JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

AN ADAPTIVE INTRUSION DETECTION SYSTEM FOR IoT NETWORKS

[1]Dr.P.Elavarasan, [2]Mr.A.Alexander,
[1]Associate Professor, [2]Assistant Professor
Department of Electronics and Communication Engineering,
Rajiv Gandhi College of Engineering and Technology, Puducherry, India

Abstract: The number of diverse interconnected Internet of Things (IoT) devices keeps increasing exponentially, introducing new security and privacy challenges. These devices tend to become more pervasive than mobile phones and already have access to very sensitive personal information such as usernames, passwords, etc., making them a target for cyber-attacks. Given that smart devices are vulnerable to a variety of attacks, they can be considered to be the weakest link for breaking into a secure infrastructure. For instance, IoT devices have recently been employed as part of botnets, such as Mirai, and have launched several of the largest Distributed Denial of Service (DDoS) and spam attacks in history. Over the past few year the development in mobile industry and development of internet, network for all, 4G, 5G etc. enable the ordinary people to elite peoples depends on mobile networks for regular business developments, entertainment, medical and educational needs. It is mandatory to provide the high level of security and dual privacy protection to the users sharing the large set of information through the cloud. Intrusion Detection System (IDS) defined as a Device or software application which monitors the network or system activities and finds if there is any malicious activity occurs. Outstanding growth and usage of internet raises concerns about how to communicate and protect the digital information safely. In today's world hackers use different types of attacks for getting the valuable information. Many of the intrusion detection techniques, methods and algorithms help to detect those several attacks.

Index Terms **- IoT, DDoS, IDS.**

I. INTRODUCTION

Protecting the network from miscellaneous activity is mandatory and that will save a huge data to be hacked in suspicious activity. The important part of any system management is to protect it from network hazards and ensuring high level of security. The large amount of confidential data, transaction data, activities and follow ups are uploaded to the network in current days through simple steps. Hence it is become flexible for end users to upload the data more frequently and hassle free steps. Every time the user login to the particular network enable the port open to accept all the inputs for a certain period of time frame. This key gap is enough for the hackers and third part users to enter and grab the most privileged information for the network grid. Intrusion detection systems are small tools or software that acts as a ingress guard in the network points to ignore the miscellaneous activity during heavy traffic. The IDSS (intrusion detection system) provides dual stack security, ensure the authenticated entry to protect the network from Internet attacks. Firewall provides basic security to the system, to protect the grid from third party attacks. The design of IDSS depends on the type of information, quality and weightage of the system. The system follows certain rules and policies to find out and troubleshoot such online threats, malware attacks or any kind of intrusions to safe guard both the edge computing devices and data.

1.1 Classification of Intrusion Detection System

NIDSS: Network Intrusion Detection: The NIDSS models are intrusion detection systems that are pre programmed to be initiated at certain node of the internet to monitor and track all internet traffic that is come across the way on all subnets. In case of any miscellaneous activity happen in the network, an immediate alert will be sent to the administrator. A common example we can discuss that firewall that is being installed in all systems, where an anonymous software trying to crack the protection[1-3].

Network intrusion detection systems (NIDS) are placed at a strategic point or points within the network to monitor traffic to and from all devices on the network. It performs an analysis of passing traffic on the entire subnet, and matches the traffic that is passed on the subnets to the library of known attacks. Once an attack is identified, or abnormal behaviour is sensed, the alert can be sent to the administrator. An example of an NIDS would be installing it on the subnet where firewalls are located in order to see if someone is trying to break into the firewall. Ideally one would scan all inbound and outbound traffic, however doing so might create a bottleneck that would impair the overall speed of the network.

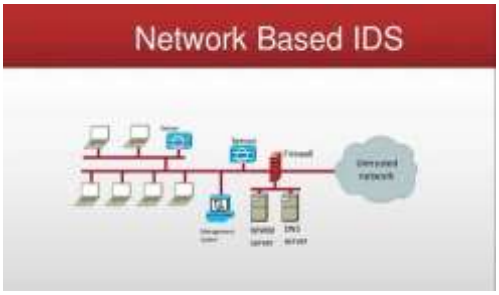

Fig.1.1 Network based IDS

NIDS can be also combined with other technologies to increase detection and prediction rates. Artificial Neural Network based IDS are capable of analysing huge volumes of data, in a smart way, due to the self-organizing structure that allows INS IDS to more efficiently recognize intrusion patterns as shown Fig.1.1. Neural networks assist IDS in predicting attacks by learning from mistakes; INN IDS help develop an early warning system, based on two layers. The first layer accepts single values, while the second layer takes the first's layers output as input; the cycle repeats and allows the system to automatically recognize new unforeseen patterns in the network. This system can average 99.9% detection and classification rate, based on research results of 24 network attacks, divided in four categories: DOS, Probe, Remote-to-Local, and user-to-root.

Network Intrusion Cover-Up Methods: Once attackers have employed common network intrusion attack techniques, they'll often incorporate additional measures to cover their tracks and avoid detection. As mentioned above, using non-malware and living off the land tools have the dual advantage of being powerful while blending into business justified usage, thus making them hard to detect. In addition, below are three practices that are frequently used to circumvent cybersecurity teams and network intrusion detection systems:

- **Deleting logs:** By deleting access logs, attackers can make it nearly impossible to determine where and what they've accessed (that is, without enlisting the help of an extensive cyber forensics team). Regularly scheduled log reviews and centralized logging can help combat this problem by preventing attackers from tampering with any type and/or location of logs.
- **Using encryption on departing data:** Encrypting the data that's being stolen from an organization's network environment (or simply cloaking any outbound traffic so it looks normal) is one of the most straightforward tactics attackers can leverage to hide their movements from network-based detections.
- **Installing rootkits:** Rootkits, or software that enables unauthorized users to gain control of a network without ever being detected, are particularly effective in covering attackers' tracks, as they allow attackers to leisurely inspect systems and exploit them over long periods of time.

Intrusion Detection and Response Challenges: Network intrusion detection and response systems have come a long way over the years. As digital networks become more and more complex, however, such products can sometimes fall flat. For example, even though non-malware is an increasingly common attack vector, traditional network intrusion, detection, and response solutions struggle to uncover these attacks and still focus primarily on malware. Similarly, despite cloud-based applications becoming an increasingly popular entry point for attackers, traditional network intrusion detection and response systems aren't designed to support such threats.

