International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

ELSEVIER SSRN

IJIEMR Transactions, online available on 5th Dec 2022. Link :http://www.ijiemr.org/downloads.php?vol=Volume-11&issue=Issue 12

DOI: 10.48047/IJIEMR/V11/ISSUE 12/02

Title **Identifying the Cyber Threats of IoT system in Edge Computing**

Volume 11, ISSUE 12, Pages: 7-14

Paper Authors

Byrapuram Sai Pravallika

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Identifying the Cyber Threats of IoT system in Edge Computing

[1]Byrapuram Sai Pravallika
[1]UG, Dept. of Computer science Engineering, Mallareddy Engineering College for Women.

Abstract: Internet of Things (IoT), connecting the things like sensors, compute machines etc for exchanging the information or perform operations via internet to achieve the quality of life. For building the smart cities, need to build the more IoT applications should deploy in to the cities and serves the citizens for their healthy and comfortable lives. For building the smart cities, deployment of the IoT applications is also require for the administration departments and for citizens security. In IoT applications, information exchange process among the different devices and networks is very common. These exchanging protocols includes may have sensitive information which should not leak. To prevent the cyber attacks in the cloud layers is a very popular topic but detection and prevention of the cyber attacks in the cloud layer is a risky process. In traditional IoT systems, fog layers are responsible to compute pre-process area of the cloud but it cannot manage complicates process. Edge Computing can provide a great assistance for loT devices to accomplish complicated tasks in an efficient way; on the other hand, its hasty development leads to the neglection of security threats to a large extent in edge computing platforms and their enabled applications. For prevention of cyber software attacks on IoT system require a new IoT architecture, especially updated Edge Computing layer in IoT architecture. In this paper, after research on literature of the IoT systems which prevents cyber attacks and its disadvantages, propose a system called Edge Malware Defense System (EMDS), it prevents cyber software attacks with help of machine learning algorithms. In our system we develop the system which dynamically analyzes security attacks from every input data comes from the input layers (Sensor's layers). We classify and compare the Malware API calls dataset with various machine learning algorithms of K-Nearest Neighbors (KNN), Random Forest, Support Vector Machine (SVM), Naïve Bayees and Artificial Neural Network algorithms.
Keywords: Edge Computing, Data Security, Machine learning, Cyber security, Fog Layer.

1. Introduction

Smart cities use IoT devices to improve cities' infrastructure, citizens utilities and their services etc by using different types of hardware and software like sensors, servers, clouds etc of IoT systems. These IoT devices also used to convert the complicated infrastructure into portable simple and digital first simpler lifestyle. According to [1], the usage of IoT devices is rapidly increasing from the past decade, five billions of IoT devices connected to the internet in 2013 expected to reach 50 billion of IoT devices will install upto 2020. There are several IoT devices which serve the citizens also it helps to reduce the human efforts and power consumption etc. For example Weather Monitoring IoT system will helps to maintain the temperature as per weather. This will helps to maintain the cooling systems of parks, transport stations etc. In this IoT system uses thermostat, anemometer and humidity sensor etc for calculation of weather data.

Generally Traditional IoT architecture consists of four layers. Those are 1) Sensors and Actuator layers, 2) Application Layer, 3) Fog Layer, and 4) Networking layer.

Sensors and Actuator layers

This layer consists of sensors and controllers for sensing the data based on hardware. These Sensors will works like input layers and based on the input data, actuators will alter the physical situations. For example in Automatic Water Leveling system [7],

soil moisture sensors identifies unwanted water level, this information will forward to fog layer using network layer and from Fog layer got command of switch off of water pump to actuator layer. In actuators layer the pump will switch off based on the Fog layer command.

Networking Layer

In this layer computes the connectivity and edge computing between Fog layer and Sensor and actuators layers. It uses many protocols to send and receive information.

Application Layer

Based on the sensors input data and computation of input data in Fog layer, computation data should be represent end users using graphical user interface. This layer is responsible for represent the data of IoT system using applications like mobile apps, web based application etc.

