T.C.

ALTINBAS UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

IMPLEMENTING INTRUSION DETECTION SYSTEM IN IOT NETWORKS USING MACHINE LEARNING

Abdulrahman Salim A.Alrahman ALKHASARA

Master of Science

Supervisor

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Istanbul, 2021

IMPLEMENTING INTRUSION DETECTION SYSTEM IN IOT NETWORKS USING MACHINE LEARNING

by

Abdulrahman Salim A.Alrahman ALKHASARA

Electrical and Computer Engineering

Submitted to the Graduate School of Science and Engineering

In partial fulfillment of the requirements for the degree of

Master of Science

ALTINBAŞ UNIVERSITY

2021

The thesis titled "IMPLEMENTING INTRUSION DETECTION SYSTEM IN IOT NETWORKS USING MACHINE LEARNING" prepared and presented by "Abdulrahman Salim A.Alrahman ALKHASARA" was accepted as a Master of Science Thesis in Electrical and Computer Engineering.

Asst. Prof. Dr.Abdullahi Abdu IBRAHIM

Supervisor

Thesis Defense Jury Members:

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM	School of Engineering and Natural Sciences, Altinbas University	_____
Asst. Prof. Dr. Muhammad ILYAS	School of Engineering and Natural Sciences, Altinbas University	_____
Asst. Prof. Dr. Tareq MOHAMMED	Computer Science and Information Technology, Imam Ja'afar AL-Sadiq University, Kirkuk Kirkuk, Iraq	_____

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Approval Date of Institute of Graduate Studies:
____/____/____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Abdulrahman Salim A.Alrahman ALKHASARA

DEDICATION

I would like to thank Allah Almighty for the power of mind, health, strength, guidance, knowledge and skills to complete this study.

This thesis is wholeheartedly dedicated to my parents. There are no words to describe what you mean to me; there is nothing that I can repay for what you have done to me. I will continue to do my best to achieve your expectations.

Lastly, I dedicated this to my wife, brother and sisters, relatives, and friends who have been encouraging me during this study.

ACKNOWLEDGEMENTS

I would like to thank my supervisor professor Asst. prof. Dr. Abdullahi Abdu IBRAHIM for all support during my study. It's great pleasure to express my deepest gratitude to my friends who have shared with me best moments during my study for the Master degree.

ABSTRACT

IMPLEMENTING INTRUSION DETECTION SYSTEM IN IOT NETWORKS USING MACHINE LEARNING

Abdulrahman Salim A.Alrahman ALKHASARA

M.Sc., Electrical and Computer Engineering, Altinbas University

Supervisor: Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Date: July/2021

Pages: 49

Since the last decade, Internet of Things (IoT - Internet of Things) solutions have been created and applied in different branches of society, such as solutions for transport and urban communication, for example. In this way, IoT is a new paradigm, which is composed of a global network of machines and devices capable of interacting with each other [4]. IoT changes the way some real-world problems can be modeled in the cyber environment, and this is due to the ability to list device units on the network to perform small tasks, and then group the results. In this work we have designed and implemented a system whose main purpose is to assist in the process of implementing this approach, a machine learning technique known as Active Learning was used. This technique was chosen in the context of this work, due to its potential to induce predictive models from databases with the lowest number of labels. Thus, a particle swarm algorithm PSO was used on the database, which consists of a supervised learning algorithm to validate the results obtained, a second predictive model was induced, which did not use active learning for sample selection. This model was used as a basis for comparative analysis of the model resulting from the proposal of this work.

Keywords: component, formatting, styling.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

LIST OF ABBREVIATIONS

IDS	:	Intrusion Detection System
PSO	:	particle swarm optimization
OSI	:	Open Systems Interconnection Model
CAN	:	Controller Area Network
ROC	:	Receiver operating characteristic
AI	:	Artificial Intelligence
AUC	:	Area under the ROC Curve
ACC	:	Accuracy
LS-SVM	:	Least square Support Vector Machine
SVM	:	Support Vector Machine

LIST OF SYMBOLS

I_{th}	:	iteration number
S^i	:	Swarm by the current iteration
Auc	:	Accuracy curve
C_i	:	Clusters by the current iteration
X_i	:	best fit element

1. INTRODUCTION

1.1 BACKGROUND

Over the last decade, people have found ways to connect anything, including vehicles and cities, with the Internet to make things and services smarter, particularly in the past decade, and this has happened in various fields including transportation and communication. In this manner, IoT is a global network of interconnected computers and embedded devices able to communicate with each other If the Internet of Things may be used to model the modern world, that is because it provides the power to connect things together in a granular fashion, and then organize their information. In [5], a cloud-based IoT sensor solution is presented, for monitoring the wine production process. This type of approach allows for 24/7 real-time supervision from anywhere in the world. Such an example shows what the essence of IoT is:

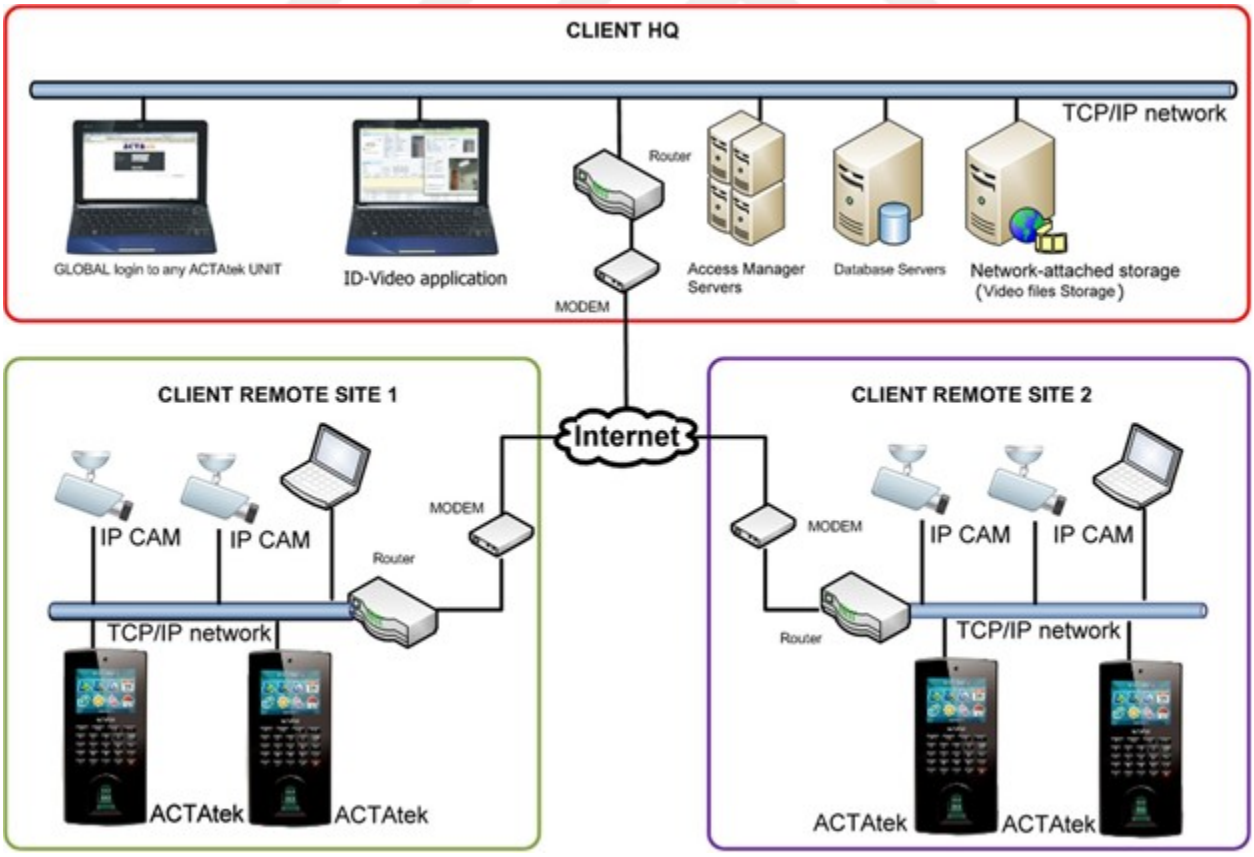

Figure 1.1: Example of IOT security system.

1.2 MOTIVATION

Time Learning was used in the implementation of this method to aid the solution. This methodology was used for its ability to generate low-complexity data models in the sense of this project since it relies on databases to build predictive models.in other words, it is not necessary for a specialist to assign labels to the entire database used in induction, only to a few samples, listed by the algorithm itself, thus optimizing the labeling process. infected by some Botnets. From this traffic, characteristics were removed, which went through a windowing process, which consists of grouping the packets trafficked at each predetermined time period of 1, 5, 10 and 30 seconds. The following were all taken out: how many packets were transmitted, how many protocols were used, and packet size. In other words, particle swarm optimization was employed on the database in a controlled fashion, self-organizing system.

1.3 PROBLEM STATEMENT

While these modern tools are intended to help mankind, they also provide avenues for those who want to access or delete third-party data Similarly, it is possible for machines and computers to provide poor encryption, including passwords that have easy-to-to-guess combinations or to use well-known terms included in a dictionary A technique used by hackers is called a botnet, which is a group of bots that can perform malicious acts. We, then, therefore, conclude that IoT protection is something that needs to be dealt with. Nearly 26 billion Internet of Things (IoT) systems will be linked in 2020, according to the creative forecasts As a result, it is imperative to improve IoT protection approaches. For protecting other networks, a typical strategy is installing intrusion detection systems. More research papers will be published with regard to these works, such as [7] and [9], which use supervised machine learning to identify potential fraud. These methods are complicated because of the tedious and complex nature of data tagging. It is almost as though the world will soon be consumed by hardware.

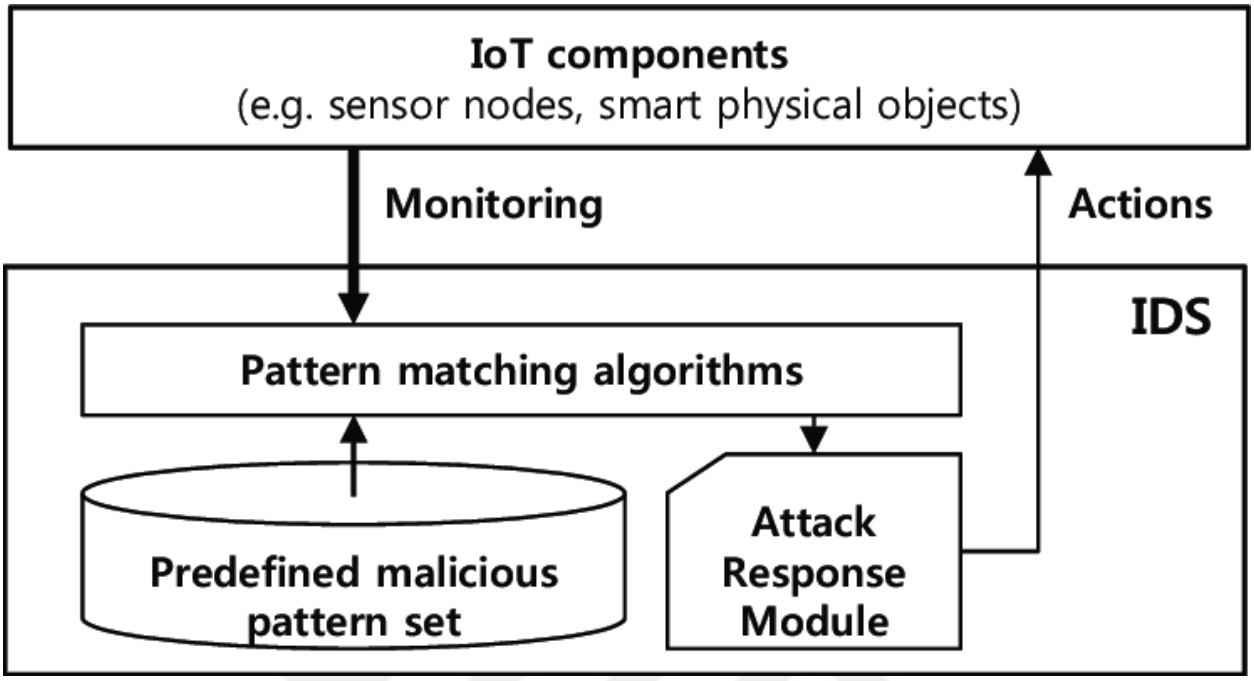

Figure 1.2: (IOT) Intrusion Detection System

1.4 CONTRIBUTION

Predefined Machine Learning has been created and applied to aid in the implementation of our approach's key goal, which is to use the power of machine learning and training to achieve recognized as Active Learning. For this project, since it has the best ability to predict with the fewest amount of labels from the available datasets. In other words, it is not necessary for a specialist to assign labels to the entire database used in induction, only to a few samples, listed by the algorithm itself, thus optimizing the labeling process. infected by some Botnets. From this traffic, characteristics were removed, which went through a windowing process, which consists of grouping the packets trafficked at each predetermined time period of 1, 5, 10 and 30 seconds. Some of the features removed were: number of packets transmitted, number of protocols used and average packet size. In other words, particle swarm optimization was employed on the database in a controlled fashion, self-organizing system. Another paradigm was introduced which bypassed the usage of active learning altogether. This study was based on the comparison of the latter's proposal's model. Therefore, it became evident that when using a sampling strategy, such as Uncertainty Sampling, the model can converge with greater efficiency, even with a reduced number of labels. Therefore, the objective of this work is to

study a Botnet detection system for an IoT network. This system used an approach based on active learning in order to categorize the behavior of the network, which is divided into normal and anomalous. This makes it possible to fire defense triggers when anomalous behavior is detected.