Host Intrusion Detection: The meaning of Host here any network connected system or device on IoT. The Host-IDSS able to run independently in host systems, that monitors the network activities, keep track the incoming and outgoing packets and alert the admin control in case of any miscellaneous activity held in the network grid. In some systems, the existing machine controls getting changed within due to internet malfunctioning software entering into the network. The anonymous software enable the machine to get updated automatically and change the control line etc. are example of host based intrusion detection systems in practice.

Protocol-based intrusion detection system (**PIDS**) is an intrusion detection system which is typically installed on a web server, and is used in the monitoring and analysis of the protocol in use by the computing system. A PIDS will monitor the dynamic behaviour and state of the protocol and will typically consist of a system or agent that would typically sit at the front end of a server, monitoring and analysing the communication between a connected device and the system it is protecting.

Hybrid Intrusion Detection System: In some cases, the IDSS modules are configured as one or more hybrid combination of intrusion detection systems. Study reveals that hybrid intrusion detection system is more robust in security comparing all other methods. The method is tending to be more secure because of the stacked security in more than one method that configured together.

1.2. Design Methodologies

Signature Based IDSS: The concept of signature based IDSS implementation approach that it monitors the specific frame of patterns to be followed in the network in common. The SIDSS also being implemented with the help of previously handled patterns or the miscellaneous activities related steps and instructions to be repeated again. In such cases, pattern recognition models, artificial neural networks and machine learning approaches [4]. The SIDSS is quite similar like malware detection in massive mobile networks or IOT networks.

A signature-based NIDS examines ongoing traffic, activity, transactions, or behaviour for matches with known patterns of events specific to known attacks. As with antivirus software, a signature- based NIDS requires access to a current database of attack signatures and some way to actively compare and match current behaviour against a large collection of signatures.

Signature-based detection system (also called misuse based), this type of detection is very effective against known attacks. It implies that misuse detection requires specific knowledge of given intrusive behaviour. An example of Signature-based Intrusion Detection System is SNORT.

Anomaly Based IDSS:AIDSS methods are enabled in many systems to provide protection grid that detect the unknown malwares as well as keep track on new malwares developing inside the network. In most of the current scenarios the anomaly based detection of intrusion elements is developed using suspicious detecting machine learning models. In case of any new suspicious activity held in the network the pre-trained machine learning models detect and probe the root cause of the problems.

An anomaly-based intrusion detection system is an intrusion detection system for detecting both network and computer intrusions and misuse by monitoring system activity and classifying it as either normal or anomalous. The classification is based on heuristics or rules, rather than patterns or signatures, and attempts to detect any type of misuse that falls out of normal system operation. This is as opposed to signature-based systems, which can only detect attacks for which a signature has previously been created [5].

In order to positively identify attack traffic, the system must be taught to recognize normal system activity. The two phases of a majority of anomaly detection systems consist of the training phase (where a profile of normal behaviours is built) and testing phase (where current traffic is compared with the profile created in the training phase). Anomalies are detected in several ways, most often with artificial intelligence type techniques. Systems using artificial neural networks have been used to great effect. Another method is to define what normal usage of the system comprises using a strict mathematical model, and flag any deviation from this as an attack. This is known as strict anomaly detection. Other techniques used to detect anomalies include data mining methods, grammar based methods, and Artificial Immune System [6].

Network-based anomalous intrusion detection systems often provide a second line of defence to detect anomalous traffic at the physical and network layers after it has passed through a firewall or other security appliance on the border of a network. Host-based anomalous intrusion detection systems are one of the last layers of defence and reside on computer end points. They allow for fine-tuned, granular protection of end points at the application level.

1.3 Types of Attacks

Dos Attack: A denial-of-service (DoS) attack is a type of cyber attack in which a malicious actor aims to render a computer or other device unavailable to its intended users by interrupting the device's normal functioning. DoS attacks typically function by overwhelming or flooding a targeted machine with requests until normal traffic is unable to be processed, resulting in denial-of-service to addition users. A DoS attack is characterized by using a single computer to launch the attack.For example, Black Friday sales, when thousands of users are clamouring for a bargain, often cause a denial of service. But they can also be malicious. In this case, an attacker purposefully tries to exhaust the site's resources, denying legitimate users access.A distributed denial-of-service (DDoS) attack is a type of DoS attack that comes from many distributed sources, such as a botnet DDoS attack.

Probe Attack: Probing attacks are an invasive method for bypassing security measures by observing the physical silicon implementation of a chip. As an invasive attack, one directly accesses the internal wires and connections of a targeted device and extracts sensitive information.

Remote to Control Attack: A remote attack is a malicious action that targets one or a network of computers. The remote attack does not affect the computer the attacker is using. Instead, the attacker will find vulnerable points in a computer or network's security software to access the machine or system. The main reasons for remote attacks are to view or steal data illegally, introduce viruses or other malicious software to another computer or network or system, and cause damage to the targeted computer or network.

User to Root Attack: Initially attacker access normal user account, later gain access to the root by exploiting the vulnerabilities of the system. Examples: Perl, Load Module and Eject attacks.

1.4. Detection Models

It is required to implement a smart detection engine that has strong analyzing capacity with respect to the input malfunctions. It has to treat the malwares as well as respond the attacks in time. The processing time and analysis epochs are need to be controlled programmatically.

Architecture Modelling: Standalone or collaborative architecture is choosing based on the crowd capability of the network devices. With respect to the edge router the architecture may be collaborative or distributive. In most of the literature suggest that collaborative approach suits for the increasing demands on IDSS in IoT networks.

Types of Attack: The basic process for IDS is that it passively collects data and preprocesses and classifies them. Statistical analysis can be done to determine whether the information falls outside normal activity, and if so, it is then matched against a knowledge base. If a match is found, an alert is sent.The IDSS implementation also considers the type of attack that frequently hit the edge computing devices. The analysis can be made through studying the pattern of highlighted attack history. The attacks would be anomaly attacks or signature based authentication attacks.