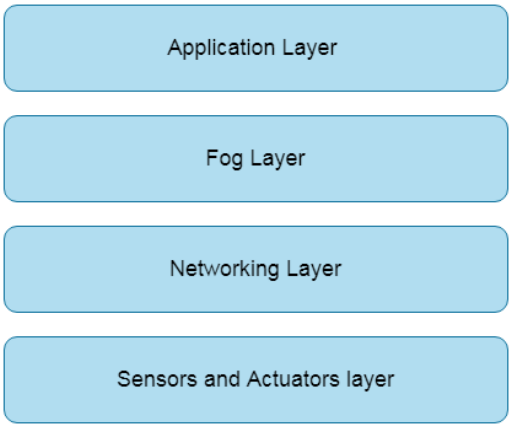

Fig 1: Standard IoT architecture

Fog Layer

For layer is a distributed layer, it is process the data in the local server before storing to the database servers or cloud servers. Based on the requirement it processes the data, techniques to make the decision on the situations. In simply, it is a decision layer but can handle minimum computation and it is not scalable.

These limitations of Fog layer, we can emerge the edge computing. This technology that is believed to be able to cope with the demands of the ever-growing IoT. The basic idea of edge computing is to employ a hierarchy of edge servers with increasing computation capabilities to handle mobile and heterogeneous computation tasks offloaded by the low-end IoT devices, namely, edge devices.

Edge computing pushes the intelligence, processing power, and communication capabilities of an edge gateway or appliance directly into devices like PLCs (programmable logic controllers), PACs (programmable automation controllers), and especially EPICs (edge programmable industrial controllers).

Many applications of IoT usages are growing up in many cities. Many cities motivated to deploy the sensors and actuators that carry the expectations of peoples of the cities. Now we discuss about few categories of IoT systems for building the smart cities. 1) Health: These categories of devices will use to monitor the patients health data 24 x 7 by using wearables. 2) Traffic monitoring: These devices help to monitor the traffic of the vehicles in cities dynamically. 3) Fleet management: These systems will helps to effective connectivity between the fleet vehicles. 4) Smart grids: These are the energy saving systems for reduce the energy of the expenditure of users by analyzing the consumption patterns of the users. 5) Wearables: fitness bands, virtual glasses etc are best examples of waerables. These systems will helps to analyze the individual information in a smart way.

In IoT systems as input data collect data from sensors and many other sources. Sometimes attacker may compromise sensors and deliver the malicious data through sensors or send malicious data through other sources. In software cyber attacks most of the cases attackers attacks the systems by sending some malicious code and data to the servers. These data and code consists of malicious API calls of read, write, update etc operations to steal or damage the system data. IoT systems contains the citizens personal data and context data like habits, user

preferences, health data etc, also it contains other sensitive data which should not leak to others. Based these reasons we need to build an architecture which can handling the software cyber attacks.

Main contributions of this paper are as follows:

- In this paper, a ML-based malware detection architecture called Edge Malware Defense System (EMDS) has been proposed.
- For developing this technique, we setup the dynamic analysis environment and run malware samples using the classification algorithms.
- Various behavior artifacts like API keys, API calls, registry changes, file operations, attacks types, etc. was extracted for malware detection in IoT system.

2. Literature Survey

Hui Suo et al. [3] described security issues in IoT framework, authors described on architecture, features and requirements of the security things in IoT levels. In the level of networks and perception various issues like authentication, DDoS, key agreement etc described.

Chen Qiang et al. [4] described security issues in IoT like RFID tag security, information security etc. In this reviewed various researches on the network security were proposed for RFID tags. Here identified that RFID can be forged. So in this research focused on data privacy protection, location privacy protection. identity privacy protection and information processing protection.

Kai Zhao et al. [5] reviewed different security issues of IoT that belongs to three-layered architecture and also proposed solution for problems of the security in each layer. Here attacks like capturing node, false node, denial of service, malicious data, timing attack, replay attacks in perception layer are discussed and elaborate to solve these attacks proposed cryptographic algorithms and key management techniques. In proposed techniques mainly used WPK1, PK1, and key management system. Here achieved problems like data access permission, authentication of identity, the privacy of the data and vulnerabilities of the software, etc.