1.5 THESIS ORGANIZATION

The thesis is structured as follows: Section 2 is where we review some of the previous work and implementation on smart grids. Section 3 is where we give a background about all the components. Section 4 is where we explain our model in details and implementation of that model in details, Section 5 is where we simulate our model and record the results obtained. Section 6 is where we conclude our work and put a future scope into perspective.

2. LITERATURE REVIEW

In [19], the author aims to connect the field of study of active learning to intrusion detection in IoT networks. For that, the main concepts related to active learning are presented, as well as about intrusion detection in IoT networks. The implementation carried out used the uncertainty sampling technique in the sample selection process, and the author was unclear as to which classifier he applied. For the evaluation of the algorithms the precision and recall metrics were analyzed, the results were based on a comparison between uncertainty sampling strategy and random sampling. In uncertainty sampling the total number of labeled instances was reduced to almost one third to achieve the same performance, in relation to random sampling.

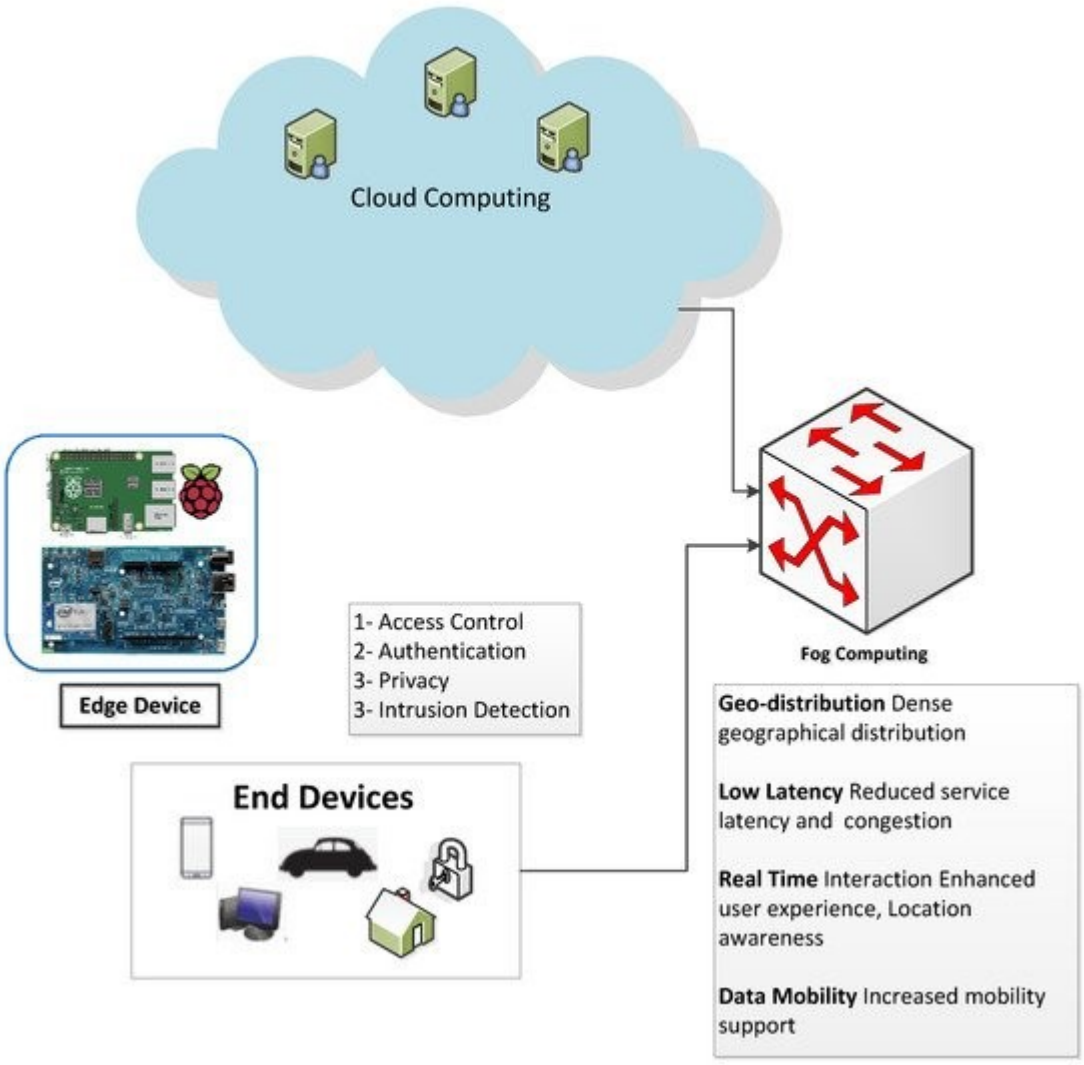

Figure2.1: Structure of (IOT) network using fog computing

Sculley [20] uses a Stream-Based active learning scenario to classify emails, so emails are divided into two classes, the ordinary email class (Ham), and the email class mails that are considered undesirable (Span). Based on the results obtained, the author concludes that Stream-Based active learning is the most appropriate solution to deal with e-mail classification, due to the low computational cost added to the method. Finally, it recommends active learning as a new standard for span filters. In research [21], active learning is explored with the aim of reducing, organizing and selecting biomedical data, such as genes, to then carry out the induction of predictive models. Thus, the following classifiers were used: forest paths (Optimum Path Forest - OPF) and support vector machine (Support Vector Machine - SVM). The results obtained

demonstrate that SVM presents better results in the result set for the data set used, but in relation to the computational time, OPF performed the test steps more quickly.

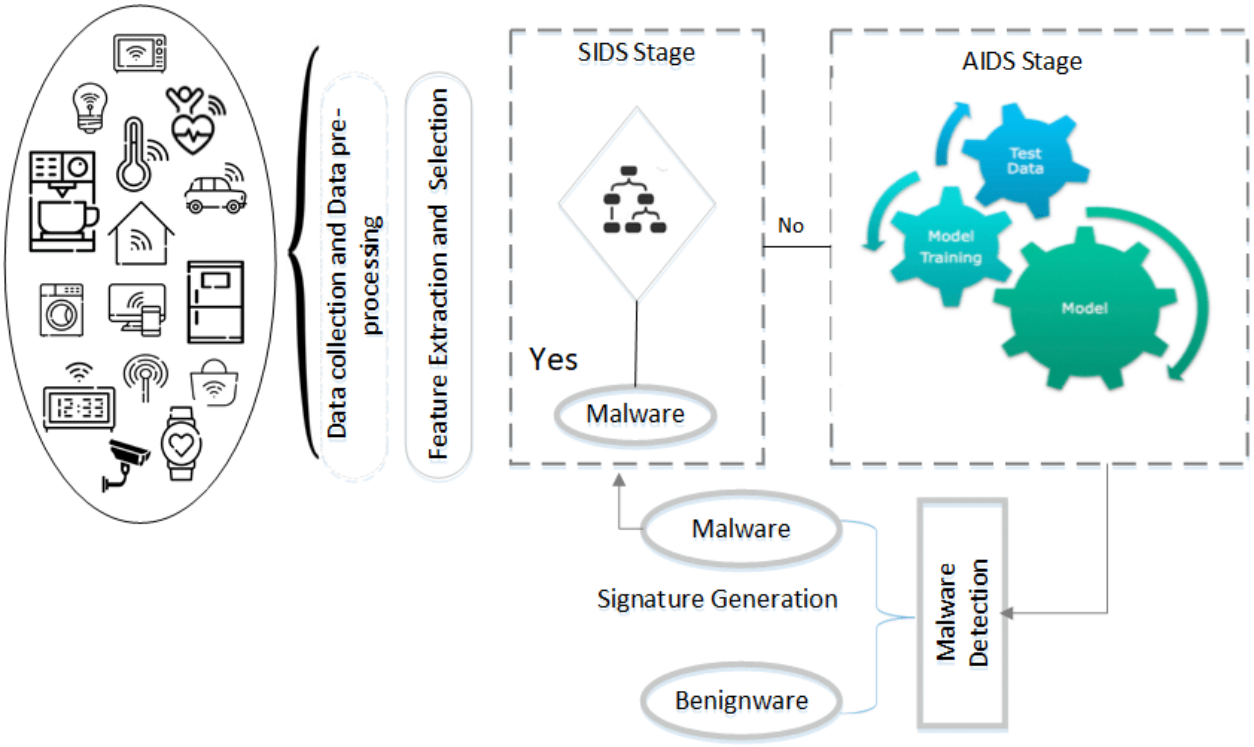

Figure 2.2: Hybrid (IDS) for the (IoT) ecosystem

In [9], the efficiency of machine learning algorithms in detecting botnets is analyzed, for which a detection methodology was created. The methodology consisted of two stages, one in which selected features that helped in the detection of botnets, and the second one that presents the algorithms that had better performance in the detection. The results showed that Naive Bayes had the worst results. The J48 algorithm in more unbalanced scenarios also did not obtain good results, while SVM showed good results, but indicates computational cost made impossible by the high processing time in the training phase. In [22], a tool based on active learning is proposed, which aims to define the behavioral profile of new malware. To this end, classifiers were induced based on information about new malware, in order to enrich the profile base and the model. The active learning technique used by the author achieved positive results, as the model used was able to define the profile of 77% more malware, in the same period of time, in relation to a random sample selection approach.

3. MATERIALS AND METHODS

In this chapter we review some of the essential components that we used to implement our model, we give a brief introduction to the component, and then we show how we simulate that component inside MATLAB environment.

3.1 INTERNET OF THINGS (IOT)

The internet of things (IoT) is a digital system that links artifacts on earth and in the sky with the aim of discovering data and interaction. This technology consists of and incorporates the Internet and contact networks, as well as sensors and connection, and resources. autonomy in its data flow, transferability of network information, interdependence attentive individual Physical- and virtual environment interactions in an IoT system are illustrated in this picture [8] The new version of the Internet, which is called "the Internet of Things," would be comprised of billions of interconnected networks with sensors located in many different ecosystems. This modern computation and networking infrastructure features include sensor networks, special entity recognition, and real-time positioning. IT means much more than only relates to physical artifacts, but includes the interrelationships between the objects in the natural universe as well. A thing is a genuine entity that exists in real-world amounts, but often several various forms with a distinct digital identifier, such as serial numbers, images, words, or metadata.