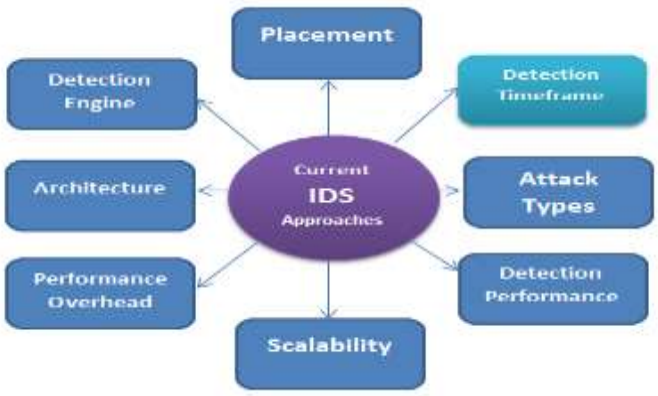

Fig. 1.2 Analysis of Current IDS approaches

An Intrusion Detection System (IDS) is a system that monitors network traffic for suspicious activity and issues alerts when such activity is discovered. It is a software application that scans a network or a system for the harmful activity or policy breaching.

Detection Performance: The detection performance of any installed IDSS modules will be determined by the success rate of accurate detection of malware attacks, prediction and prevention of such vulnerabilities in the IoT networks. The effectiveness of the attack prediction depends on the fundamental attributes of the IDS. The approach performance is measured through the False positive rate and True positive rate. The accuracy, precision and recall are the parameters that impact the performance measure of the IDSS implemented approaches.

Scalability: In terms of increasing demand on IDSS in all networks, in order to address the raising problem of all devices, significant scalability of IDSS systems is required. The future implementations are suggested to focus on improving the scalability factor of the IDSS in IoT networks. Massive IoT systems need adaptive and predictable IDSS modules in the network. Machine learning approach with incorporated in each IDSS modules enable the future IDSS platforms more robust undoubtedly.

Performance Overhead: Overhead is any combination of excess or indirect computation time, memory, bandwidth, or other resources that are required to perform a specific task. It is a special case of engineering overhead.There are three main types of intrusion detection software, or three main "parts," depending on if you view these all as part of one system: Network Intrusion Detection System. Network Node Intrusion Detection System.

Detection Engine: After the detection engine alerts on the rules, and after thresholding but before logging, there is one last step to go through: suppression. *Suppression* prevents rules from firing on a specific network segment without removing the rules from the rule set. By using suppression, you can quickly tune rule sets for a specific environment, without disabling rules that may be useful in general but that analysts have deemed acceptable when targeting specific IP addresses.

The detection engine is the primary Snort component. It has two major functions: rules parsing and signature detection. The detection engine builds attack signatures by parsing Snort rules. Snort rules are read line by line, and are loaded into an internal data structure. The rules are loaded only when the Snort service is started, meaning that to modify, add, or delete a rule you must refresh the Snort daemon.The detection engine runs traffic through the now loaded rule set in the order that it loads them into memory. You can dictate which rules are run first by prioritizing and then organizing in the manner you see fit. Rules are split into two functional sections: the rule header (rule tree node) and the rule option.

Detection Performance: Its main function is to send an alert immediately when it identifies any activity in the system. It recognizes various security incidents. Also helps to examine the quantity and types of such suspicious attacks.

It also detects bugs and issues relating to their network device configurations.

II. LITERATURE SURVEY

In [1] Year 2019, authors Keigo Ogawa; Kenji Kanai; Kenichi Nakamura ; Hidehiro Kanemitsu ; Jiro Katto ; Hidenori Nakazato, entitled Smart City IoT device virtualization, IEEE published 2019, stated that in their research work Smart city enabled IoT device virtualization, they concluded that dynamic asset scaling is considered as one of the factor. The fundamental test and confirmations are validated in their research work. They found that the application handing duration is need to be managed to evaluate the secure cloud.

In [2] Year 2019, authors named Eirini Anthi; Lowri Williams ; Małgorzata Słowińska; George Theodorakopoulos ; Pete Burnap, ORCA, journal published on Intrusion detection system for smart Home., the mentioned in the research work that three layer of protection framework is evaluated in the paper. The proposed model composed of three major categories of capacities. The first one is based on sorting of different group of applications organized with the single could and their nature. The second one is based on malware attacks, the third one is based on lack of security in the transmitted information, and the encryption schemes are evaluated. Their research was concluded in a way that they found out the lack of security facility in the devices connected in IoT.

III. EXISTING WORK

3.1 Detection of Data

Mobile ad hoc networks (MANETs) consist of mobile nodes that work independently without an infrastructure. They are useful in application areas like disaster management emergency and rescue operations where it is not possible to have well-defined infrastructure. MANETs are characterized by its great flexibility. However, MANET's inherent vulnerability increases their security risks. Though MANETs are dynamic and cooperative in nature, it needs efficient and effective security mechanisms to safeguard the mobile nodes. Intrusion detection and prevention are primary mechanisms to reduce possible intrusions. Intrusion detection using classification algorithms effectively discriminates "normal" behaviour from "abnormal" behaviour. Therefore, intrusion detection and prevention system can be used as a secondary mechanism of defence in any wireless environment and mobile databases so that it can be a part of the reliable communication in MANETs.

Intrusion detection systems (IDS) play a major role in providing security to networks. In this paper, we introduce a new intelligent agent-based intrusion detection model for securing the mobile ad hoc networks. The main function of the proposed intrusion detection system is to monitor the computer system and network in order to find the intrusion activities in the system. In such system, attacks are divided into two categories, namely, host-based attacks and network-based attacks. Hence, IDSs are also classified into two categories, namely, network intrusion detection system (NIDS) and host-based intrusion detection system (HIDS). Another way to classify IDSs is with respect to the way in which they detect intrusions. According to this classification, there are two categories of intrusions, namely, anomaly-based intrusion and misuse-based intrusion. Anomaly-based IDS is able to identify malicious traffic based on the deviations from the preestablished normal network traffic patterns. In misuse-based detection, network traffic is examined for preconfigured and predetermined attack patterns.

3.2 Existing System

In the existing system IDP-IoT is based on agent technology to support mobility, rigidness, and self-started attributes. Due to IoT limitations, the proposed solution is implemented in the middle, between IoT devices and the router that can be installed in a gateway. The existing IDP IoT is a hybrid solution as it is based on misuse and anomaly. The prevention agent instance sent to perform prevention on IoT devices in case of attack or intrusion to isolate the IoT from the protected network until it is cured.