Omar Said et al. [6] discussed the research challenges and open problems related to the Internet of things. The concept of IoT database was introduced and IoT database architecture was suggested. The six layers namely the IoT layer, data collection layer, data warehousing layer, event processing layer, data mining service layer, and application layer of IoT database model and their functions were discussed and demonstrated. The future vision of IoT was also discussed. The two IoT architectures viz. three-layer architecture, five-layer architecture, and other special purposes architecture were presented. Numerous challenges and open problems in IoT were discussed.

In these surveys, we can observe that previous surveys are depending on security-related issues and network-based but software cyber attacks are also very harmful. Attackers inject malicious data into the servers for gaining unauthorized access. To prevent these types of attacks in this paper we are proposing an architecture using advanced ML and DL technologies.

3. System Model

In this section we discuss about proposed architecture model and its methodology. In our proposed architecture, we have four layers those are sensor's layer, network layer, edge layer and cloud layer. Let's discuss these four layers and proposed methodology in the following.

Overview

Smart City builds by using advanced IoT technologies and its tools for providing services to the users in information exchange, smart medical services, communications etc. These IoT technologies are also used for analytics for business prediction, prediction of nature crisis etc.

Why IoT system should secure?

While building smart cities with advanced IoT technologies, we not only focus on the services of

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

the end users, we should also focus on the security of the system data. Generally attacks in the IoT systems are two types, network and software attacks. In the network attacks, attackers target the network layer of the IoT architecture. This process can be conducted remotely by the attacker. For example DDoS attack is very famous attack in this category. But in the software attacks, attackers target on the Sensors layer of the IoT architecture, they forward the malicious code or data to the IoT system for attack the system. Malicious attack is best example for software attacks. In our system we consider on the software attack detection which is held on Edge layer.

EMDS Architecture

In our proposed architecture, we have four layers. Those are,

1. Sensor's layer
2. Network layer
3. Local Server
4. Edge layer
5. Cloud layer

Sensor's Layer

This layer consists of different type's sensors and controllers based on the device which we are using. These Sensors will works like input layers and based on the input data, these input data may me malicious. Forwarding the malicious data is depending on the different types of situation. Most of the time attacker's will compromise the sensors and forward the malicious content to system through network layer.

Networking Layer

This layer is bridge between Fog layer and Sensor layer. By using internet protocols the data forward to Fog layer from the sensor's layer.

Edge Layer

Edge layer is a decentralized architecture, its process the data in the local server before storing to the cloud servers. Here we apply the machine learning

Local Server

For training of the of the data using the machine learning algorithms like Artificial Neural Network algorithm takes heavy computation but for prediction of data takes low computation. After classification we can store the machine learning object data in a file, namely, model file. To reduce the computation in the Edge layer, in our system we can train dataset and generate the model file in the local server and then we can shift the best accuracy model file in to the edge layer.

For Training the dataset with machine learning algorithms we used Malware API Call Sequence dataset [8], after training the dataset we test the algorithms with 30% of dataset with same algorithms in the local servers. Based on the accuracy scores we deploy the classification algorithm in to the fog layer for prediction input data.