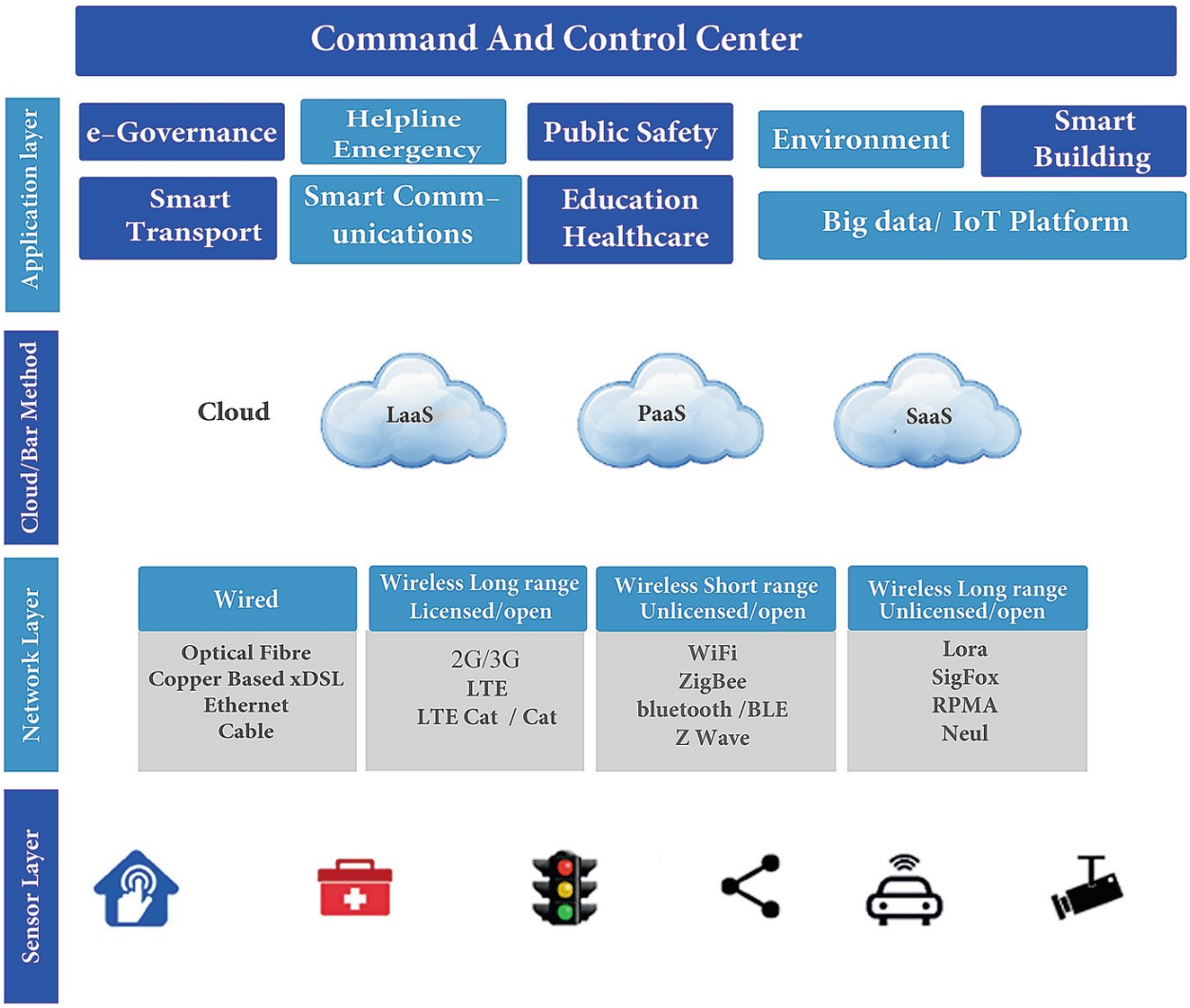

Figure 3.1: IOT Structure

IoT data capture and processing process, which occurs as follows: the data generated by the physical objects are read by the interrogating device, also known as the gateway. This data is sent to the information management system, from there, using the Internet or other network technologies, other data associated with the objects or the initial data itself is obtained. The result of processing by an application or service is a performance in the environment in which they are the objects are present and / or a performance on them [6] As noted in [7] three classes of IoT devices are proposed:

i. Purely passive objects with identification and fixed data, are objects present in the Physical Interface Zone

ii. Objects with moderate computational power and context perception, which through sensors can generate messages and vary the information associated with them according to the time and place, represented by the objects present in the Physical Interface Zone

iii. Objects that have network connectivity, without human intervention, enabling the emergence of intelligence in network systems, represented by the objects present in the Physical Interface Zone.

The model illustrated by Figure 3.1 can be exemplified through an IoT-based process [34] In a given process, a real-world object has information, for example, inside an electronic tag. The information is retrieved through an interrogator, which can be connected to a service or application, in which users or applications, who are in the real world, can read or share the object's information in real time. Finally, this application can act in the environment, through the actuators present in objects [21] It should be noted that the electronic tag can be replaced by other unique identification technologies, that is, it is not restricted to RFID technology.

3.1.1 IOT Security Challenges

There are several protection issues with IoT because of these unique features and specifics outlined by [8], such as: connectivity, a feature that involves a great numbers of devices communicating with each other, as well as a protocol interaction that influence one another; Internet of Things will bring a lot of opportunity, but it has a lot of risks; because of this, it is challenging to secure all the methods.

Specification constraints are linked to computers, such as memory and battery use Myriad is a common term to describe the high number of IOT (Internet of Things) devices in use an unchecked [device] will function will remain in a long time without servicing due to the fact that they are placed where physical access is not possible. Nitty-gritty-gritty; types of data with regard to the personal details of people When the number of different hardware devices increases in popularity, new reasons for communication appear on the smartphone side of IoT; finally, completely IoT would likely takes its place in a range of social applications at some stage in the future, and for that purpose could come to be defined as “omnipresent” While you can see the features as liabilities, malicious agents have recognized them as vulnerabilities. These are in the following variations: Since interdependence is a function, an intruder would probably attempt to

manipulate the goal or its surrounding setting. Since several of these machines don't have advanced security algorithms, they're easy to attack. unmonitored, neglected Malicious agents are often found in operational settings, challenging to spot. Certain components in this function enable malicious agents to determine when a home is inhabited, enabling them to access various levels of personal information Usage of this function provides the opportunity for several computers to become infection vehicles, allowing for quicker distribution of malware. An inescapable characteristic with IoT systems is that they are often used with passwords being set to the defaults. Before the manufacturer's end consumers improve, the IoT feature would be a problem. CAs the amount of Internet of Things (IoT) devices has grown, target infrastructures such as cities and towns have been identified.

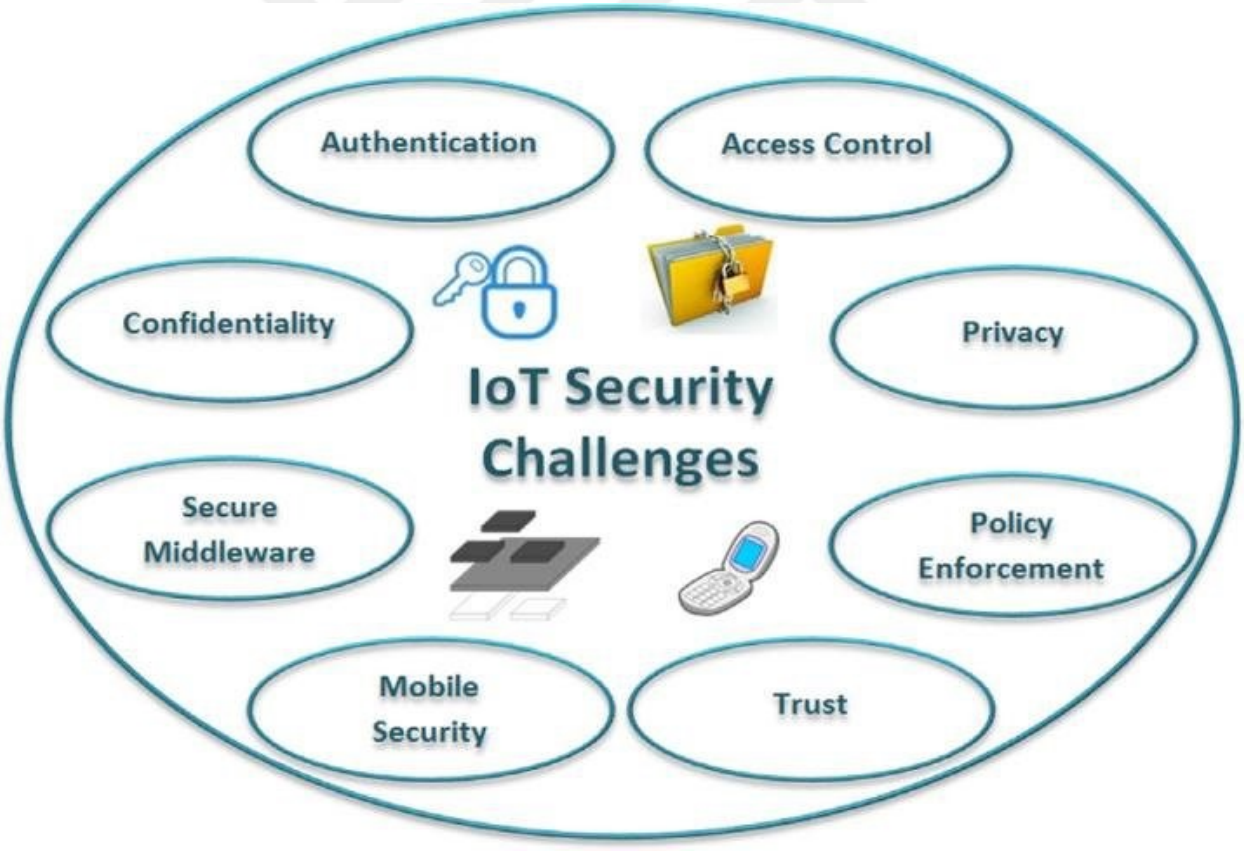

Figure 3.2: Security Challenges of IOT

3.1.2 Detection of Botnets

The detection of botnets is usually carried out by intrusion detection systems and, for this reason, the detection of botnets has characteristics common to these systems. Botnets are networks of machines infected by malware, which can be used to trigger large-scale attacks, which have already reached 1Tbps traffic [10]. They are robust and flexible networks, which are constantly improving their potential for contamination, which is why it is considered a challenge related to information security today [12]. Usually a botnet consists of 4 main parts, according to [1]: the first is the bot and its job is to spread the malware to other devices, and also to participate in orchestrated attacks on targets, previously selected by the botmaster; the second part is command and control (Command and Control - C&C), a user interface used by the botmaster to check the status of each device that is part of the botnet, and also to define new targets; the third part is the loader, its function is to provide the correct binary of the malware for different architectures, for this it contacts the victims; the fourth part is a reporting database, in the which all botnet related events are stored. the operation of a botnet is shown, and it occurs as follows. In step 1 in Figure 1, there is an infected bot that comes into contact with a potential victim, after that, in 2 the same bot reports to the report server the discovery of a new device. At moment 3, the botmaster initiates the action of checking the status of the report server, he does so through the command and control interface. Upon verifying the existence of a new target, it then begins steps 4 and 5, in which it sends an order for the loader to send a specific malware version for the target's architecture in question. The steps previously presented are those necessary for the botnet to grow, such a process can last as long as the botmaster wishes. At a given moment, if it is of interest to the botmaster, he can start steps 6 and 7 in which are the attack steps, for this it is enough that a target is defined and that a command is sent to all the bots that make up the network. Mirai, Hajime, Aidra, Bashlite, Dofloo, Tsunami and Wroba are examples of some of these botnets. As a security measure to mitigate attacks as described above, it is feasible to use intrusion detection systems, these systems work by detection of attacks, and execution of countermeasures. Although this type of system already performs optimally in traditional networks, there are still security challenges to be solved so that they work fully in Internet of Things [13]. One of the challenges involves the fact that the detection system should not affect the performance of the standard activities of the IoT device [14]. Traditionally, these systems can

be classified based on some guidelines, such as: IDS detection method and positioning strategy [13].

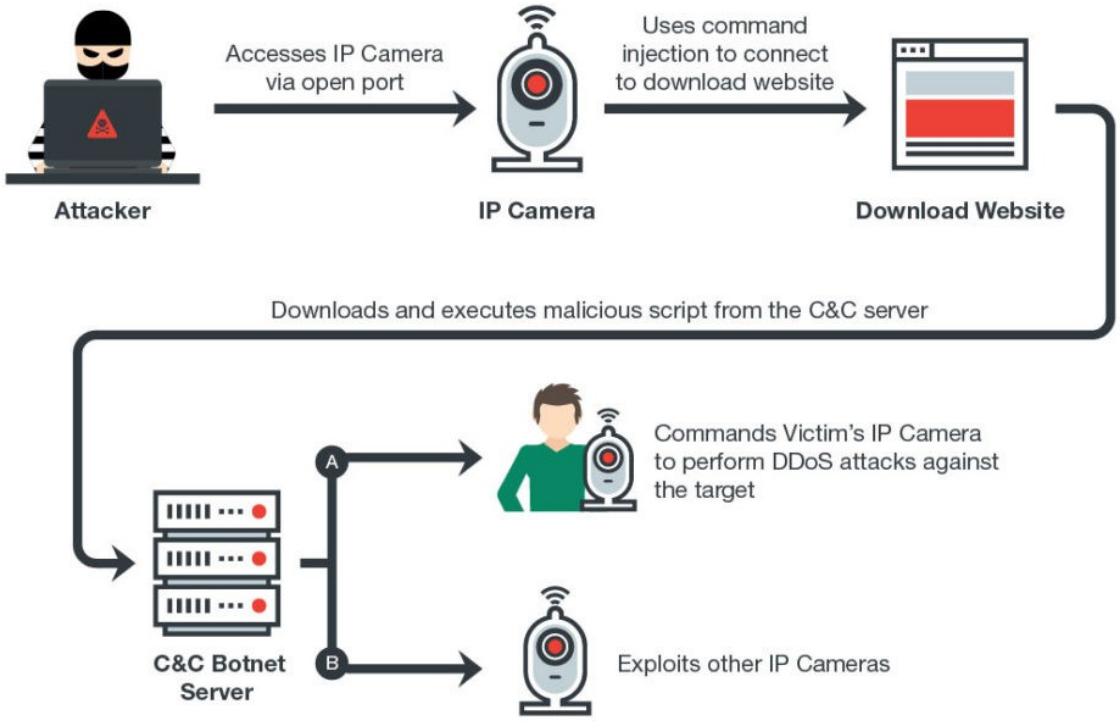

Figure 3.3: IOT Botnets

The method of detection is divided into 4 categories, defined by [13], as:

i. Anomaly: information regarding network functioning, hardware such as CPU usage, and software as system calls are previously captured. The captured information is processed to define the thresholds of normal behavior for the system in question. With the threshold set, it is possible to determine whether new behaviors are anomalous or not. This type of detection is not free from false positives, so techniques for treating false positives should be suggested.

ii. Signature: in this type of treatment, it is necessary to create behavioral profiles for each possible threat. Based on the profiles created, IDS will evaluate current events,

comparing them with the information that was given to it in a first time. The problem with this type of approach is that it is unable to respond to new threats, given that a signature has not yet been created. It is important to note that signatures are details about system calls, or some specificity about the network, for example.

iii. Specification: it is similar to the one based on anomalies, but the creation of the normal behavior profile is not performed automatically from the analysis of the system's functioning. In this case, it is an expert agent that will define what behavior is accepted for a given network and devices. Soon, the quality of the IDS is at the mercy of the ability of the specialist involved. If it has not defined good rules, the system can issue many false positives for threats, for example.

iv. Hybrid: it is when two types are combined, seeking the best functioning of each one. For example, combining anomaly detection with signature. Although an attractive strategy, developing this type of solution can be very complicated.