Issues in Existing System: No alert when the intruder attack, Machine learning algorithm is not suitable for large data sets, It does not perform very well when the data set has more noise i.e. target classes are overlapping, In cases where the number of features for each data point exceeds the number of training data samples, it will underperform.

3.3 Agent Based Technology

Agent-based technology provides a new computing paradigm, where intelligent agents can be used to perform tasks such as sensing, planning, scheduling, reasoning and decision-making.

A agent-based intrusion detection model called Intelligent Agent based Feature Selected Hybrid Classifier (IAFSHC) for detecting the intruders in wireless ad hoc networks has been proposed for securing the networks. For this purpose, a combination

of an intelligent agent-based weighted outlier detection algorithm and an intelligent agent-based enhanced multiclass SVM algorithm for classification have been proposed in order to classify the attacks effectively.

3.4 Multi-Agent IDS Scheme

The use of MAS enables taking advantage of some of the properties of agents such as re-activity, pro-activity and sociability and makes the task of intrusion detection more robust, faster and easier since the tasks are shared between several agents in the system. However, generally previous works used signature-based technique with which the system does not enable to detect unknown or a variation of known attacks .In this, it used a deep learning algorithms which are known for their effectiveness in detecting unseen attacks. The proposed deep learning-based multi-agent for IDS is composed of four types of agents: Pre - processor agent, Reducer agent, Agent classifier, Decision-maker agent

Pre-processor Agent (PA): This agent can be extensible so it can be adapted to an online version, and then, it will have to collect the data end information about the traffic of the network. After preparing the data, PA sent it in a format accepted by a neural network to the agent reducer.

Reducer Agent (RA): Agent reducer executes the autoencoder algorithm to reduce the dimension of the data. The structure of our autoencoder is represented. Then, the reduced data are sent to the agents classifier to perform the classification task.

Agents Classifier (CA): The number of the agents classifiers can be more than 1 (from 1 to n) located on the same node of the network; each one builds its model; and once the model is built, it can be used for next predictions. Finally, each agent sends its result to the decision-maker agent. In experiments, two agents classifiers are built: K-NN agent and MLP agent.

Decision-Maker Agent (DA): DA asks periodically for the decisions of agents classifiers. Once they are received, DA analyses the situation: If one CA sent, its prediction for a given location; DA takes this prediction as a final decision. In the case of the existing of several CA in the same location, the DA compares the results: If they are identical, DA takes it as a final decision; if there is a difference, DA takes the prediction having the highest accuracy rate as a final decision.

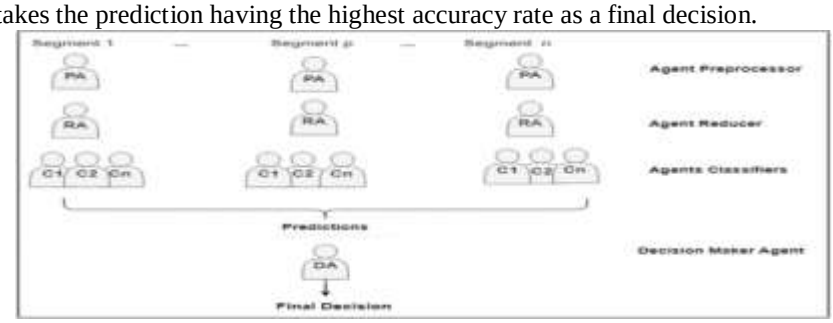

Fig. 3.1 Decision maker agent

IV. PROPOSED WORK

In the past few decades, Deep Learning has proved to be a very powerful tool because of its ability to handle large amounts of data. The interest to use hidden layers has surpassed traditional techniques, especially in pattern recognition. One of the most popular deep neural networks is Convolutional Neural Networks. Since the 1950s, the early days of AI, researchers have struggled to make a system that can understand visual data. In the following years, this field came to be known as Computer Vision. In 2012, computer vision took a quantum leap when a group of researchers from the University of Toronto developed an AI model that surpassed the best image recognition algorithms and that too by a large margin.

4.1 Background of CNNs

CNN's were first developed and used around the 1980s. The most that a CNN could do at that time was recognize handwritten digits. It was mostly used in the postal sectors to read zip codes, pin codes, etc. The important thing to remember about any deep learning model is that it requires a large amount of data to train and also requires a lot of computing resources. This was a major drawback for CNNs at that period and hence CNNs were only limited to the postal sectors and it failed to enter the world of machine learning.

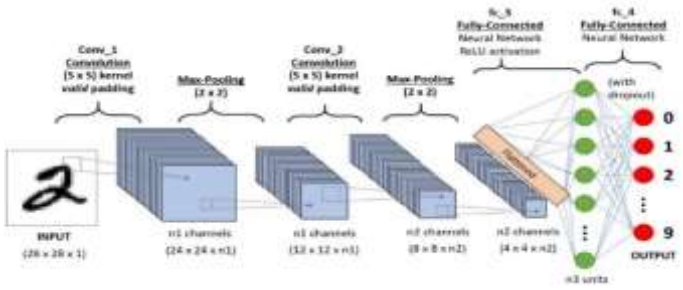

Fig.4.1 Convolutional Neural Network

It was time to bring back the branch of deep learning that uses multi-layered neural networks. The availability of large sets of data, to be more specific ImageNet datasets with millions of labeled images and an abundance of computing resources enabled researchers to revive CNNs.

In deep learning, a convolutional neural network (CNN/ConvNet) is a class of deep neural networks, most commonly applied to analyze visual imagery. Now when we think of a neural network we think about matrix multiplications but that is not the case with ConvNet. It uses a special technique called Convolution. Now in mathematics convolution is a mathematical operation on two functions that produces a third function that expresses how the shape of one is modified by the other.

