Due to open nature, global connectivity and resource constrained nature of smart devices and wireless networks the Internet of Things is susceptible to various routing attacks. In this paper, we purpose an architecture of Ensemble Learning based Network Intrusion Detection System named ELNIDS for detecting routing attacks against IPv6 Routing Protocol for Low-Power and Lossy Networks. We implement four different ensemble based machine learning classifiers including Boosted Trees, Bagged Trees, Subspace Discriminant and RUSBoosted Trees. To evaluate proposed intrusion detection model we have used RPL-NIDDS17 dataset which contains packet traces of Sinkhole, Blackhole, Sybil, Clone ID, Selective Forwarding, Hello Flooding and Local Repair attacks. Simulation results show the effectiveness of the proposed architecture. We observe that ensemble of Boosted Trees achieve the highest Accuracy of 94.5% while Subspace Discriminant method achieves the lowest Accuracy of 77.8% among classifier validation methods. Similarly, an ensemble of RUSBoosted Trees achieves the highest Area under ROC value of 0.98 while lowest Area under ROC value of 0.87 is achieved by an ensemble of Subspace Discriminant among all classifier validation methods. All the implemented classifiers show acceptable performance results.**

***Index Terms*—Ensemble Learning, RPL, 6LoWPAN, Classification, NIDS, ELNIDS, Internet of Things**

## I. Introduction

Advancement in the development of low powered tiny embedded devices has facilitated the growth of new networking paradigm called the Internet of Things (IoT) [1] in which anything can communicate to anyone and anytime. IoT consists of objects also known as “Things" (i.e. human, animal etc.) which carry smart devices with built-in intelligence that provides it with a capability to connect and share information over the internet and control the physical world [2]. Smart devices share information to make decisions and perform actuating tasks. IPv6 enables this communication by providing each smart device with a unique IP address thereby making it globally addressable [3], [4]. IoT enables a lot of applications that make human life better, however with a lot of benefits it also carries a lot of risks associated with users security and privacy [5], [6]. In order to standardize IoT different organizations have proposed several protocol standards in the past decade. Most popular ones include 802.15.4 standard for physical and MAC layer by IEEE, IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [7] for network layer by IETF, and CoAP for application layer by IETF [4], [8]. As most of the IoT applications are based on tiny resource constrained (memory, processing, communication and energy) devices which are expected to operate for a long time thus the need for low power consuming protocols are desired. 6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks) networks full fill these critical needs of IoT by enabling nodes or smart devices to operate on low power while maintaining cost-effective wireless personal area networks (WPAN). In order to provide a efficient routing in 6LoWPAN [9] networks RPL protocol has been standardized [7]. While giving major benefits in routing the RPL protocol also suffers from various security and privacy risks. Due to the open and self-organizing nature of IoT, nodes are vulnerable to insider and outsider attacks. We have seen a huge literature in the field of routing attacks particular to wireless sensor networks (WSN). Such attacks can also be performed on 6LoWPAN networks. In addition to it, some newly tailored attacks for RPL are also present in the literature. Many solutions towards securing RPL protocol have been proposed in the literature [10]. These include Intrusion Detection Systems (IDS) and trust based secure RPL protocols. These solutions provide security against a very small number of attacks which is a major concern when we talk about securing IoT. In this paper, we have focused on the development of a Network Intrusion Detection System (NIDS) named ELNIDS which provides defense against seven types of routing attacks. ELNIDS is based on ensemble learning and uses four types of classifiers namely Boosted Trees [11], Bagged Trees [12], Subspace Discriminant [13] and RUSBoosted Trees [14]. We have proposed the architecture for ELNIDS and performed a performance analysis of individual classifiers using different



DOI: 10.1109/IoT-SIU.2019.8777504

validation methods and evaluation metrics. For training and testing of classifiers, we have used our own dataset named RPL-NIDDS17 [15].

