Hochschule für Technik Stuttgart

Anomaly Detection using Machine Learning Techniques

Master Thesis

Written by:
Aptheeri Reethika Hrudya Reddy

MatNr: 754156

Course of Studies: Software Technology (M.Sc.)

Start Date: February 12, 2019
End Date: August 12, 2019

Academic Supervisor: Prof. Dr. Ulrike Pado
Hochschule für Technik Stuttgart

Company Supervisor:Daniel Deckers
iC.Consult Gmbh,Fasanenhof

Affidavit

I hereby declare that this Master Thesis titled, “ Anomaly Detection using Machine Learning Techniques” and the work presented in it are my own. Furthermore, I declare that no sources have been used in the preparation of this document, other than those indicated in the document itself.

Place/Date/Signature

Acknowledgements

I have been fortunate to complete my thesis at one of the astounding software corporations iC-Consult. During the completion process of this thesis, I have collected great experiences and new lessons. I was able to get mental support from my surrounding people.

First of all, I would like to thank my supervisor at iC-Consult Mr.Daniel Deckers for his valuable guidance and kindness to me during the thesis at the company. It was really a great support and understanding where he let me find extra time to balance both thesis and family. Also, special thanks to my Prof. Ülrike Pado for all the motivation and appreciation in every occasion, without her support and guidance on how to deal with thesis and with my kid it would have been hard to imagine the completion of thesis. Many Thanks to a Professor who is also a mother which made me very lucky to have her as my supervisor.

Special thanks to my Parents for always believing in me. My Baby and my Husband to support me in hard times. And finally, I would like to thank my all time support, all my best friends who took care of my kid when I needed time and help. I am very grateful that all of you who gave me real help beside words of encouragement. Without all of you, this journey would have been very hard.

Abstract

Anomaly detection has become one of the essential tools in every company as it identifies the abnormal behaviors across data logs that are not identified when using traditional security methods.In today's world the design principles of the internet do not have a check point of how many requests the HTTP receives.As a result there are more chances that it undergoes an attack, one abnormal situation is Distributed Denial of Service attack.

Distributed Denial of Service(DDoS) attack is something that is highlighted in today's cyberworld. It specifically aims to bring the server down, by interrupting the normal server or network operation. According to some surveys DDoS attack has been increased 3 times by last year than any other attack[25].
There are several vectors that can be taken into consideration for an anomaly detection in this thesis but, the main reason to select this particular vector compared to all other vectors is, the company iC Consult accepts the challenging projects involving consumer identity, social media login, IAM managed services and so on[1]. So, it is very important to keep tracking the anomalous behaviour and prevent the attacks with the possible measures.The main aim of this thesis is to detect the chances of DDoS attack in a server by using number of counts to detect anomalous behaviour of particular ip_addresses in the data logs of the User-app.

This thesis will be ending in a way by justifying the name of the thesis "Anomaly Detection Using Machine Learning Techniques", successfully detecting anomalies in the data logs that are in hand using unsupervised learning method. By choosing 3 algorithms, applying them to the data and then comparing all the results to get an idea of which algorithm is best and accurate for anomaly detection.

Keywords: Anomaly Detection, Unsupervised Algorithms, DDoS attack, K-means, Isolated Forest,Clustering, Possible attack vectors

Contents

1 Introduction

In the recent times, daily schedule starts with the internet as a medium to exchange data or money in an open internet, hence prone to vulnerabilities. Distributed Denial of Service (DDoS) attack is a cyber attack that floods the victims system with several requests from several systems. By this attack the attacker gains control over the system and protected data which may lead to financial loss or may lead to the negative reputation of the company if their customers details are hacked and so on.This thesis intends to find DDoS attack through unsupervised anomaly detection algorithms.

1.1 Company Introduction

iC Consult designs and implements IAM(Identity and Access Management) solutions that are both compatible with existing systems and forward-looking.They are specialized in traditional enterprise IAM technologies such as governance, provisioning, single sign-on and federation. The company is actively engaged in projects involving consumer identity, social media login, IAM managed services and so on.

Identity Management controls the lifecycle of digital identities. Through an identity management system, the individuals or groups of people have access to applications, systems or networks by associating user rights as it identity management is nothing but an organizational process for identifying, authenticating and authorizing access. Access Management continuously identifies the users identity and ensures that only authorized persons access to a constantly growing number of resources and enforce access to each application(1). iC Consult takes care in all aspects and reduces the integration effort, which is also more efficient and safe.