The positioning strategy is what defines where the data will be captured and analyzed. There are 5 types of positioning, as described in [13]: purely centralized, which occurs when the IDS is organized in a location in the system where it has access to all the information to be analyzed; purely distributed, when the IDS is disposed in some points of the network; and the hybrid, which is a more elaborate approach, but it is the one that can bring better results, considering that it overcomes the adversities encountered by the other two previous techniques - in this case, IDS can detect threats that circulate the network internally, and threats that are directed to the network by an external agent. In addition, within IDS there are devices responsible for capturing information, and as this information travels in two ways (internally to the device, operating system and hardware; and externally to the device, the network that connects the devices), then if necessary, existence of IDS in which the capture is based on network (Network-based IDS - NIDS), in this type of configuration the IDS connects to the network and monitors it for malicious traffic. And finally, IDS based on devices (Host-based IDS - HIDS), it is directly connected to a device, so it is feasible to detect anomalies internal to the system of the same.

3.2 DEEP LEARNING MODELS

The field of machine learning focuses on developing computer programs that can acquire new knowledge, learning, and skills, as well as well as improving on previously learned methods According to 15, which argues that since the advent of computers, the question of whether they could be programmed to learn has existed since then has been answered because they would imply that you could understand how to do so, this would be a substantial step forward in their development. Linking refers to extracting concepts from data, and thus is a process by which computers define things through descriptors and characteristics, and derives labels, so learning is the process of devising techniques to accurately predict new entries. Classical method: A classifier is used to teach techniques and the techniques are then applied to train the classifier The training procedure can be conducted in an unsupervised, semi-supervised, or fully supervised way. In supervised approaches, all training data are available for the machine learning algorithm. The model predicts that there is an external force that aids it in classification, enabling the algorithm to arrange samples when they mature based on shared characteristics over time. Unsupervised learning requires pick methodologies such as Consultation by committee and Uncertainty sampling; for methods that use those, unsupervised training samples are chosen by criteria, such as 'correctness' and 'accuracy' and 'call.' after this, the selected sample is sent to an oracle, who is responsible for classifying the sampler.

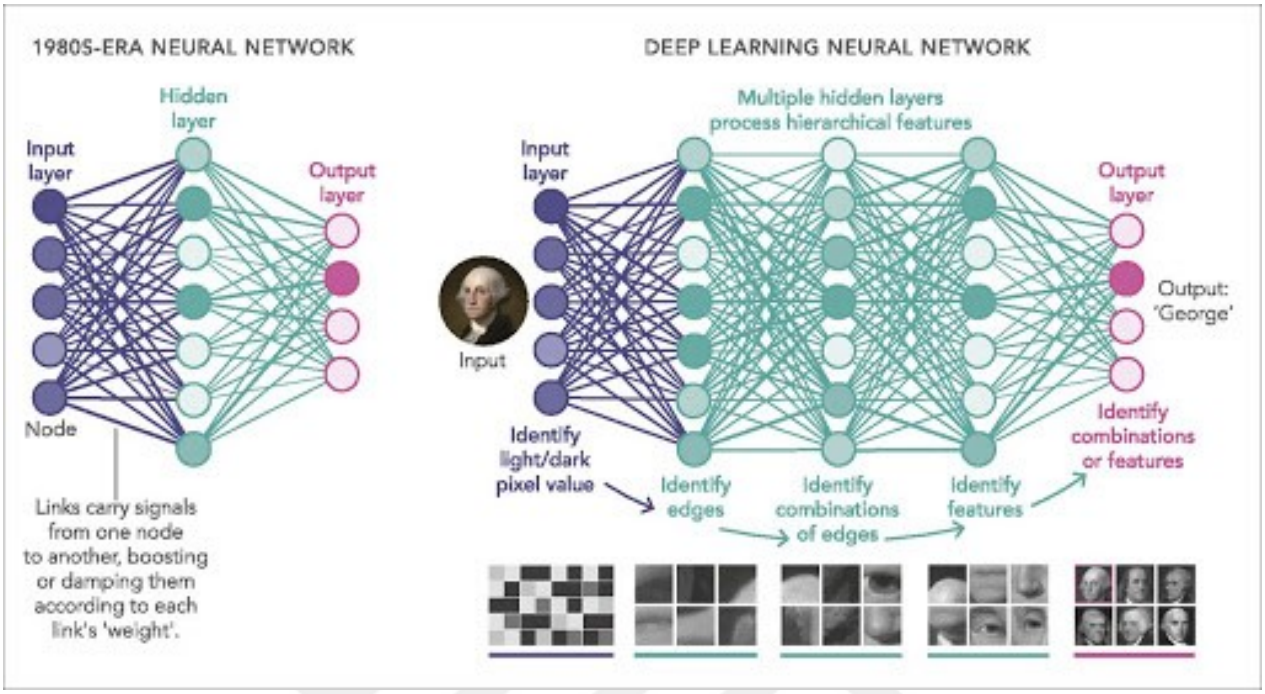

Figure 3.4: AI and Deep learning

Unsupervised approaches are those that use unstructured input values and categories in order to produce clusters. In short, you want to manage samples, without a guide oracle. In this semi-supervised system, we have input data that has labels and input data that doesn't have them Depending on the above, machine learning algorithms may produce various classifications. For an issue to have multiple performance values and also be characterized as a separate, independent, it has to have particular values. As long as a model's objects may be regressed to an analyst's predictions, it is possible to use classifiers to identify models of machine learning. regions may be beneficial as the issue as well as the expected temperature over the next several days in a particular location by contrast, classifiers are used while the results quality is more varied, for example when differentiating regular or exceptional network traffic. the more help vectors they have, the better (Support Vector Machine - SVM). When using a Support Vector Machine, you strive to find a hyperplane that will divide the problem space into predefined groups with maximum efficiency. It is possible to divide data in an ideal, while simultaneously finding the best separating lines by finding the support vectors. This scheme allows you to derive information from unpatented sets of data. These experiments seek to identify possible implementations of active learning in an IoT network. The interest in active learning grew

because active learning databases elicit less classifiers than machine learning classifiers have found.

It is recognized that marking databases is costly, which motivates the development of methods that can employ supervised methods with this in mind, we conclude that successful learning is an active quest. The choices for examples are made from well-defined metrics called sampling strategies. The use of data reduction makes for improved model convergence, since less unlabeled samples are found in the database Active learning may be used in various ways, the most common ones being Stream-Based and Pool-Based. The methodology that is applied to the input data determines the form of case, as does the sampling technique. The data comes with a certain quantity of samples per moment in this sort of situation. Thus, each new input must be examined when it is given to the algorithm to avoid getting a different result. These data were extracted from a variety of studies, using diverse sampling techniques. Algorithmically evaluating samples, the system can opt to discard the current sample, look for a new mark, or perform all functions at the same time. This can be rendered on the method of "some indicator of in formativeness," or sampling approach adding the found sample to the model enables the algorithm to self-update. Because the algorithm is given access to a vast numbers of cases early in the process, scenarios from pools will behave differently. The looping nature of the sampling techniques enables the application to inspect an enormous amount of data before choosing the sample. Following the procedure, the algorithm gives the result is given as follows: The algorithm allows a lot of measurements through the use of huge numbers of tests. A model with a defined search algorithm iterates through the data looking for a sample and, based on a technique, determines whether or not to use it (it is important to note that the computational cost of acquiring the data without labels is minimal). The oracle has been selected. Thus, the predictive model will keep up with the latest labeling discovered.

3.2.1 Sampling Strategy

There are three approaches to get through uncertainty in the results: "Uncertain Sampling", "Qualification Sampling", and "Questioning Sampling". For Sampling, it's easy to iterate to have the most helpful results. There are numerous ways to quantify information, some of which include using entropy. The Predictive Models technique utilizes constructing a panel of models, which acts as a jury, to determine issue samples. Using three graphical shapes, three models are

displayed: the red triangle, which is already labelled as a sample, and the green square, which serves as a classifier, which has no sample positions; and the unadorned rectangle, which symbolizes each committee member's duty to apply labels on samples on a Cartesian grid.

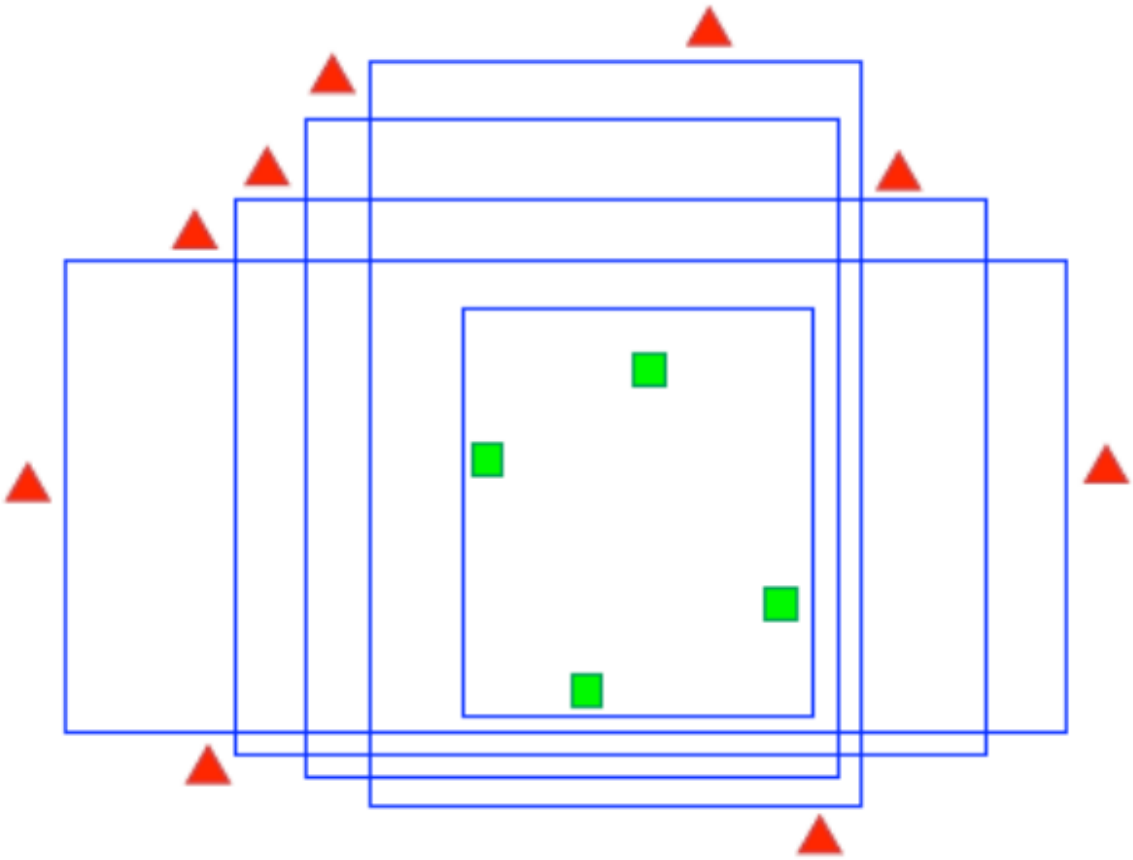

Figure 3.5: SVM data visualization in the classification process

3.3 ENVIRONMENT AND DATABASE

The experiments carried out used the database provided by Bezerra et al., Such a base was created from an environment in which profiles of IoT networks were emulated, with the objective of creating an environment in which botnets could be inserted and subsequently analyzed. For this, a Raspberry Pi model 3B (a single board computer with wireless and Bluetooth) was used, which has the Rasbian Stretch OS, with 1 GB of memory RAM and a 1.4

GHz ARM CPU, and also 4 computers, with Machine 1 and Gateway having 4 GB of RAM and 3.2 GHz Intel i5 CPU, and Machine 3 which has 8 GB of RAM and an Intel Xeon 3.1 CPU GHz. Two virtual machines were also used, which have the following configurations, 1 GB of RAM and 1 GHz CPU, and were emulated with the aid of a virtualization program installed on Machine 2 - all computers have the Ubuntu 17.04 OS. The scheme developed by Bezerra et al. Thus, for this work, the data collected related to the traffic of an IoT network emulated in different circumstances were used, as well as with different botnets inserted, considering that the Raspberry Pi was used as: Central Multimídia (MC), it is a device responsible for receiving a certain stream of video data and also accessing a central that provides software updates and installation; Security Camera (SC), is a device that captures video images, and transmits them to a central; and also an IP Security Camera with Additional Traffic (ST), the same as the previous one, but with additional configuration traffic, in which a user accesses the device through SSH connections, or Telnet.