### A. RPL protocol

RPL is based on the distance-vector and source routing protocols and operates on top of standard IEEE 802.15.4 protocol. It supports point-to-point, multipoint-to-point and point-to-multipoint topology. RPL creates a Destination Oriented Directed Acyclic Graph (DODAG) from IoT nodes. A DODAG consists of nodes (i.e. router, host and gateway) which organize themselves into a particular form of topological structure in order to carry out routing in Low Power and Lossy Networks (LLNs). A single IoT network contains multiple parallel RPLInstance running at a single time, and a single RPL Instance may contain multiple DODAGs. RPL Instance is identified by RPLInstanceID while DODAG is identified by DODAG ID which is a unique IPv6 address. The main characteristics of the RPL protocol include auto-configuration, self-healing, loop avoidance and detection, transparency, and support for multiple edge routers or sink. RPL uses four types of control messages (DIO, DIS, DAO, DAO-ACK) for creating and maintaining DODAG. Routes between DODAG nodes are selected and optimized using an Objective function (OF). An OF uses various metrics and constraints in order to select the optimal path and parent among different preferred choices. Nodes are assigned a rank value (16 bit) which represents the node's individual position with respect to DODAG root. The rank concept is used to maintain the parent-child relationship, as well as prevent loops in the network.

### B. RPL-NIDDS17 dataset

The RPL-NIDDS17 [15] is a synthetic dataset created using NetSim [16] tool. NetSim is capable of simulating various networking environments i.e. IoT, MANET, FANET, VANET etc. To create the dataset the IoT network scenario is configured with sensor nodes, gateway, router, and a wired node. For every attack, packet captures were retained in separate CSV files. Finally, all the CSV files were merged to form the complete RPL-NIDDS17 dataset. The dataset consists of 20 features and 2 additional labelling attributes. RPL-NIDDS17 contains traces of attacks including Sinkhole, Blackhole, Sybil, Clone ID, Selective Forwarding, Hello Flooding and Local Repair attacks. Features of the dataset have been classified into three categories namely flow, basic and time. Table I shows the full description of the RPL-NIDDS17 dataset and Table II shows the description of the part of the dataset used in this study.

Table I: Full Dataset description

| Category | Number of Instances |
|---|---|
| Attack | 33,337 |
| Normal | 431,981 |

Table II: Part of the dataset used in this study

| Category | Number of Instances |
|---|---|
| Attack | 33,337 |
| Normal | 141,740 |

## II. Related Work

In [17] an architecture for specification-based IDS is proposed for detecting rank and local repair attacks. The proposed IDS uses distributed placement strategy for placing monitoring modules. No simulation study is done in support of IDS performance analysis. Further, in [18] proposed an extension to their previous work [17]. In this work, a specification based IDS is proposed to detect rank, sink-hole, local repair, neighbour and DIS attacks. The proposed specification-based IDS used hybrid placement strategy. Main limitations of this work include the added communication overhead, prior requirement of network trace and fall in IDS accuracy when it operates for a long time. Raza *et* al. [19] proposed a hybrid anomaly-based IDS named as SVELTE. It uses several modules IDS modules with a firewall that provides security against malicious traffic from the outside network. SVELTE is capable of defending against sink-hole, selective forwarding and spoofed or alteration attacks. SVELTE posses several limitations including synchronization issue, strategic placement of IDS modules, high false positive rate and vulnerability to coordinated attacks. It performs well in terms of packet delivery ratio, control packet overhead and energy consumption and the true positive rate.

In [20] a compression header analyzer based IDS named CHA-IDS is proposed. It uses signature-based detection mechanism which is embedded in the border router. It requires high memory and energy consumption. Moreover, it cannot locate the attacker. A signature-based IDS to detect DIS attack and Version number attack is proposed in [21]. The proposed IDS requires detection and monitoring modules to be placed on nodes itself as in the case of hybrid detection schemes. However, authors consider using two types of additional nodes. The first type of nodes IDS routers which carry detection and firewall modules. The second type is IDS detectors which are responsible for monitoring and sending malicious traffic information to the router nodes. Kfoury *et* al. [22] proposed an IDS for detecting Sinkhole, Version number, and HELLO flooding attacks in particular to RPL protocol. The authors used Self Organizing Map for clustering the attack and normal traffic. In depth details of methodology behind labelling of clusters is not elaborated in this work. In addition, the proposed IDS is not evaluated in terms of the implementation overhead i.e. node resource.

## III. Proposed Work

In this paper, ensemble learning [23] methods are used to develop IDS [24] modules. This is because ensemble learning provides advantages in the case of classification problems. Main advantages include better prediction and model stability. Ensemble methods help in improving classification results

by combining multiple models. Thus, using multiple models helps in gaining better prediction accuracy. The aggregated output of the ensemble is less noisy than any other machine learning methods. In addition to this ensemble, models are avoid overfitting by utilizing bagging methods. RPL-NIDDS17 dataset from Zenodo has been used to train and test classifiers and results have been compared in terms of Accuracy and Area under ROC (receiver operating characteristic) [25] curve. Accuracy refers to the ratio of the total number of correct predictions to the total number of predictions. ROC curve is plotting True Positive Rate (TPR) against False Positive Rate (FPR), the area under ROC refers to the area under the ROC curve.