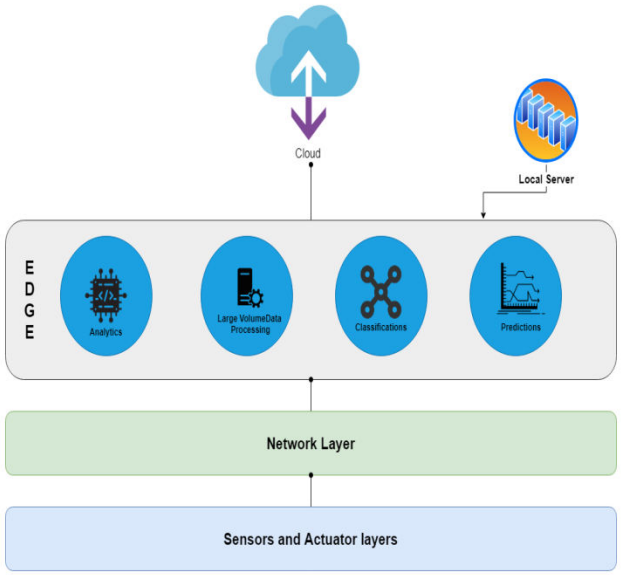

Fig 2: IoT Architecture with Edge Computing

classification algorithm to predict the malicious content. In the following Edge layer Procedure diagram we clearly mention the procedure of detection system of malicious data. Based on the requirement it applies the algorithms, techniques to make the decision on the situations.

Cloud Layer

This is top level management of IoT data. It contains cloud services which acts as a remote servers. Due to advantages of cloud computing we store the data of IoT devices to the cloud.

4. Methodology

4.1 Problem Statement

Various smart city IoT sensors connect to the IoT system for providing inputs for computation and storage into the cloud. These sensors data collect in the fog layer through the network layer, from the network layers there are more chance of unauthorized access to manipulate the data from the outsiders, but to prevent these types of attacks we have lot of surveys, but main problem is that, by compromised IoT sensor's input data maly contain malicious API calls to read the data, update, delete data etc, we can't verify these cyber attacks with network security concepts. We need architecture for dynamic analysis of each data collecting from the sensors before storing in to the cloud.

4.2 Proposed Methodology

Edge computing is recently designed between cloud and loT layers, in our work we designed edge layer to verify the outgoing data to the cloud. Recent surveys provide solutions for preventing the attacks in network based using traditional concepts like DDoS, Encryption techniques etc. But in our proposed work ADS, we verify and analysis of the each and every income data in the edge layer with machine learning algorithms of data which is coming from the sensors layers. The detection in the edge layer would be more significant to automatically alert the administrators or users, when quickly and effectively detecting compromised loT devices in the IoT layer and interrupt the connection of the IoT attacks from the network of urban life.

Our proposed model is shown in Figure 3; the model tracks the input data that goes through each edge node. Since edge nodes are closest to IoT sensors, they will be more effective at identifying the cyber-attacks at edge nodes instead of the cloud centers. In edge layer there we set up a classification model to identify the malicious content coming from the input layers. This classification model we setup with one machine learning algorithm which is suitable for detection of malicious calls from the multiple ML algorithms by applying training and testing methods. After performing the training and testing the malicious API dataset we can find the one classification algorithm to predict the malicious content which is coming from the IoT layers. This process we deployed in the edge layer because of its intermediate between IoT layer and Cloud. In our architecture for each and every input data edge layer will classify with ML algorithm and if any malicious found then it will prevent to stop the storage in to the cloud.

In Figure 3, we can see every data which is coming from the IoT layers are verified in the Edge layer, here edge servers verifies the incoming data for detection of malicious data with machine learning model. But for training the dynamical malicious labeled dataset takes heavy computation. By using Edge processors like Intel Xeon etc we can predict the data using the model files. Model files are selected and created in the local servers by training and testing process.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