3.4 FEATURE EXTRACTION

A primary goal of this is to introduce the data-LSVM principles and to an LS-SVM and PSO audience, with a little more depth The theoretical structure for a project can help to find out what problems, activities, theories, etc. are necessary to guide its evolution, and as well as providing a background in which the terms, algorithms, ideas, and any findings that are derived from such elements may be located. At the beginning, the individual would be considered, along with existing data collection practices, research methods, and assessment requirements, as the classifications are being made to provide information about the class. It is also explained how SVMs can be used in the issue in question, illustrating the theoretical foundations and operations, highlighting the LS-SVM explanation. The metaheuristics of combinatorial optimization PSO will be discussed, the algorithm's behaviors will be discussed, as well as its use for combinatorial problem classification will be analyzed, and how it has progressed since it was first developed will be detailed.

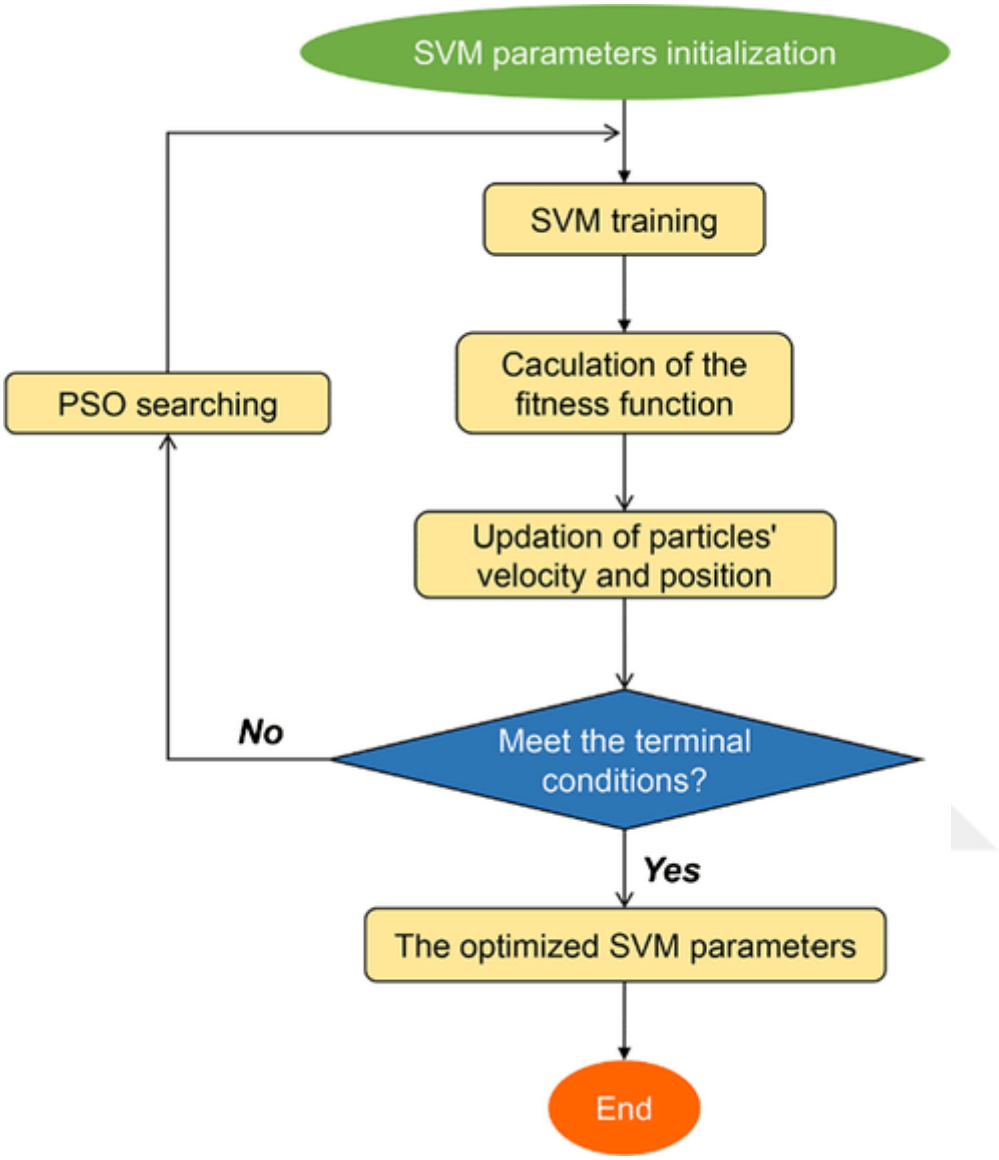

Figure 3.6: PSO-SVM Classification scheme

4.PROPOSED METHOD

4.1 DESIGNING THE IOT NETWORK

An entity-relationship model is a paradigm for comprehending various entities in a particular domain in order to agree on requirements and expectations. A reference model is focused on the primary principles, their interrelationships, and their connections to the world considerately tear-jerking an architecture acts as a point of reference for the development and device systems. A reference model has a point of reference. Two reference architectures are given here: one to developers and one to IoT architects The Reference Model is provided by ITU-T (2012) for the layers, as well as a more detailed overview of the ITU-T (or Internet of Things) Ecosystem According to the ITU (2012), the Internet of Things refers to a global infrastructure of content.

There is a need for complex services that are extended by the integration of physical and interactive devices and are often, dependent on the need for the interconnectedness of knowledge and connectivity protocols. Something in the physical or intellectual universe may be incorporated into the communications networks.

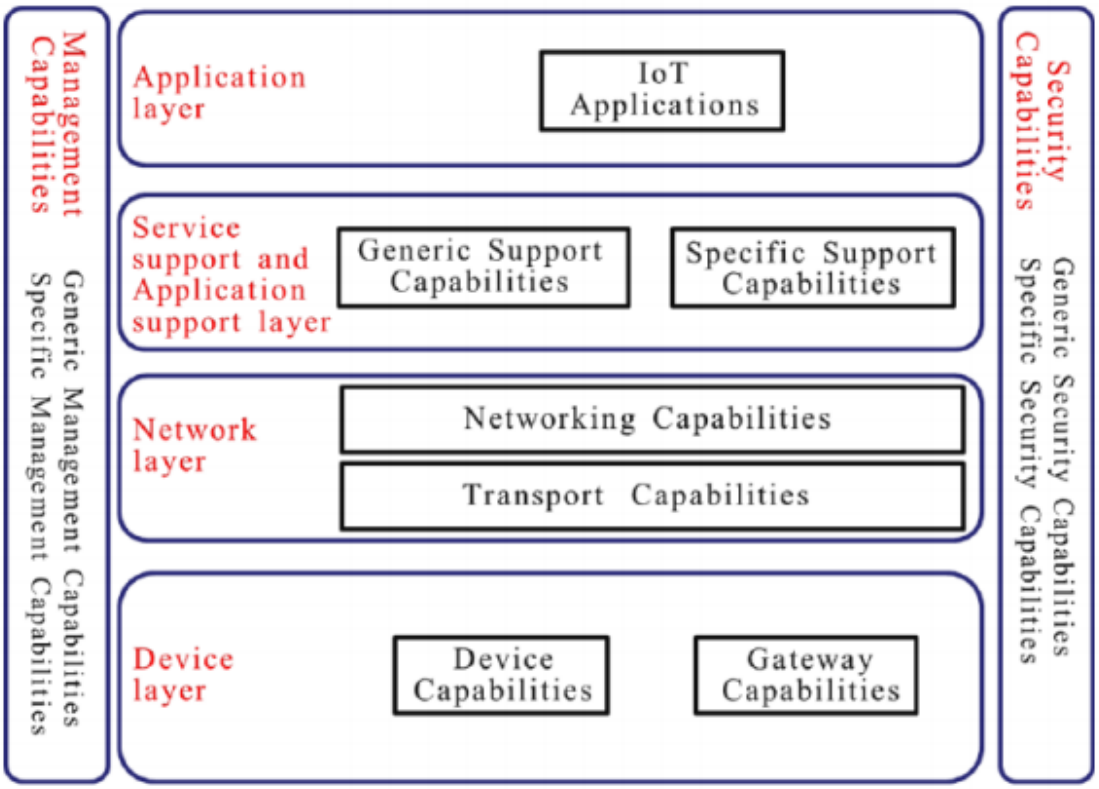

Figure 4.1: IOT Reference for network design

A detailed description of each of the 4 layers follows: On the layer above the platform, it includes solutions for IoT. These groups of features provide two separate categories of capabilities: support and application support, Typically, IoT application can use generic support capabilities including data processing and storage. You may define certain features based on unique attributes, such as features or attributes may be used to create certain features. Basic IoT features: These capabilities are meant for different IoT applications only. While specific implementations support mechanisms are provided for various IoT applications, it is important to group these specifics into discrete concepts.

The "networking" and "transport" layers found in the Open Systems Interconnection (OSI) model are different from the "networking" and "transport" layers found in the Open Systems Interconnection (OSI) model. Two levels of versatility are offered by this layer:

Permits network operations, such as network connection and transport control, telephone identity management, verification, and authorization. the ability to move Internet of Things (IoT) files, software, and knowledge around high-end design and low-end design. often contains but is not restricted to Gathering and submitting details directly to network devices without utilizing any gateway feature Through that network, the designers may even collect input or carry out instructions. On the other side, the computer gathers and sends information over the network through the communication capability of the gateway. A hybrid system is able to quickly deploy for ad hoc or configurable networks that requires scalability and rapid growth. the computers with Hibernate and save features Support for this system is made available through wire or non-wire connections. Smart Gateways Besides these features, gateways can include:

Help for various interfaces: the CAN bus, Bluetooth, and Wi-Fi systems all use separate layers Many functionalities at the network layer, such as well as publicly owned switched telephone networks (PSTN), Long Term Evolution (LTE) networks, and Ethernet will all operate together.

Conversion of functionalities are important as connectivity happens at both the computer and at the network level, as Bluetooth does layer.

4.2 PSO-SVM WORKFLOW

IDSs typically use a wide variety of methodologies for the detection process, which are intended to broaden their scope and accuracy. There are three methodologies that are the most used and each one of them has advantages and disadvantages compared to the others. However, these can

be integrated, complementing each other to benefit your objectives. Below is a brief description of each of these methodologies based on what established in [5].

a) Signature-Based Detection This methodology is based on known attacks and uses simple comparisons to conclude whether what is being detected matches an attack or not. In short, it is a considerably useful technique against known attacks, however, its performance changes when faced with variations of these and when analyzing attacks that are not in its known data set.

b) Anomaly-Based Detection

This detection is based on the behavior of all that activity considered normal within the system that is being audited and will determine that an event is anomalous when its behavior has a deviation from that considered normal. To resolve normal activities, user profiles, host, networks, etc. are created. The disadvantage of this methodology is that in the case in which the profiles are created automatically, the situation may arise in which, while

create the profiles, an attack that is being carried out in a subtle and periodic way could easily fall into one of the profiles. Furthermore, if the attack is subsequently intensified, it is likely that it will not be detected given the little deviation in the observed behavior. In the other side, it is a perfect way to combat new threats.

creative explanation Predefined profiles of benign protocol states for each state are used to distinguish differences between states. "Stateful" indicates that the IDS is able to recognize and keep track of how network, transport, and device protocols think of themselves. There are several methodologies that can be used to model and classify natural and abnormal patterns, including immune systems, mathematical methods, file and corruption detection models, neural networks, dedicated languages, and genetic algorithms. [6].

4.3 IDS DESIGN

An Intrusion Detection System, or simply IDS, is an automated system that constantly examines the data traffic that flows through a computer network, in order to detect abnormal events. In the case of discovering behavior of this nature, the IDS must react and respond in some predetermined way, such as, for example, making a detailed record of the event that has occurred or executing an alert. In a similar way, it could be said that the operation and objectives of an IDS are similar to the work that a security guard must carry out in a private room. The guard

must be aware of all access points for those who enter or leave the perimeter that corresponds to monitor. In addition, said character should be continuously observing the images captured by surveillance cameras, in order to go checking that everything is in order. In case of detecting an unknown, suspicious person or some irregular behavior, the guard should systematically proceed to carry out a certain action. For example, questioning an unknown visitor who has just entered or giving immediate notice to the police in the event of a crime that is untenable or difficult to handle.