### *A. Experiment Flow Design*

Fig. 1 shows the experimental flow design followed during this work. In the first step, the RPL-NIDDS17 dataset is preprocessed by applying cleaning, encoding and scaling methods. Cleaning refers to handling missing values, encoding is used to handling nominal data by one-hot-encoding i.e. conversion from nominal to numeric form, and scaling has been used to scale the concerned feature between 0 to 1. The preprocessed dataset is divided into train and test sets. In the second step, ensemble classifiers (Bagged Trees, Boosted trees, Subspace Discriminant and RUSBoosted Trees) are trained with the train set. The main reason behind the selection of these classifiers is that they perform well on different types of datasets i.e. balanced and imbalanced. We conducted experiments with other ensembles including AdaBoost and Random Forest and found better results with selected four ensembles (Bagged Trees, Boosted trees, Subspace Discriminant and RUSBoosted Trees) in case of RPL-NIDDS17. In the third step, trained models are then tested using the test set. In testing, phase models output their predictions for input test instances into attack or normal class. Classifier details are depicted Table III.

Table III: Ensemble classifier details

| Classifier type | Ensemle method | Learner type |
|---|---|---|
| Boosted Trees [11] | AdaBoost | Decision Tree |
| Bagged Trees [12] | Bag | Decision Tree |
| Subspace Discriminant [13] | Subspace | Discriminant |
| RUSBoosted Trees [14] | RUSBoost | Decision Tree |

### *B. Ensemble Learning based Network Intrusion Detection System*

We propose a signature-based NIDS architecture named EL-NIDS for detecting routing attacks like Sinkhole, Blackhole, Sybil, Clone ID, Selective Forwarding, Hello Flooding and Local Repair attacks on RPL. Fig. 2 shows the architecture of ELNIDS. The proposed IDS architecture consists of the sniffer, sensor events/traffic repository, a feature extraction module, the analysis engine, signature database, user interface, alarm/attack notification manager. Sniffer is responsible for listening to all the packet transmissions within the 6LoWPAN

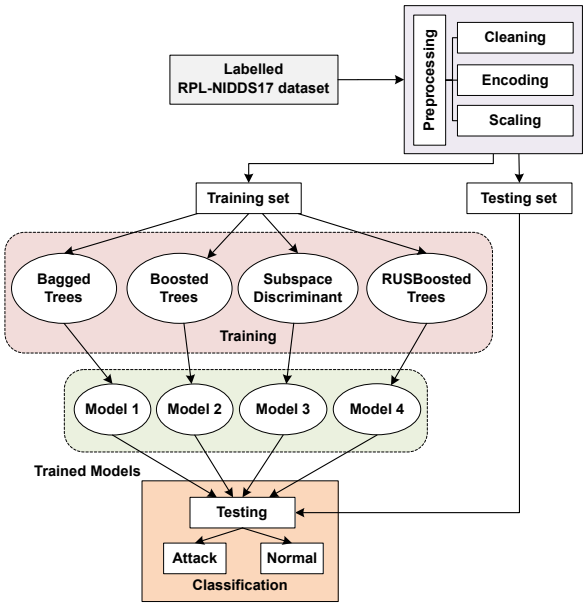


Figure 1: Experiment flow design

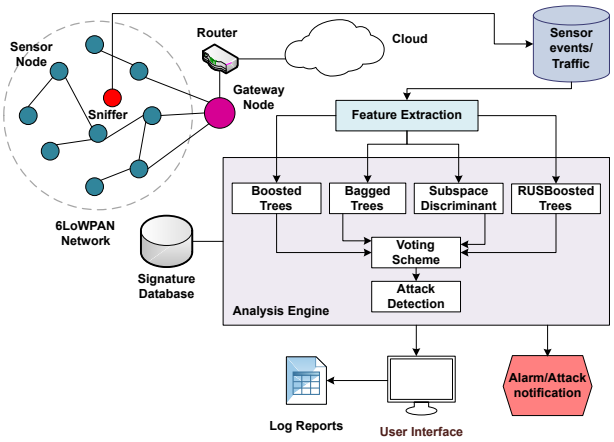


Figure 2: Architecture of ELNIDS

network. Sniffer is assumed is a device working in promiscuous mode and has a large amount of energy storage for long-term operation, it can be connected to the main line of power if 6LoPWAN network is static. Sniffers are directly connected to sensor events/traffic collection repository where all the sniffed sensor events and packet transmissions are stored in the form of packet traces (PCAP format). Feature extraction module is dedicated towards extraction of useful features from the collected packet traces which are further utilized for traffic classification purpose. The heart of ELNIDS is the analysis engine which is responsible for classifying traffic into attack or normal. It consists of trained ensemble models which classify traffic instance and send their predictions to the voting scheme module. The results of models are aggregated by voting scheme module in which the majority vote or class results as the final class of traffic instance. The result of the voting module is sent to the attack detection module which sends