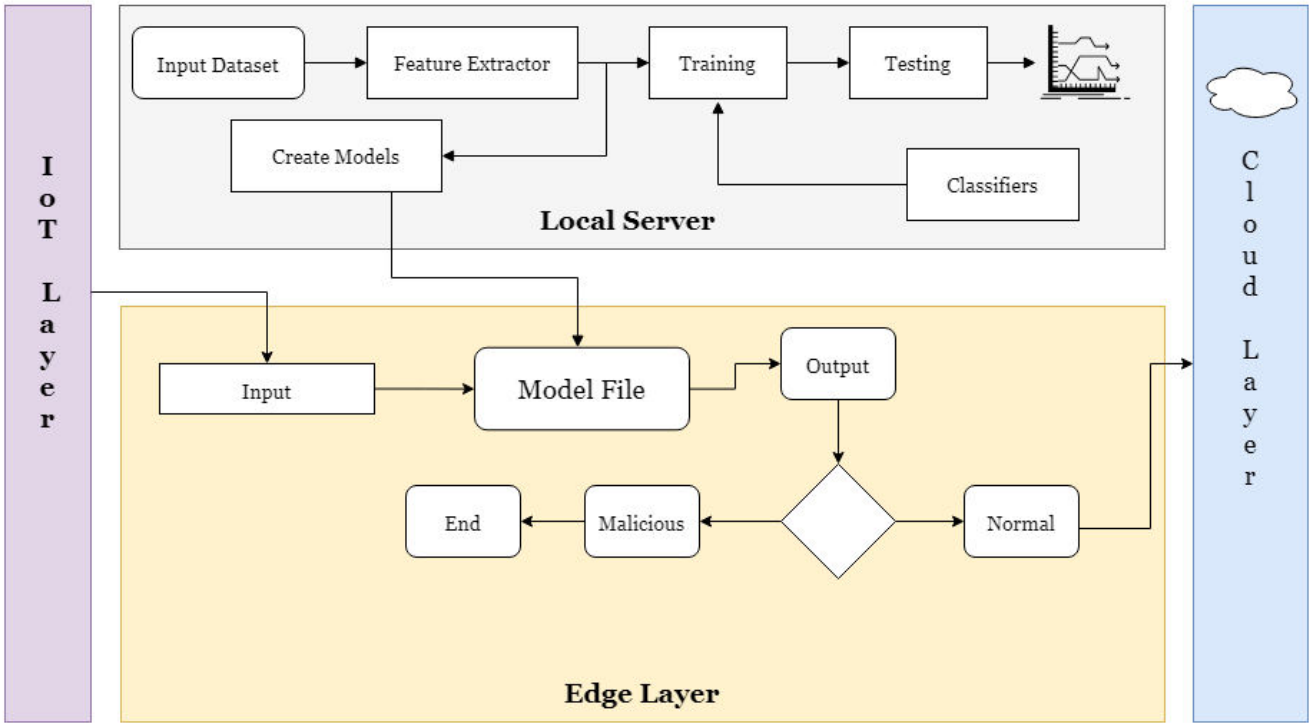

Fig 3. Proposed model of EMDS

We classify and compare the malware API calls dataset with various machine learning algorithms of K-Nearest Neighbors (KNN), Random Forest, Support Vector Machine (SVM), Naïve Bayees, Decision Tree and ANN algorithms.

Description of Dataset

This malware API calls dataset provided by the Hacking and Countermeasure Research Lab (HCRL), it consists of API call sequences, API key and malware's class (analyzed by Kaspersky AntiVirus).

5. Results

In this chapter will calculate the performance measures of algorithms. For performance measure calculation of three algorithms taken four types of measures, those are accuracy, precision, recall and F1-score. For calculating all these scores we require testing dataset having original classes of attributes. Then only we can compare the original results with predicted results. For calculating scores of Accuracy, Precision, Recall and F1-score we need to find the answers of 4 terms those are True Positive, True Negative, False Positive, and False Negative. We describe clearly about these four terms in the following.

True Positive (TP): In test dataset input as normal input then algorithm also predict output as normal then it's True Positive.

False Positive (FP): In test dataset input as normal input then algorithm predict output as malicious then it's False Positive.

False Negative (FN): In test dataset input as malicious input then algorithm predict output as normal then it's False Negative.

True Negative (TP): In test dataset input as malicious input then algorithm also predict output as malicious then it's True Negative.

For calculation performance measures of Accuracy, Precision, Recall and F1-score these four above metrics is enough.