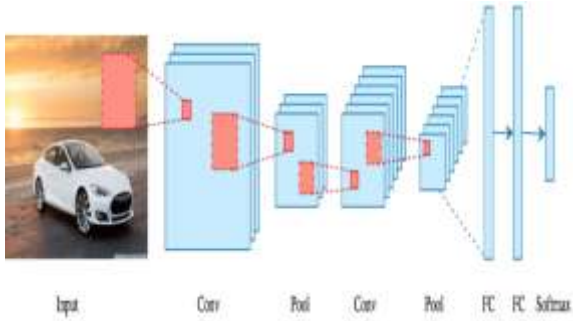

Fig 4.2 Process of CNN

Before we go to the working of CNN's let's cover the basics such as what is an image and how is it represented. An RGB image is nothing but a matrix of pixel values having three planes whereas a gray-scale image is the same but it has a single plane. Take a look at this image to understand more.

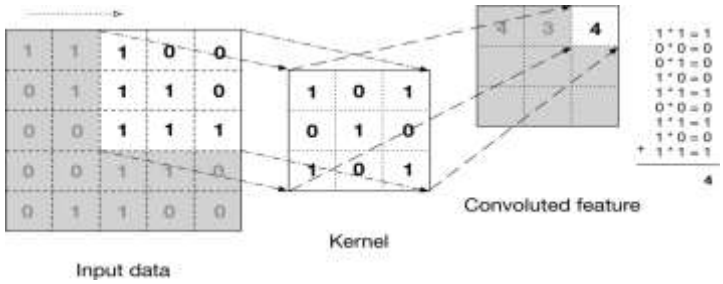

Fig 4.3 Method of Calculation

The above image shows what a convolution is. We take a filter/kernel (3×3 matrix) and apply it to the input image to get the convolved feature. This convolved feature is passed on to the next layer.

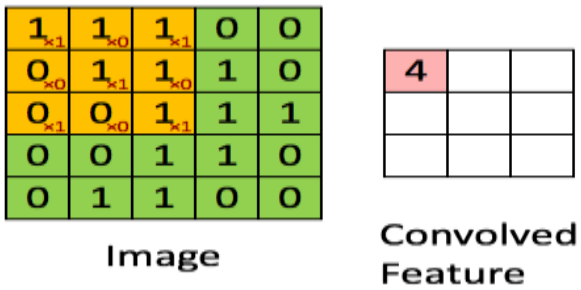

Fig 4.4 Table for CNN

Convolutional neural networks are composed of multiple layers of artificial neurons. Artificial neurons, a rough imitation of their biological counterparts, are mathematical functions that calculate the weighted sum of multiple inputs and output an activation value. When you input an image in a Conv-Net, each layer generates several activation functions that are passed on to the next layer.

The first layer usually extracts basic features such as horizontal or diagonal edges. This output is passed on to the next layer which detects more complex features such as corners or combinational edges. As we move deeper into the network it can identify even more complex features such as objects, faces, etc.

4.2 CLOUD BASED TECHNOLOGY

In the proposed research work cloud based advanced intrusion detection model is developed. The robust architecture provides the collection of number of possible attacks in the massive internet of things network. The collection of intrusion models we call as bags of attacks. The proposed machine learning algorithm creates an robust prediction system for detection of feasible intrusions in the IoT network, the vulnerability of the IoT attacks act as a key for detecting the intrusion present in the network. The proposed design focus on creating a Novel architecture though hybrid the Deep convolution neural network for improving the accuracy and increased security.

4.3 ARCHITECTURE

The robust architecture provides the collection of number of possible attacks in the massive internet of things network. The collection of intrusion models we call as bags of attacks.

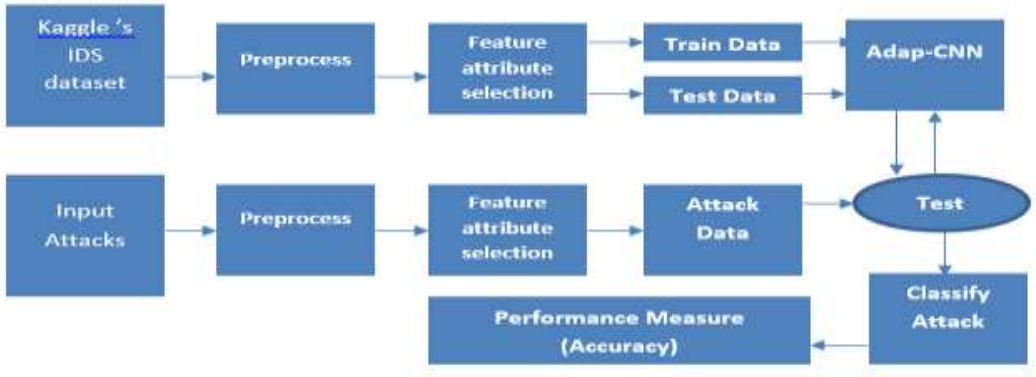

Fig 4.5 Architecture of proposed model

4.4 INTERNET OF THINGS

The great role of internet of things is to provide wide coverage on connectivity of devices that provides flexible connectivity and cyber world. It connects and communicated with different applications and enables the internet act as a medium for transferring data and hassle free services. Because of growth in internet of things in current scenario, the device is able to connect with small applications to complex applications, to provide connectivity between home appliances, IoT operations, general connectivity etc.

Data collection act as the important factor in growth of IoT. The data can be primary data and secondary data. The data are collected from sensors, camera, signal generators, text, video etc. The data collection phase are designed to act with node communications, constraints about the distance of the node, connecting priority nodes etc. the protocols are designed in such a way to provide efficient IoT connectivity strategies.

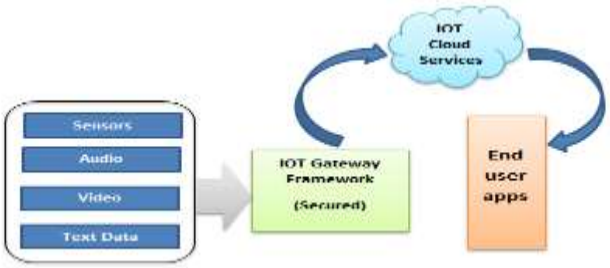

Fig 4.6 Architecture of IoT

4.5 Methodology

Considered attributes or wrong connectivity flag, login stamp, number of times, compromised login root stamp, scheduling units destination host. The training vector and testing vectors are formulated and normalized into 1x1000 samples. The training session and testing session of formulated using the deep convolutional neural network in which the layers or arranged like input 1000 x 1 X 1 followed by convolutional layer 1 x 10 stride 4 sized filter.