commands to the alarm/attack notification module for raising alarm in case attack is detected. In addition, the analysis engine constantly sends information to the user interface where the traffic is being monitored regularly. User interface logs all the information collected from the analysis engine in the form of log reports. The signature database contains signature information which is used by the analysis engine while performing pattern matching. It is directly connected to the analysis engine. The main reason for using a dedicated voting scheme is to generalize the idea of prediction aggregation which additionally improves overall IDS performance.

## IV. Experimental Setup

The performance assessment has been carried out on a machine operated on 64-bit Windows 10 Pro and equipped with Intel® i7-7700 four core CPU having 3.60 GHz clock speed and 12GB main memory. Matlab 2017b is used for the implementation and evaluation of ensemble classifiers. Dataset preprocessing is performed using Pandas library of Python programming language.

## V. Results and Discussion

We have used all 20 features of the dataset for performance analysis of classifiers. In stage 1, we perform preprocessing of the dataset features. We removed all the instances which consisted of missing values and then converted all the nominal or symbolic features to numeric form using one-hot-encoding. Then all the features are scaled between 0 to 1 i.e. normalization. In stage 2, the classification learner module of Matlab 2017b is used for the evaluation of ensemble classifiers. Every classifier is evaluated with four validation methods which include 30% hold-out, 40% hold-out, 5-fold and 10-fold cross-validation. In stage 3, all the evaluation results are tabulated and compared and practicality of ELNIDS is generalized.

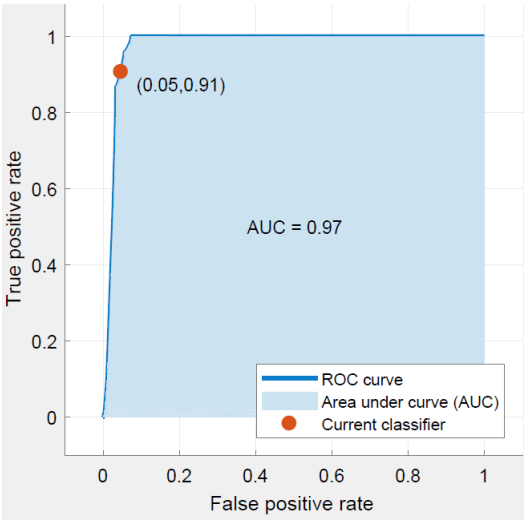


Figure 3: AUC achieved in case of Boosted Trees

From the experimental results in Table IV, the ensemble of Boosted Trees achieves the highest accuracy of 94.5% and