Let see the each performance measure Accuracy, Precision, Recall and F1-score process clearly.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Accuracy: Accuracy of our algorithms is calculated by number of correctly predicted normal inputs and malicious inputs divide by total number of dataset.

$$\text{Accuracy} = \frac{\text{Correctly predicted Normal Inputs (TP)} + \text{Correctly predicted Malicious Inputs}}{\text{Total inputs (TP + TN + FP + FN)}}$$

Precision: Precision of our algorithms is calculated by Total number of malicious inputs correctly identified by total number malicious inputs identified.

$$\text{Precision(P)} = \frac{\text{Correctly predicted Malicious Inputs (FP)}}{\text{Total number malicious inputs identified (FP + FN)}}$$

Recall: Precision of our algorithms is calculated by Total number of malicious inputs correctly identified by total number malicious inputs identified.

$$\text{Recall(R)} = \frac{\text{Correctly predicted Normal Inputs (TP)}}{\text{Total number malicious and normal inputs identified (TP + TN)}}$$

F1-Score: F1 score measure is depends on precision and recall scores, it's mean of precision and recall.

$$\text{F1} - \text{score(R)} = 2 \text{ x } \frac{\text{P X R}}{\text{P + R}}$$

After testing of the algorithms, we apply performance measure scores to the algorithms individually based on the testing results. In Table 1 described performance measures of the Accuracy, Precision, Recall and F1-score for each algorithm in a single table.

Algorithms	Accuracy	Precision	Recall	F1Score
Naïve Bayes	0.31	0.31	0.31	0.32
SVM	0.66	0.64	0.664	0.66
K-Nearest Neighbors	0.39	0.41	0.41	0.42
Random Forest	0.79	0.77	0.78	0.77
Decision Tree	0.83	0.801	0.81	0.82
ANN	0.91	0.89	0.88	0.88

Table 1. Performance Measures

In Figure 4, represents the Accuracy scores of the five algorithms in graph. In Figure 5, represents the precision scores of the five algorithms in graph. In Figure 6, represents the recall scores of the five algorithms in graph. In Figure 7, represents the F1 scores of the five algorithms in graph.