Followed by ReLu layer and fully connected layer of 384 refer to number of neurons in hidden layer. The fully connected layer is arranged with output class of 7. Softmax layer and classification layer are followed by the same. The training accuracy and testing accuracy are plotted by configuring the toolbox using SGDM model which is nothing but the gradient boosted Optimization model. At the end of the module, based on the correlation of training vector and testing vector the classification is made and relevant accuracy is calculated. The classification result shows the type of intrusion names such as 'Normal', 'Neptune', 'Guess Password',' Smurf', 'teardrop' etc. Traditional approach of calculating the accuracy is considering the predicted label and actual label with respect to the obtained test sequences.

CNN (ConvNet): A convolutional neural network (CNN or ConvNet), is a network architecture for deep learning which learns directly from data, eliminating the need for manual feature extraction. CNNs are particularly useful for finding patterns in images to recognize objects, faces, and scenes. They can also be quite effective for classifying non-image data such as audio, time series, and signal data.

Feature Learning, Layers, and Classification: Like other neural networks, a CNN is composed of an input layer, an output layer, and many hidden layers in between.

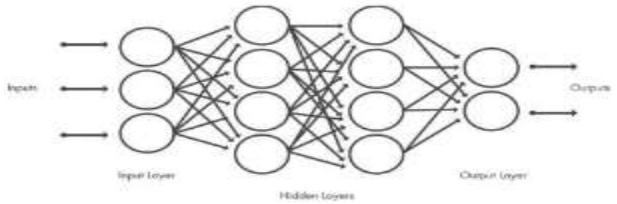

Fig 4.7 Layers

These layers perform operations that alter the data with the intent of learning features specific to the data. Three of the most common layers are: convolution, activation or ReLU, and pooling.

Convolution puts the input images through a set of convolutional filters, each of which activates certain features from the images.

Rectified linear unit (ReLU) allows for faster and more effective training by mapping negative values to zero and maintaining positive values. This is sometimes referred to as*activation*, because only the activated features are carried forward into the next layer.

Pooling simplifies the output by performing nonlinear down sampling, reducing the number of parameters that the network needs to learn.

These operations are repeated over tens or hundreds of layers, with each layer learning to identify different features.

Shared Weights and Biases: Like a traditional neural network, a CNN has neurons with weights and biases. The model learns these values during the training process, and it continuously updates them with each new training example. However, in the case of CNNs, the weights and bias values are the same for all hidden neurons in a given layer.

This means that all hidden neurons are detecting the same feature, such as an edge or a blob, in different regions of the image. This makes the network tolerant to translation of objects in an image. For example, a network trained to recognize cars will be able to do so wherever the car is in the image.

Classification of Layers: After learning features in many layers, the architecture of a CNN shifts to classification. The next-to-last layer is a fully connected layer that outputs a vector of K dimensions where K is the number of classes that the network will be able to predict. This vector contains the probabilities for each class of any image being classified. The final layer of the CNN architecture uses a classification layer such as soft max to provide the classification output.

4.6 Sequence Diagram

The sequence diagram, of IDS is,

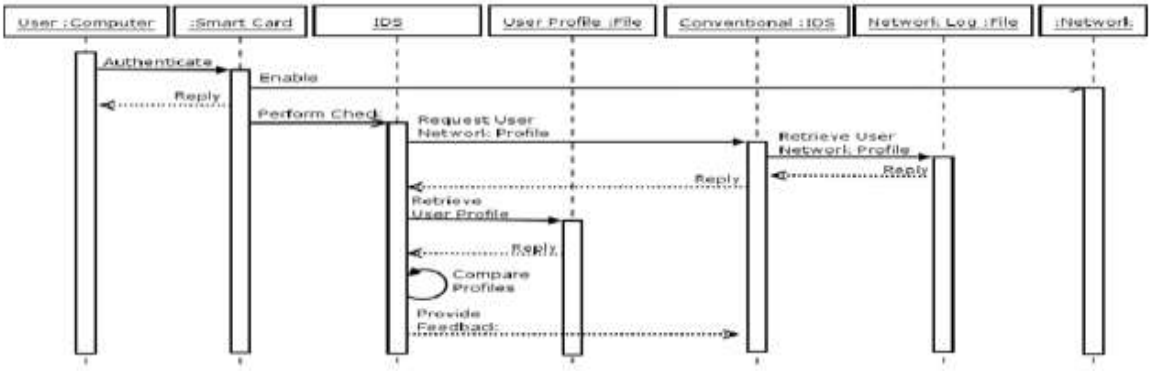

Fig 4.8 Sequence Diagram

4.7 Pooling Layer

Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data by reducing the dimensions. There are two types of pooling average pooling and max pooling. I've only had experience with Max Pooling so far I haven't faced any difficulties.

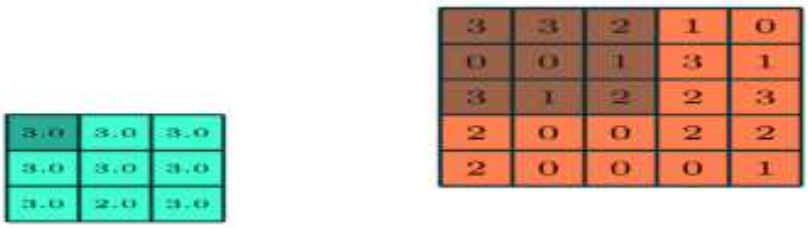

Fig 4.9 Max Pooling

So what we do in Max Pooling is we find the maximum value of a pixel from a portion of the image covered by the kernel. Max Pooling also performs as a Noise Suppressant. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction.

On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel. Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that Max Pooling performs a lot better than Average Pooling.

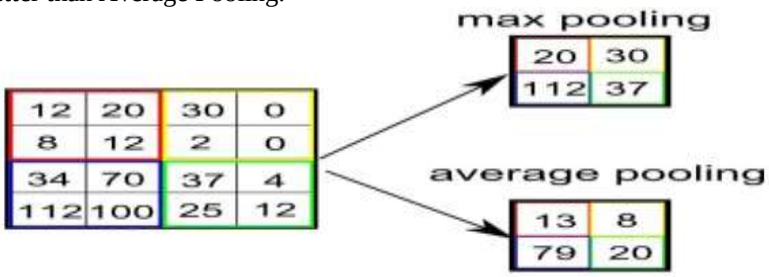

Fig 4.10 Classification

V. SOFTWARE REQUIREMENTS AND SIMULATION RESULTS

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

A Smurf attack is a form of distributed denial-of-service (DDoS) attack that occurs at the network layer. Smurfing attacks are named after the malware DDoS. Smurf, which enables hackers to execute them. More widely, the attacks are named after the cartoon characters The Smurfs because of their ability to take down larger enemies by working together.