In order for the private area and everything it houses to be safe from the types of behavior mentioned, a highly trained person must be available to carry out such work, so that, in the face of any eventuality, everything results from the

best possible way. In other words, the security guard must have the ability to distinguish between normal activities and those that are not, in addition, he must react efficiently and effectively to minimize the damage that could eventually be caused. This is how an IDS that monitors a certain network should work. To achieve this, robust processes are necessary that have high accuracy when checking or analyzing a new event in order to determine, with full security, what kind of activity it is facing. The IDS concept was coined in the 1980s. Since then it has evolved from the use of statistics to detect anomalies, through expert systems based on rules, to the use of data mining to find characteristic patterns. Today, intrusion detection is the process of monitoring the events that occur in a computer system or network, with the aim of analyzing and searching signs of potential incidents: threats of (imminent) violations of computer security policies, acceptable use policies, or standard security practices. An IDS is, technically, software that automates the previously mentioned intrusion detection process [5].

According to Rebecca Bace and Peter Mell [2], most IDS can be described in terms of three fundamental functional components, which are presented below.

a) Information sources they correspond to the different sources of event information used to determine when an intrusion is taking place. These sources can be obtained from different levels of the system, the most common being those for monitoring the network, host and applications.
b) Analysis It is the section of the IDS that currently organize and give meaning to the events derived from the information sources, deciding when these events will indicate that an intrusion is taking place or if it has already taken place. The most common

analysis approaches are "misuse detection" and "anomaly detection". Misuse Detection is based on known anomalous behaviors to detect anomalies according to certain patterns or signs, and Anomaly Detection, in activities that are normal or regular for the network, in order to determine or classify as anomalies all those behaviors that are different from those already known as common or normal.

c) Response It is the set of actions that the system takes when an intrusion is detected. They are typically grouped into active and passive measures. With active measures, some automated interventions are carried out by the system, and with passive measures, the results of the IDS are reported to humans, who, finally, are the ones who will have the option to make a decision.

5. SIMULATION AND RESULTS

This Section is dedicated to the presentation of experimental results made with the proposed algorithm and its discussion.

5.1 DATABASE AND PERFORMANCE

Complementary standards are used to judge the strength of the IDS procedure, and this is dependent on what kind of occurrence and how it is graded. It is important to recognize which divisions the incidents fall into place in order to be able to carry out the test. It should be remembered that, with this work in mind, that the IDS normal/anomaly categorization applies to incidents, not to data. In this scenario, the method arrived at its conclusion after evaluating four possibilities. Generalized observations in Table 5.1 are seen in column 5: "+" indicates an attack and "-" in column 5 denotes an answer.

Table 5.1 the response in classification phase

RESPONSE +	**TR**	**FP**
RESPONSE -	**FN**	**TN**

a) **TP (True Positives): Knowns as "hit". it is when there really is an intrusion and it is correctly detected.**

b) **TN (True Negative or True Negatives): When an intrusion has not been carried out and no response has been given as if it were on.**

c) **FP (False Positive or False Positives): Non-existent intrusion, but some event has signaled the IDS to give a response, also known as a "false alarm".**

d) **FN (False Negative): Known as "failure". It is when there is a real intrusion, but the IDS did not have the ability to detect it.**

Table 5.2: Malicious activity log and the recorded features

Destination port	Pkt count	pkt size	Email Reputation	web Reputation	Ip Reputation	pkt Size	Country Score	Malicious or Not
25	4	1920	0.5	0.5	0.5	0.1	1	0
25	4	1920	0.5	0.5	0.5	0.1	1	0
25	4	1920	0.5	0.5	0.5	0.1	1	0
443	1	512	1	0.1	0.1	0.1	1	1
443	1	480	1	0.1	0.1	0.1	1	1

5.2 EXPERIMENTS AND RESULTS

Based on the previous quantifiers, the most used indices in binary classification are shown below.

i. Sensitivity

Known as the True Positive Rate, it measures the proportion of TP that have been correctly identified as such. That is, the percentage of intrusions that are identified as actually having such a condition. In the case of IDS, the sensitivity is also known as the "Detection Rate" and is calculated as follows:

ii. Specificity

Known as the True Negatives Rate, it measures how specific the classifier is. That is, it measures the proportion of TN that are correctly identified. That is, the percentage of normal events that are classified as such or for not having intruder status. Specificity is calculated as follows:

iii. Accuracy

It is the degree of closeness of the measurements of a quantity to its real value. That is, the proportion of true results (TP and TN) in the population. It is the criterion that is most used in the testing phase and is calculated as follows:

- Accuracy: Known as repeatability, it is the degree to which repeated measurements under unchanged conditions show the same results. That is, the ratio of TP against all positive results (TP and FP). Accuracy is calculated as follows:

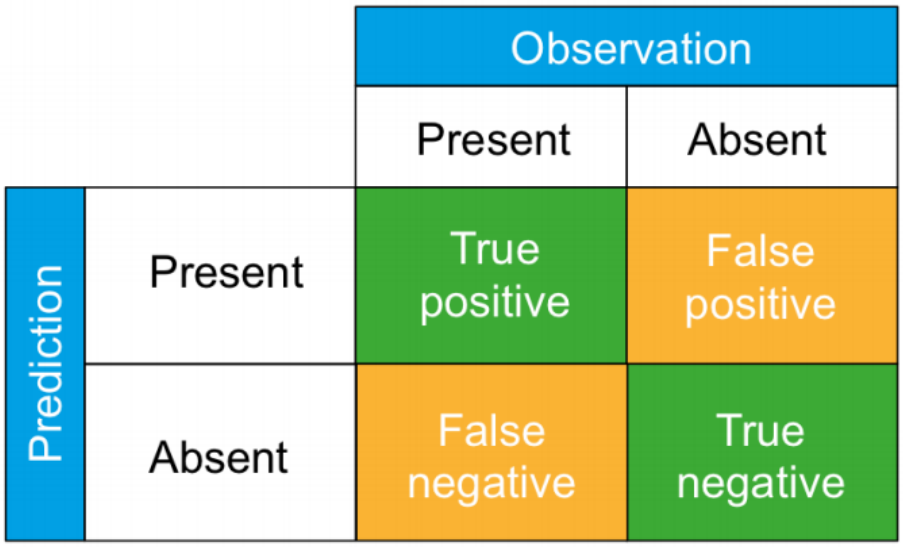

Figure 5.1: contingency or confusion matrix

The ideal performance for a classification model is to hit all the types of events analyzed, that is, to obtain 100% of TP and TN, or in other words, 0% in FP and FN, which would make all of them newly indexed. exposed coincide in reaching 100% in their results. However, when faced with real and highly complex classification problems, such as the IDS problem, the path to achieving the expected perfection is difficult. It should be considered that the behavior of the activity of a network is dynamic and chaotic, which implies a great challenge at the time of detecting and differentiating patterns correctly. When the data transmitted over a network is valuable and, moreover, is at stake, it is preferable that the nature of an IDS is more onservative, in other words, it is more convenient that an alert is generated for a normal event, than be inclined to ignore anomalous activity and fall victim to some kind of threat. However, one should not fall into extremes, an IDS could eventually have a detection rate of 100%, which would be ideal, but this would not make sense if at the same time the system maintained a false rate extremely high alarm. Under these conditions, the use of such a system would not be justified, since classifying all activity as an intrusion would generate such a number of alerts that the normal behavior on the network could be totally altered or interrupted. In a case like this, the IDS could congest the network with alert messages or, if this were even more preventive, it could lead to a continuous traffic cut for all types of events occur, regardless of their nature. Based on the above, it is necessary for an IDS to be effective, but without falling into extremes. Therefore, when evaluating a classifier, it must have a small tendency to decrease its Specificity,

but maintaining a high Sensitivity so that its behavior is more conservative. One way to find out how the classification model is trending is to rely on the detection rate and the false alarm rate. There is usually a balance or "trade-off" between the Sensitivity and Specificity measures, or in other words, between the detection rate and the false alarm rate (1-Specificity). This balance can be transformed into a complementary tool to help measure the goodness of the classification model. In order to be able to represent graphically, this relationship is that a graph called Receiver operating characteristics or, simply, ROC curve is used.

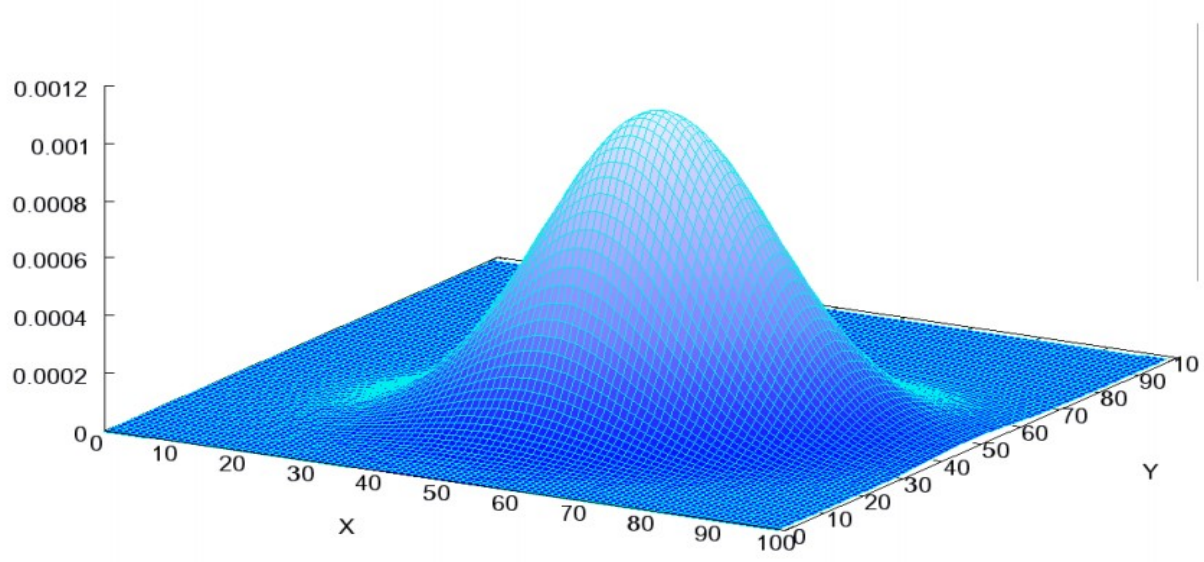

Figure 5.2: ROC curve

The ROC curve is a two-dimensional graph that represents relative trade-offs between benefits (TP) and costs (FP). Generally, it is used to compare different classification models. The best possible prediction of a method will produce a point in the upper left corner or in the coordinate (0,1) of the ROC curve space, representing 100% sensitivity (without FN) and 100% specificity (without FP). The point (0,1) is called "the perfect ranking". According to the authors of [3], the ROC curve shows the reparative abilities of a binary classification model. If area under the ROC curve is equal to 1, it means that the model was able to correctly separate each of the data points involved in its training process. In the case that the area under the curve is equal to 0.5, it means that the constructed classifier does not have any discrimination power. If you weren't even able to separate the data points, your ability to classify data correctly in a test phase will be less.

According to Fawcett's interpretation [7], the AUC (Area Under an ROC Curve) is equivalent to the probability that a classifier determines a positive instance higher than that of a negative one,

considering that both are chosen randomly. This is, from the IDS perspective, that the probability of correctly classifying a randomly chosen event is greater when it is an anomalous event than a normal or non-intrusive event. In short, it is much more desirable for an IDS to correctly classify anomalous events than normal ones. However, it should be noted that this is only a probability and that there will always be room for error.