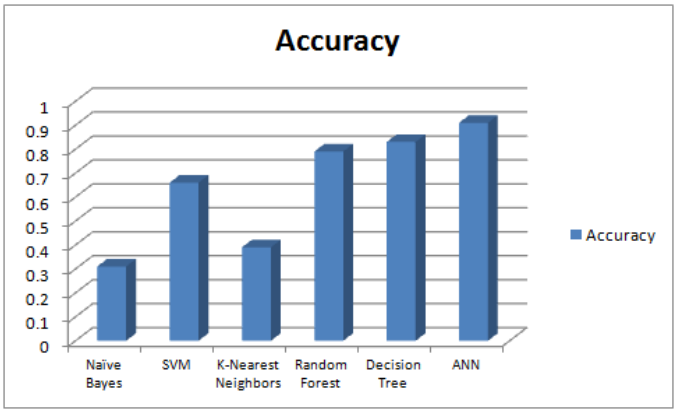

Fig 4. Accuracy Score

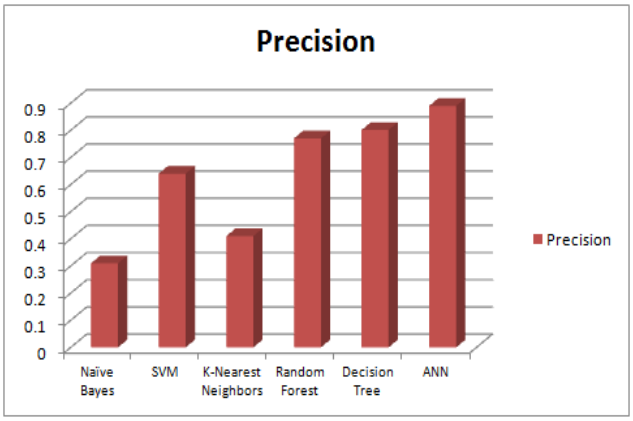

Fig 5. Precision Score

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

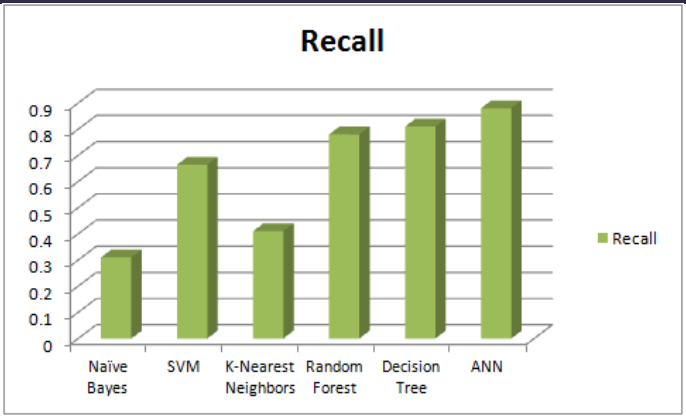

Fig 6. Recall Score

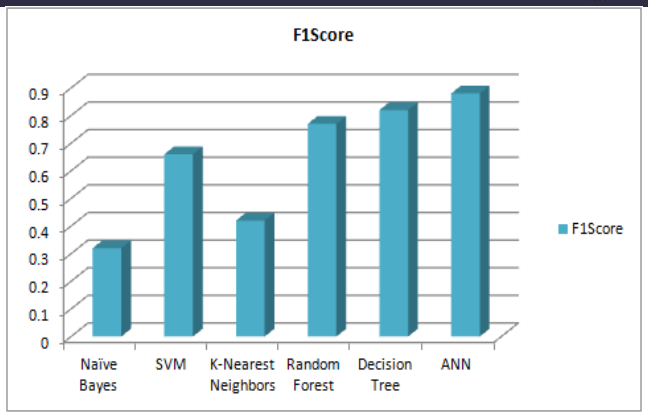

Fig 7. F1score

6. Conclusions

In this paper, we understood importance of the Edge Computing and proposed a methodology to prevent the software attacks in IoT architecture using the Edge Comptuting. Our experiments with the most recent IoT attack database show that our ensemble approach, especially stacking, performs better than survey models in identifying attacks. The proposed EMDS can significantly detect malicious behavior using anomalies based on deep learning through the evaluation of the HCRL dataset to detect the binary labeled classification before distributing on fog nodes. By comparing ANN classifier with ML algorithms we got the best results to the ANN algorithm in terms of accuracy, precision, recall, and f1-score.

References

[1] J. Chase, ”The Evolution of the Internet of Things”, Strategic marketing, Texas Instruments, Dallas, 2013.

[2] P. Vlacheas et al., ”Enabling smart cities through a cognitive management framework for the internet of things,” in IEEE Communications Magazine, vol. 51, no. 6, pp. 102-111, June 2013.

[3] Hui Suoa, Jiafu Wana and Caifeng Zoua, Jianqi Liua, “Security in the Internet of Things: A Review”, International Conference on Computer Science and Electronics Engineering, 2012, pp. 649-651.

[4] Chen Qiang, Guang-ri Quan, Bai Yu and Liu Yang, “Research on Security Issues on the Internet of Things”, International Journal of Future Generation Communication and Networking, 2013, pp.1-9.

[5] Kai Zhao and Lina Ge, “A Survey on the Internet of Things Security”, IEEE, International Conference on Computational Intelligence and Security, 2013, pp. 663-667.

[6] Omar Said and Mehedi Masud, “Towards Internet of Things: Survey and Future Vision”, International Journal of Computer Networks (IJCN), Vol.1, Iss.1, 2013, pp. 1-17.

[7] B. N. Getu and H. A. Attia, "Automatic water level sensor and controller system," 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA), Ras Al Khaimah, 2016, pp. 1-4, doi: 10.1109/ICEDSA.2016.7818550.

[8] Ki, Youngjoon & Kim, Eunjin & Kim, Huy Kang. (2015). A Novel Approach to Detect Malware Based on API Call Sequence Analysis. International Journal of Distributed Sensor Networks. 2015. 1-9. 10.1155/2015/659