DDoS Smurf attacks are similar in style to ping floods, which are a form of denial-of service (DoS) attack. A hacker overloads computers with Internet Control Message Protocol (ICMP) echo requests, also known as pings. The ICMP determines whether data reaches the intended destination at the right time and monitors how well a network transmits data. A Smurf attack also sends ICMP pings but is potentially more dangerous because it can exploit vulnerabilities in the Internet Protocol (IP) and the ICMP.

Basic Smurf Attack: *A basic smurf attack occurs when the attacker floods the target network with infinite ICMP request packets. Packets include a source address set to the network's broadcast address, which prompts every device on the network that receives the request to issue a response. This causes a massive amount of traffic, which will eventually take the system down.*

Advanced Smurf Attack: An advanced smurf attack starts as a basic attack. However, the echo requests are capable of configuring sources so they can respond to additional third-party victims. This enables attackers to target multiple victims simultaneously, which means they can slow down more extensive networks and target bigger groups of victims and larger sections of the web.

5.1 Simulation Results

The simulated results provide warning dialog box and alerts when intruder enters the system. And it is effective in detecting the intruder and it is easy to change the attacked area by the intruder. Simulated results provide good improvement in detecting the part whick has been attacked by the intruder. And the time which is taken to change the data which is attacked by the intruder is also less.

Warning Dialog: An alert dialog box is a special dialog box that is displayed in a graphical user interface when something unexpected occurred that requires immediate user action. The typical alert dialog provides information in a separate box to the user, after which the user can only respond in one way: by closing it.

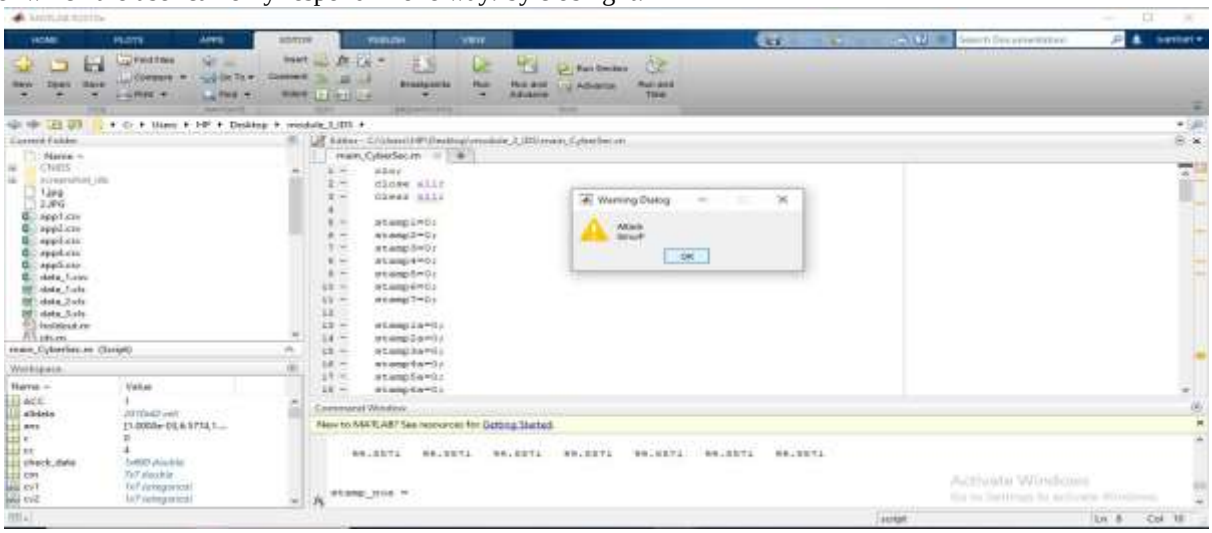

Fig 5.1 Warning Dialog Box

PCA Graph: Principal components analysis (PCA) is the most popular dimensionality reduction technique to date. It allows us to take an dimensional feature-space and reduce it to a kk-dimensional feature-space while maintaining as much information from the original dataset as possible in the reduced dataset. Specifically, PCA will create a new feature-space that aims to capture as much variance as possible in the original dataset.

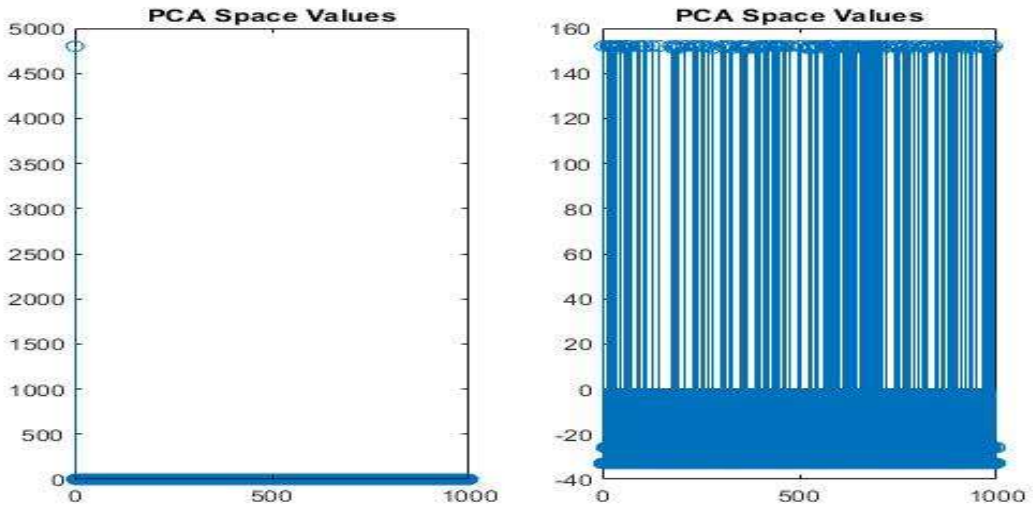

Fig 5.2 PCA Graph

Attacked Node: An attacker is an external node which intrudes into the WSN. A compromised node is an normal node (an unknown or an anchor node) in the WSN compromised by the attacker. Attacks on nodes are listed as follows:

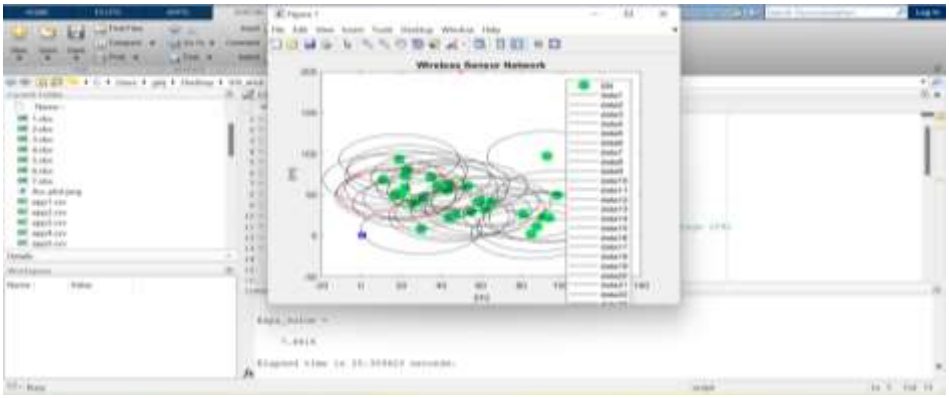

Fig 5.3 Attacked Node

Alert for Attacked Node: Alert Logic provides network intrusion detection (IDS) capabilities in order to inspect network traffic for signs of attack or compromise within a deployment. It is important to consider various factors when deploying an IDS, such as how traffic will be collected for analysis.

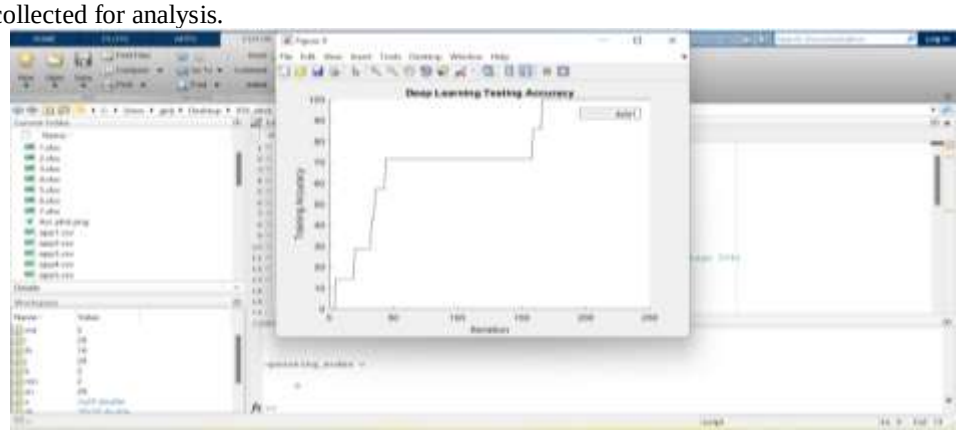

Fig 5.4 Graph For Attacked Node

VI. CONCLUSION

Cybercriminals are targeting computer users by using sophisticated techniques as well as social engineering strategies. Some cybercriminals are becoming increasingly sophisticated and motivated. Cybercriminals have shown their capability to obscure their identities, hide their communication, distance their identities from illegal profits, and use infrastructure that is resistant to compromise. Therefore, it becomes increasingly important for computer systems to be protected using advanced intrusion detection systems that are capable of detecting modern malware. In order to design and build such IDS systems, it is necessary to have a complete overview of the strengths and limitations of contemporary IDS research. Intrusion detection system can be used for monitoring file system for changes. It is helpful in detecting what changes are made to the system after an attack. An intrusion detection system is used to detect several types of malicious behaviours that can compromise the security and trust of a computer system. We intend to avoid the access and keep track of the intruder's attempts and intentions. Such a system can make a big addition to the security in today's world to avoid different kinds of attacks happening around. In existing system, it does not provide any alerts. Hence it took a lot of time in detecting the intrusion. But in proposed system, it performed a number of experiments to measure the performance of neural networks in intrusion detection, using the data for intrusion evaluation. All classifications were performed on the binary (attack / normal) basis. neural networks deliver highly-accurate (99% and higher) performance results.

REFERENCES

[1] K. Ogawa, K. Kanai, K. Nakamura, H. Kanemitsu, J. Katto and H. Nakazato, "IoT Device Virtualization for Efficient Resource Utilization in Smart City IoT Platform," *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Kyoto, Japan, 2019, pp. 419-422, doi: 10.1109/PERCOMW.2019.8730806.

[2] Eirini Anthi, Lowri Williams, Małgorzata Słowinska, George Theodorakopoulos, Pete Burnap, " A Supervised Intrusion Detection System for Smart Home IoT Devices", IEEE Internet of things 6 (5) , pp. 9042-9053.

[3] A. Krishna, A. Lal M.A., A. J. Mathewkutty, D. S. Jacob and M. Hari, "Intrusion Detection and Prevention System Using Deep Learning," *International Conference on Electronics and Sustainable Communication Systems* (ICESC), Coimbatore, India, 2020, pp. 273-278, doi: 10.1109/ICESC48915.2020.9155711.

[4] S. Choudhary and N. Kesswani, "Detection and Prevention of Routing Attacks in Internet of Things," *IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, New York, NY, USA, 2018, pp. 1537-1540, doi: 10.1109/TrustCom/BigDataSE.2018.00219.

[5] A. Ali and M. M. Yousaf, "Novel Three-Tier Intrusion Detection and Prevention System in Software Defined Network," *in IEEE Access*, vol. 8, pp. 109662-109676, 2020, doi: 10.1109/ACCESS.2020.3002333.

[6] P. Illavarason and B. Kamachi Sundaram, "A Study of Intrusion Detection System using Machine Learning Classification Algorithm based on different feature selection approach," *Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, India, 2019, pp. 295-299, doi: 10.1109/ISMAC47947.2019.9032499.