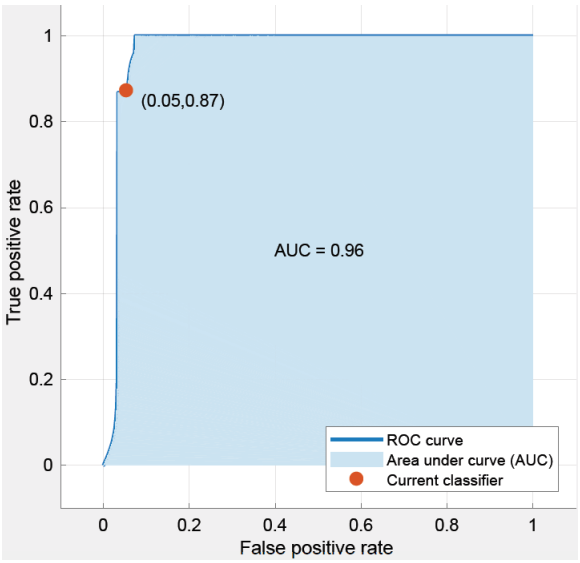


Figure 4: AUC achieved in case of Bagged Trees

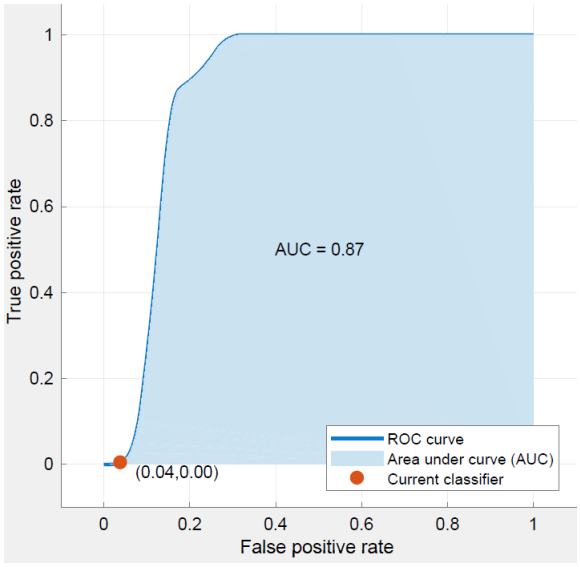


Figure 5: AUC achieved in case of Subspace Discriminant

an AUC value of 0.98 in case of 30% hold-out validation. Lowest accuracy and AUC is achieved by the ensemble of Subspace Discriminants. Similarly, for 40% hold-out validation method, an ensemble of Boosted Trees achieves best results as compared to others. In the case of 5-fold cross and 10-fold cross-validation methods, Highest accuracy is achieved by an ensemble of Boosted Trees while the highest AUC value is achieved by the ensemble of RUSBoosted Trees. Fig. 3-6 show the ROC curve and AUC of Boosted Trees, Bagged Trees, Subspace Discriminant, and RUSBoosted Trees respectively. From experimental results, it can be concluded that the ensemble methods can improve the performance of network-based IDS. Thus, they can help protect RPL based 6LoWPAN networks from various routing attacks.

Table IV: The comparison of accuracy and AUC values achieved with ensemble classifiers.

| Validation method | Ensemble Method | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Boosted Trees | | Bagged Trees | | Subspace Discriminant | | RUSBoosted Trees | |
| | Accuracy | AUC | Accuracy | AUC | Accuracy | AUC | Accuracy | AUC |
| 30% hold-out | 94.5 | 0.98 | 93.4 | 0.97 | 78.7 | 0.87 | 94.1 | 0.98 |
| 40% hold-out | 94.5 | 0.98 | 93.4 | 0.97 | 77.8 | 0.87 | 93.9 | 0.98 |
| 5-fold cross-validation | 94.4 | 0.97 | 93.4 | 0.97 | 78.0 | 0.87 | 94.0 | 0.98 |
| 10-fold cross-validation | 94.4 | 0.97 | 93.3 | 0.96 | 78.6 | 0.87 | 94.0 | 0.98 |

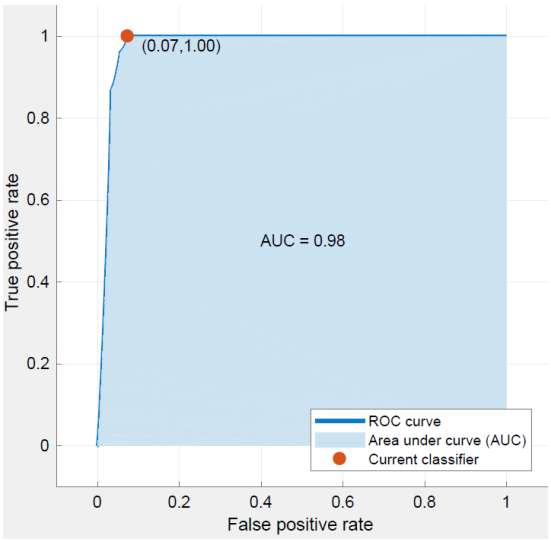


Figure 6: AUC achieved in case of RUSBoosted Trees

## VI. Conclusion and Future Scope

In this paper, we emphasized using ensemble-based machine learning models for creating a network intrusion detection system. We proposed an architecture for a network intrusion detection system which we call ELNIDS. The proposed architecture is capable of detecting Sinkhole, Blackhole, Sybil, Clone ID, Selective Forwarding, Hello Flooding and Local Repair attacks. We implemented for different classifiers including the ensemble of Boosted Trees, Bagged Trees, Subspace Discriminant and RUSBoosted Trees. To evaluate the performance of classifiers we used the RPL-NIDDS17 dataset which contains traces routing attacks on RPL protocol. The simulation results show that ensemble classifiers based on Boosted Trees and RUSBoosted Trees achieve the best performance in terms of accuracy and Area under ROC. Thus, the overall classifier performance evaluation results show the effectiveness of ELNIDS. In future, we target to implement and evaluate ELNIDS on smart nodes. In addition, our aim to develop lightweight defense solutions for securing the Internet of things.