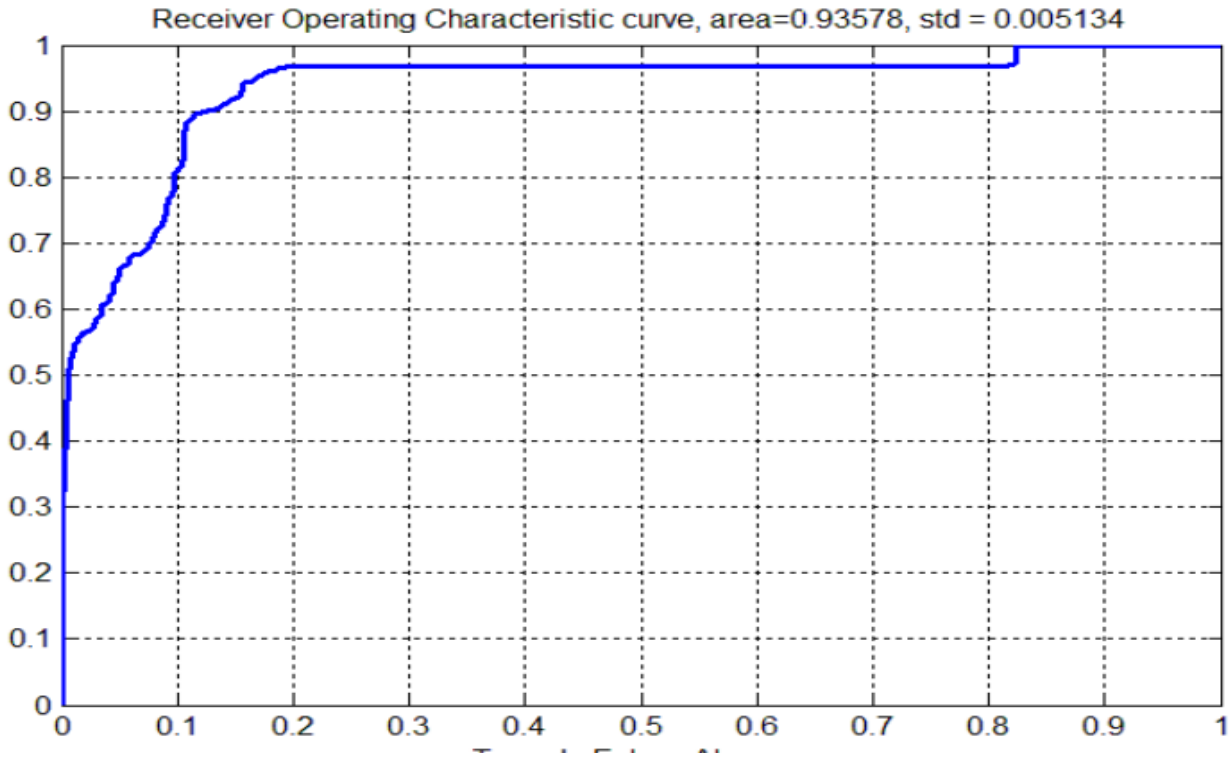

Figure 5.3: Accuracy by iteration

To carry out reliable evaluations of classification models for the IDS problem it is necessary to use an appropriate data set. That is, the data set has been originated, as far as possible, from real data or else from complex simulations of the common behavior of events in a network. After having the base data, it is convenient that these be refined, eliminating redundancy and unusual distributions. Additionally, it is advisable to use data that has been tested or analyzed in other publications since they will be useful to make comparisons between results, follow suggestions and good practices that guide the good development of the project. Regarding the data audited in this work, a data set widely used by most of the work done on IDS in the last decade was available. Therefore, from the beginning there is the advantage that there is a wide availability of useful information related to the data set, which has also been increasing or improving over time

and also, the aforementioned results that are very useful when making comparisons with other classification models, even when they have been developed with other methodologies. K99 IDS data are made public at the KDD case, and is referred to as the K99 data package. This database simulates various intrusions that may occur in a military network. This data are created and handled by the Lincoln Labs "Intrusion Detection Research" software." In the year 1999, the competition sought to develop a network intrusion detection system, which was capable of discriminating between “in concerning” (informative) links (intruders, assault situations) and “mild or safe” ones, thereby earning it the title of “The Third International Knowledge and Data Acquisition Competition." This is accessible via the University of the University of California, Irvine, which has a section on their webpage where you can see all of the details broken down by category and percentage of failure.

Table 5.3: Features of the normal and abnormal activity

Source IP	Source Port	Destination IP	Destination Port	Pkt Count	Pkt Size	Malicious or Not
202.54.250.39	36728	23.20.239.12	25	4	1920	0
202.54.250.39	51100	18.213.250.117	25	4	1920	0
202.54.250.39	51110	18.213.250.117	25	4	1920	0
202.54.250.39	52932	52.4.209.250	25	4	1920	0

corrected versions, among other things. In other words, the files that have 100% of the data for training and tests contain 4,898,431 and 311,027 data points respectively. In addition, each data point is composed of 41 characteristics (see Appendix 1) and a label that categorizes it, this can be "normal" or one of the four pre-established subcategories of attacks (see Appendix 2). In short this translates into a matrix of almost 5 million rose by 41 columns freely available to researchers so that they can train and evaluate the performance of their classification models. Regarding the data classes of the KDD'99 set, as mentioned above, there are those of normal data and anomalous data. The normal data corresponds to the normal events carried out by the users of the network, whereas the anomalous data, or simply attacks, are the events deliberately caused by attackers to the network. Furthermore, this last category can be divided into four

subcategories: Command Injection (R2U), Denial of Service (DoS), Stealing User Credentials (S2U), Permissions Amplification Here is a list of the "Malicious types" protections which are included in KDD. Blocking access to access a server denying service is a kind of DoS assault where the intruder allows memory or computing to be overused, thus blocking legitimate customers. An example of a hostname to log from: The Apache server, the Back script, the TeLEGS Flood script, Process attack, the SYS Logs script, the Process attack kit, the Smurf script, and the Sys Logs log. malicious user/administrator attacks a kind of assault in which the intruder obtains user authentication on the server and gains root access from there. Notably, include 'Eject', 'Fboots.fbasic', 'Fdformat', 'Ps', and 'Xterm'. Simple to obtain and hard to defend

A compromise is similar to an assault where the intruder sets up a computer on a network without a user account, then attempts to leverage a local privilege escalation loophole to achieve administrator access. the Creative+ offers URLs, Imap, named, sftpw, smtp, sendmail, xlock, and xnoop.

An intrusion type that involves scanning a network to discover weaknesses and details is called a Penetration Testing. There is a correlation with being familiar with a network location, systems, facilities, and the ability to find weaknesses. Some web-analysis programs include Ipsweet, Nmap, and Saint. Although having the original KDD records, the results obtained with this classifier will not be reported in that state. During this recent intensive study, these issues were observed in 2009. When data from an updated classification was evaluated against, these issues were revealed.

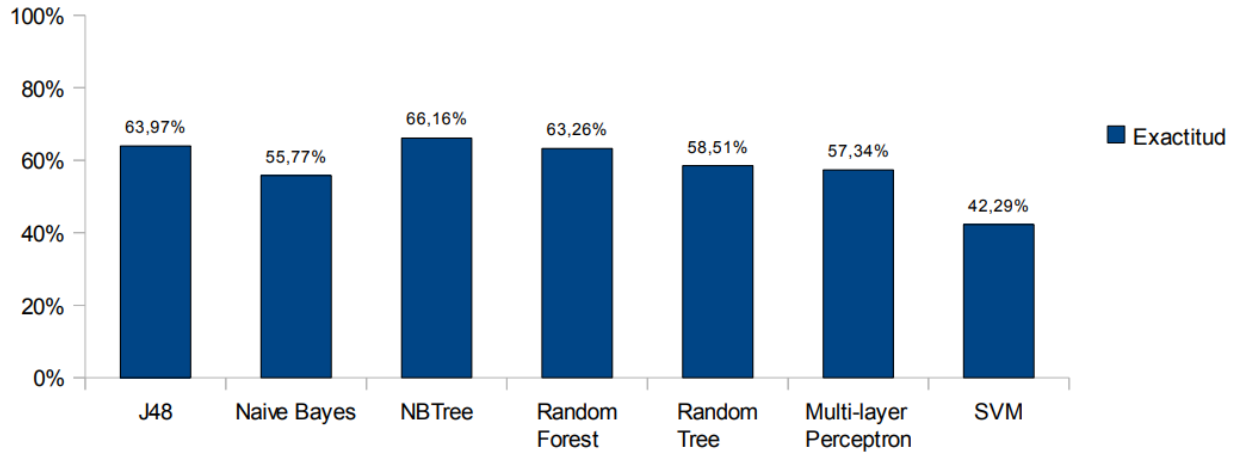

Figure 5.4: Performance of selected learning algorithms on KDD Test

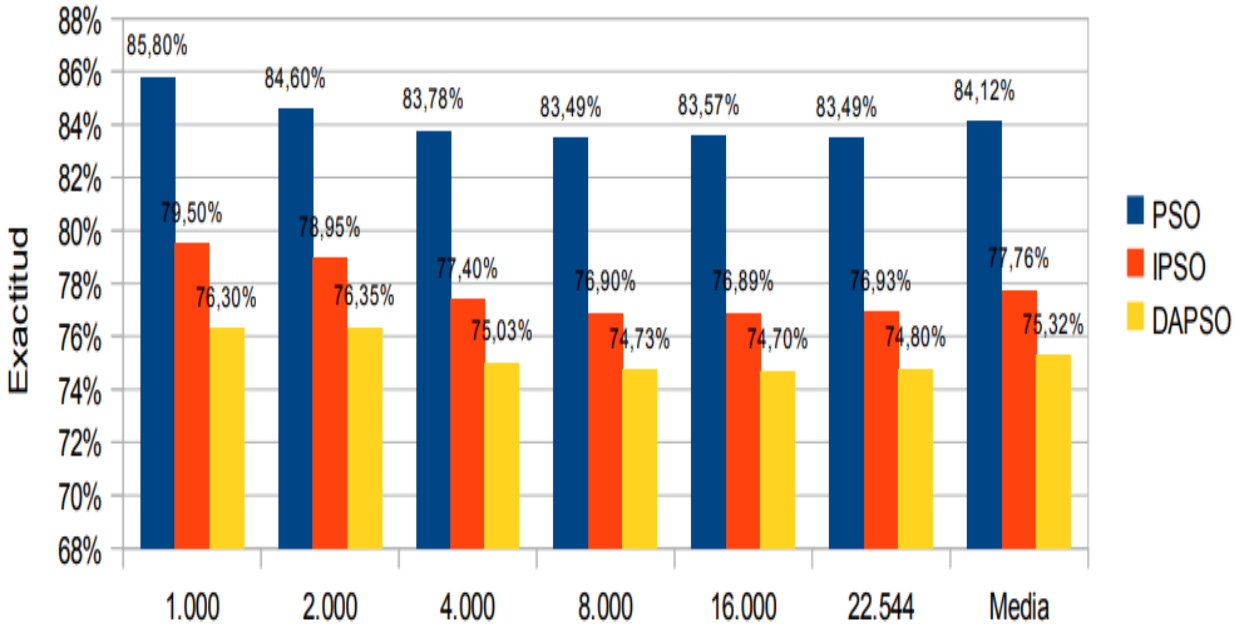

Figure 5.5: Accuracy of Models Created by PSO Variants

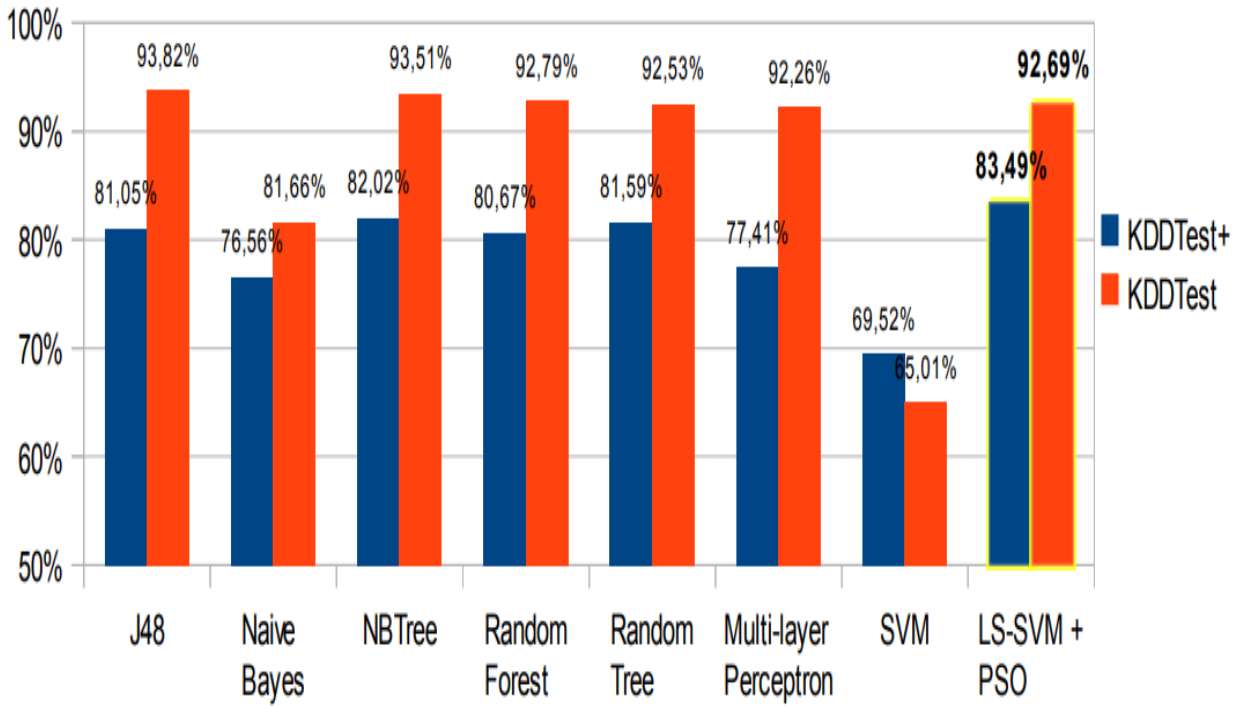

Figure 5.6: Classifier Accuracy vs. Other Models

6. CONCLUSION

Today's Internet of Things (IoT) networks have been broadly adopted in many sectors. It has been shown, moreover, that its characteristics have been used by hostile parties to orchestrate attacks on target systems. The current number of botnets that seek to compromise IoT devices has increased from millions to over the course of a single day to billions of devices regular operations. The machine learning-assisted technique is useful in the identification of botnets. Because of this, an investigation into the identification of botnets has been completed in the sense of IoT, various machine learning methods have been studied. Due to the additional computing and human expense involved with the conventional supervised machine learning for IoT, traditional supervised learning methods are infeasible. It was based on an IoT network that uses three distinct profiles. In this experimental database, an active learning pool was set up, and an ambiguity sampling technique was used. An example of active learning making it possible to reduce the amount of oracle consults involved in botnet identification was shown with the findings. Experiment 2 shows that the amount of classified samples used to generate an effective model can be reduced from 15% to 2% by using active learning to determine the accuracy of this model's use, the amount of samples included in the model's initial stage was decreased from 15% to 0.5% Therefore, relying on models that are as predictive As a result, it's understood that a deep learning methodology such as active learning may be used to identify IoT botnets. The long-term objective of further research is to investigate and confirm all characteristics of the database, and explore their impact on consulting, and also to double the amount of kinds of database sampling strategies.

REFERENCES

[1] Anderson J. P., Computer Security Threat Monitoring and Surveillance, 1980. James P. Anderson Co., Fort Washington, Pa.

[2] [Bace R. and Mell P., NIST Special Publication on Intrusion Detection Systems, 2002. Disponible vía web en: www.21cfrpart11.com/files/library/.../nist_intrusiondetectionsys.pdf Revisada por última vez el 01 de Julio de 2010 National Institute of Standards and Technology

[3] Suykens J.A.K. and Vandewalle J., Least Squares Support Vector Machine Classifiers, 1999. Neural Processing Letters, vol. 9, no. 3, pp. 293–300.

[4] Kennedy J. and Eberhart R., Particle swarm optimization, 1995. In Proc. IEEE Int. Conf. Neural Netw. (ICNN) vol. 4, pp.1942–1948

[5] Scarfone K. and Mell P., Guide to Intrusion Detection and Prevention Systems (IDPS), 2007. Disponible vía web en: csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf. Revisada por última vez el 01 de Julio de 2010. Special Publication 800-94, Recommendations of the National Institute of Standards and Technology

[6] Verwoerd T. and Hunt R., Intrusion Detection Techniques and Approaches, 2002. Disponible vía web en: http://sureserv.com/technic/datum/pdf/ids/english/Intrusion%20Detection%20Techniques%20and %20Approaches.pdf Revisada por última vez el 01 de Julio de 2010 Department of Computer ScienceUniversity of Canterbury, New Zealand

[7] Fawcett T., An introduction to ROC analysis, 2006. Pattern Recognition Letters 27 (2006) 861–874

[8] KDD, Knowledge Discovery and Data Mining Cup 1999 Dataset, 1999. Disponible vía web en: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, Revisada por última vez el 28 de Junio de 2010

[9] Tavallaee M., Bagheri E., Lu W. and Ghorbani A., A Detailed Analysis of the KDD CUP 99 Data Set, 2009. Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009)

[10] NSL-KDD, The NSL-KDD Data Set, 2009. Disponible vía web en: http://nsl.cs.unb.ca/NSL-KDD Revisada por última vez el 29 de Junio de 2010

[11] Aizerman M., Braverman E. and Rozonoer L., Theoretical foundations of the potential function method in pattern recognition learning, 1964. Automation and Remote Control 25:821--837.

[12] Poli R., Kennedy J. and Blackwell T., Particle swarm optimization. An overview, 2007. Swarm Intelligence, 1(1):33-57.

[13] Del Valle Y., Venayagamoorthy G., Mohagheghi S., Hernandez J. and Harley R., Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems, 2008. IEEE Transactions on Evolutionary Computation, vol. 12, no. 2, pp. 171–195.

[14] Shi Y. and Eberhart R., A modified particle swarm optimizer, 1998b. In Proceedings of the IEEE internationalconference on evolutionary computation (pp. 69–73).

[15] Clerc, M. and Kennedy, J., The particle swarm--explosion, stability, and convergence in a multidimensional complex space, 2002. IEEE Transaction on Evolutionary Computation, 6(1), 58-73

[16] Eberhart R. and Shi Y., Comparing inertia weights and constriction factores in particle swarm optimization, 2000. In Proceedings of the IEEE congress on evolutionary computation (CEC) (pp. 84-88), San Diego, CA. Piscataway: IEEE

[17] Refaeilzadeh P., Tang L. and Liu H., Cross-Validation, 2008. Arizona State University 59

[18] Kohavi R., A study of Cross-validation and Bootstrap for Accuracy Estimation and Model selection, 1995. 1995 International Joint Conference on Artificial Intelligence (IJCAI)

[19] WEKA, Waikato environment for knowledge analysis (weka) version 3.5.7. Disponible vía web en: http://www.cs.waikato.ac.nz/ml/weka/, Revisada por última vez el 30 de Junio de 2010

[20] Wu S. and Yen E., Data mining-based intrusion detectors, 2009. Expert Systems with Application`s, Vol. 36, Issue 3, Part 1, pp. 5605-5612.

[21] Chang C. and Lin C., LIBSVM -- A Library for Support Vector Machines. Disponible vía web en: http://www.csie.ntu.edu.tw/~cjlin/libsvm/ Revisada por última vez el 30 de Junio de 2010

[22] Wang J., Hong X., Ren R. and Li T., A Real-time Intrusion Detection System Based on PSO-SVM, 2009. Proceedings of the 2009 International Workshop on Information Security and Application (IWISA 2009)Qingdao, China

[23] Jiao B., Lian Z. and Gu X., A dynamic inertia weight particle swarm optimization algorithm, 2008. Chaos, Solitons and Fractals 37, 698-705

[24] Yang X., Yuan Jinsha, Yuan Jiangye and Mao H., A modified particle swarm optimizer with dynamic adaptation, 2007. Applied Mathematics and Computation 189, 1205-1213

[25] Dong C., Wang G., Chen Z. and Yu Z., A Method Of Self-Adaptive Inertia Weight For PSO, 2008. 2008 International Conference on Computer Science and Software Engineering

[26] LS-SVMlab, Least Squares - Support Vector Machines Toolbox. Disponible vía web en: http://www.esat.kuleuven.be/sista/lssvmlab/. Revisada por última vez el 5 de Mayo de 2010.

[27] PSO, Particle Swarm Optimization Toolbox, 2006. Disponible vía web en: http://www.mathworks.com/matlabcentral/fileexchange/7506. Revisada por última vez el 17 de Octubre de 2010.

[28] Mahmud W., N.Agiza H. and Radwan E., Intrusion Detection Using Rough Sets basedParallel Genetic Algorithm Hybrid Model, 2009. Proceedings of the World Congress on Engineering and Computer Science 2009 Vol IIWCECS 2009, October 20-22, 2009, San Francisco, USA

[29] Kayacik H., Zincir-Heywood A. and Heywood M., Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets, 2005. Proceedings of the Third Annual Conference on Privacy, Security and Trust, October 2005, St. Andrews, Canada.

[30] Boston, MA : Springer US, 2007, pp. 159–180. DOI : 10.1007/978-0-387-33112-6_7. En ligne : https://doi.org/10.1007/978-0-387-33112-6_7

[31] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis et al., “Understanding the mirai botnet”, dans USENIX Security Symposium, 2017, pp. 1092–1110.

[32] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, et C. Siemens, “DREBIN : Effective and Explainable Detection of Android Malware in Your Pocket.” dans Ndss, vol. 14, 2014, pp. 23–26.

[33] L. Atlas, R. Cole, Y. Muthusamy, A. Lippman, J. Connor, D. Park, M. El-Sharkawai, et R. Marks, “A performance comparison of trained multilayer perceptrons and trained classification trees”, vol. 78, no. 10. IEEE, 1990, pp. 1614–1619.

[34] L. Atzori, A. Iera, et G. Morabito, “The Internet of Things : A survey”, vol. 54, no. 15, 2010, pp. 2787 – 2805. DOI : https://doi.org/10.1016/j.comnet.2010.05.010.

[35] Website : http://www.sciencedirect.com/science/article/pii/S1389128610001568

[36] S. Babar, A. Stango, N. Prasad, J. Sen, et R. Prasad, “Proposed embedded security framework for internet of things (IoT)”, dans Wireless Communication, Vehicular

Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), 2011 2nd International Conference on. IEEE, 2011, pp. 1–5.

[37] L. M. Belue et K. W. Bauer Jr, “Determining input features for multilayer perceptrons”, vol. 7, no. 2. Elsevier, 1995, pp. 111–121. T. Bertauld et M. R. Dagenais, “Low-level trace correlation on heterogeneous embedded systems”, vol. 2017, no. 1, Jan 2017, p. 18. DOI : 10.1186/s13639-016-0067-1.

[38] Website: https://doi.org/10.1186/s13639-016-0067-1

[39] S. Bharadwaja, W. Sun, M. Niamat, et F. Shen, “Collabra : a xen hypervisor based collaborative intrusion detection system”, dans Information technology : New generations (ITNG), 2011 eighth international conference on. IEEE, 2011, pp. 695–700.

[40] T. Bird, “Measuring function duration with ftrace”, dans Proceedings of the Linux Symposium. Citeseer, 2009, pp. 47–54. 82

[41] M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares, et D. Haussler, “Knowledge-based analysis of microarray gene expression data by using support vector machines”, vol. 97, no. 1. National Acad Sciences, 2000, pp. 262–267.

[42] Z. U. Burguera, Iker et S. Nadjm-Tehrani, “Crowdroid : Behavior-based Malware Detection System for Android”, dans Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices, série SPSM ’11. New York, NY, USA : ACM, 2011, pp. 15–26. DOI : 10.1145/2046614.2046619. En ligne : http://doi.acm.org/10.1145/2046614.2046619

[43] J. Cannady, “Artificial neural networks for misuse detection”, dans National information systems security conference, vol. 26. Baltimore, 1998. B. S. Center, “Extrae”, https://tools.bsc.es/extrae. B. A. K. Chandola, Varun et Vipin, “Anomaly detection for discrete sequences : A survey”, vol. 24, no. 5. IEEE, 2012, pp. 823–839.

[44] AAMIR LAKHANI , HACKING, (2016), "WHEN BABY MONITORS ARE A MODEL FOR IOT SECURITY", available at http://www.drchaos.com/when-baby-monitorsare-a-model-for-iot-security/Accessed on 15th/05/2017

[45] MR. SHASHANK DHANESHWAR, (2015), "INTERNET OF THINGS APPLICATION FOR CONNECTED VEHICLES AND INTELLIGENT TRANSPORT SYSTEMS", Available at https://www.slideshare.net/shashankdhaneshwar/iotapplications-for-connected-vehicle-and-its accessed on 13th/05/2015

[46] http://searchsecurity.techtarget.com/definition/botnet

[47] "5 Common Cyber Attacks in the IoT - Threat Alert on a Grand Scale" (2016),

[48] STEPHEN COBB, (2016), "10 things to know about the October 21 IoTDDoS attacks" Available at https://www.welivesecurity.com/2016/10/24/10-things-knowoctober-21-iot-ddos-attacks/ accessed on 15th/05/2017

[49] SANS Institute 2001, Intrusion Detection Systems: Definition, Need and Challenges, available on https://www.sans.org/readingroom/whitepapers/detection/intrusion-detection-systemsdefinition-challenges-343

[50] Yulong Fu, Zheng Yan, Jin Cao, Ousmane Koné and Xuefei, (2017), "An Automata Based Intrusion Detection Method for Internet of Things" available at https://www.hindawi.com/journals/misy/2017/1750637/ accessed on 13th/05/2017

[51] Elike Hodo, Xavier Bellekens, Andrew Hamilton, PierreLouis Dubouilh, Ephraim Iorkyase, Christos Tachtatzis and Robert Atkinson, "Threat analysis of IoT networks Using Artificial Neural Network Intrusion Detection System", available at https://arxiv.org/ftp/arxiv/papers/1704/1704.02286.pdf accessed on 15th/05/2017

[52] Gemalto solutions, (2017), "Securing the Internet of Things (IoT)", available at https://safenet.gemalto.com/dataprotection/securing-internet-of-things-iot/ accessed on 2nd/05/2017

[53] "The 10 Biggest IoT Security Vulnerabilities Brian Buntz", (2017), available at http://www.ioti.com/security/10-biggestiot-security-vulnerabilities accessed on 5th/05/2017

[54] "Design of Complex Event-Processing IDS in Internet of Things" available at http://ieeexplore.ieee.org/document/6802673/ accessed on 10th/05/2017 [24] JAI Vijayan, (2016), "7 Imminent IoT Threats" available at http://www.darkreading.com/endpoint/7-imminent-iotthreats/d/d-id/1327233accessed on 12th/05/2017

[55] http://www.darkreading.com/endpoint/7-imminent-iotthreats/d/d-id/1327233?image_number=4 ACCESSED ON 14TH/05/2017

[56] http://www.ioti.com/security/10-biggest-iot-securityvulnerabilities/gallery?slide=1 ACCESSED ON 15TH/05/2017

[57] http://www.ioti.com/security/10-biggest-iot-security vulnerabilities /gallery?slide=2 ACCESSED ON 13TH/05/2017