## References


[1] K. Ashton, "That 'internet of things' thing," *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.

[2] L. D. Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov 2014.

[3] J. Granjal, E. Monteiro, and J. S. Silva, "Security for the internet of things: A survey of existing protocols and open research issues," *IEEE Communications Surveys Tutorials*, vol. 17, no. 3, pp. 1294–1312, thirdquarter 2015.

[4] 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 Tutorials*, vol. 17, no. 4, pp. 2347–2376, Fourthquarter 2015.

[5] A. Whitmore, A. Agarwal, and L. Da Xu, "The internet of things—a survey of topics and trends," *Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, Apr 2015.

[6] D. Miorandi, S. Sicari, F. D. Pellegrini, and I. Chlamtac, "Internet of things: Vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497 – 1516, 2012.

[7] T. Winter, "Rpl: Ipv6 routing protocol for low-power and lossy networks," 2012. [Online]. Available: https://tools.ietf.org/html/rfc6550

[8] M. R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. A. Grieco, G. Boggia, and M. Dohler, "Standardized protocol stack for the internet of (important) things," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1389–1406, Third 2013.

[9] J. Olsson, "6lowpan demystified." [Online]. Available: http://www.ti.com/lit/wp/swry013/swry013.pdf

[10] A. Mayzaud, R. Badonnel, I. Chrisment, and I. Grand Est -Nancy, "A Taxonomy of Attacks in RPL-based Internet of Things," *International Journal of Network Security*, vol. 18, no. 3, pp. 459–473, 2016.

[11] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proceedings of the 22Nd International Conference on Machine Learning*, ser. ICML '05. New York, NY, USA: ACM, 2005, pp. 625–632.

[12] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06. New York, NY, USA: ACM, 2006, pp. 161–168.

[13] J. Hamm and D. D. Lee, "Grassmann discriminant analysis: a unifying view on subspace-based learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 376–383.

[14] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "Rusboost: A hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 1, pp. 185–197, 2010.

[15] A. Verma and V. Ranga, "RPL-NIDDS17- A Data set for Intrusion Detection in RPL based 6LoWPAN Networks (Internet of Things)," Aug. 2018. [Online]. Available: https://doi.org/10.5281/zenodo.1406034

[16] "NetSim - Network Simulator, and Emulator," https://www.tetcos.com/netsim-std.html, 2018, [Online; accessed 19-April-2018].

[17] A. Le, J. Loo, Y. Luo, and A. Lasebae, "Specification-based IDS for securing RPL from topology attacks," *IFIP Wireless Days*, vol. 1, no. 1, pp. 4–6, 2011.

[18] A. Le, J. Loo, K. K. Chai, and M. Aiash, "A specification-based IDS for detecting attacks on RPL-based network topology," *Information (Switzerland)*, vol. 7, no. 2, 2016.

[19] S. Raza, "Lightweight security solutions for the internet of things," Ph.D. dissertation, , SICS, 2013.

[20] M. N. Napiah, M. Y. I. Idris, R. Ramli, and I. Ahmedy, "Compression Header Analyzer Intrusion Detection System (CHA -IDS) for 6LoWPAN Communication Protocol," *IEEE Access*, vol. 6, pp. 16 623–16 638, 2018.

[21] P. Ioulianou, V. Vasilakis, I. Moscholios, and M. Logothetis, "A signature-based intrusion detection system for the internet of things," in *Information and Communication Technology Form (ICTF)*, June 2018.

[22] E. Kfoury, J. Saab, P. Younes, and R. Achkar, "A self organizing map intrusion detection system for rpl protocol attacks," *International Jour-*

*nal of Interdisciplinary Telecommunications and Networking (IJITN)*, vol. 11, no. 1, pp. 30–43, 2019.

[23] T. G. Dietterichl, "Ensemble learning," in *The Handbook of Brain Theory and Neural Networks*, M. Arbib, Ed. MIT Press, 2002, pp. 405–408.

[24] N. Moustafa, B. Turnbull, and K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," *IEEE Internet of Things Journal*, pp. 1–1, 2018.

[25] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (roc) curve." *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.