1.2 Motivation and Goal

In login-based services, data integrity is very important. Every detail must be secure to prevent any third party modification i.e the data must remain the same in the system of the client and in the company's database until it is changed by the same user. Frequent Audit trail must be done to point where and when the data is differed in the company's system or in the client system when comparing both the systems, i.e every event is time stamped and recorded, the user has access to view it but cannot modify. Now one of the best ways to detect potential issues is by monitoring for anomalies.
Anomaly detection is the identification of events that are not belonged to expected pattern or to any other items that are present in a dataset. The goal of Anomaly Detection is to detect any meaningful rare event in a system, like fraud or network intrusion [2].

The main aim of the Thesis is to detect anomalies in a login based service. Select and analyze a particular vector and process the data. In reality these anomalous items have the potential to get into some kind of problems such as errors or frauds like Credit Card Fraud Problem, modeling past credit card transactions with the knowledge of the ones that turned out to be fraud.

There are various approaches like Unsupervised anomaly detection, semi-supervised, supervised for detecting anomalies. In this thesis the selected method is Unsupervised learning as the data that is used for anomaly detection is unlabeled data. This is the most preferred type of method when it comes to Anomaly Detection as the data logs are a set of unlabeled data. Unsupervised anomaly detection algorithm is easy to perform on the data and to obtain as they are the raw audit logs that saved on the server [3]. Then we identify if there exist any abnormal behaviour in the vector using the Unsupervised algorithms.For Example:Denial of Service (DoS) attack allows attackers to interrupt the server i.e the server is brought down by sending several requests by a single attacker to the victim until the server is flooded with requests, Password brute-force attacks aim to give an attacker the permission of administrators rights for the web server[2].

Now in this thesis we ensure the security of web-based applications and use Anomaly Detection(AD). As a rule, AD gathers data logs from the server under inspection, analyzes this data to detect abnormal activities.The selected vector is Distributed Denial of Service(DDoS), as understood from the above example DoS attack means the attacker brings down the system by sending more HTTP requests from a single system which becomes a traffic flood and the process of the system or the controlling of the attacks is slowed and the attacker succeeds by entering the system.DDoS attack is nothing but the attacker brings down the system by sending more requests from multiple systems which becomes a traffic flood[15]. Another important part in this is that we compare two unsupervised algorithms and its results on the same data logs.

1.3 Outline of work

This section outlines the structure of the further work in the form of the chapter structure. In addition, a brief description of the contents of each chapter is given.

- Chapter 2: BASICS

This chapter contains information that is essential to understanding the content of the Work required. Furthermore, a basic understanding of the anomalies i.e introduction to the anomaly detection, machine learning methods and their types and functions.

- Chapter 3: Possible Attack vectors and Data

  Some of the possible attack vectors that can be used in this thesis for anomaly detection depending on the user application.In this chapter, we will discuss about one vector that is selected from a list of vectors

  A descriptive analysis of data and how architecture showing how it is generated. A small description of the User-app that is used and the data logs collected from it which are important for the implementation of this thesis.

- Chapter 4: Approach

  This chapter gives an overview of the clustering and the selected algorithms. Why these particular unsupervised algorithms are taken into consideration compared to the other unsupervised algorithms. The three unsupervised algorithms are explained with briefly explaining the configuration set up, with the help of diagrams, and code snippets wherever needed.

- Chapter 5: Input Data

The input data that is considered and processing of it is also discussed in this chapter.

- Chapter 6: Algorithm Implementations

  In this chapter the implementation of the three selected algorithms are shown with the detected anomalies which is visualized in a t-SNE graph and also the ip_addresses that are anomalous.

- Chapter 7: COMPARISON OF ALGORITHMS and RESULT

  The chapter on the project should give an insight into the process of comparison and results of the project.

- Chapter 8: SUMMARY AND OUTLOOK

  This last chapter will first discuss the findings of this work, outlook necessary steps as well as meaningful extensions for the future scope using this thesis.

2.Basics

The basics chapter gives the idea about the anomalies, machine learning methods and the differences between the learning methods. The selected learning method is also given an overview to get the idea of it.

2.1 Anomalies

Anomalies are considered to be the abnormal behavior or something that deviates from the standard, normal or expected action(1). For example: If a person is changing their password once in a month from long time and suddenly that accounts password changes two to three times a week, this abnormal behaviour is considered to be an anomaly and finding this sort of behaviour in early stages is called Anomaly Detection(AD).

Anomaly detection techniques constructs an overall view of expected behaviour that would be normal using data logs that are collected from long period of normal system activity. These data logs are collected in an environment that is secured, then it is analysed and kept in a proper way to ensure that the anomaly detection is possible. When the anomaly detection is applied, the system sometimes detects false alarms due to the difficulty in obtaining the normal behaviour of the system. Sometimes the system May detect or generate excessive number of false alarms and report as anomalous events, which are not necessarily abnormal activities[31].

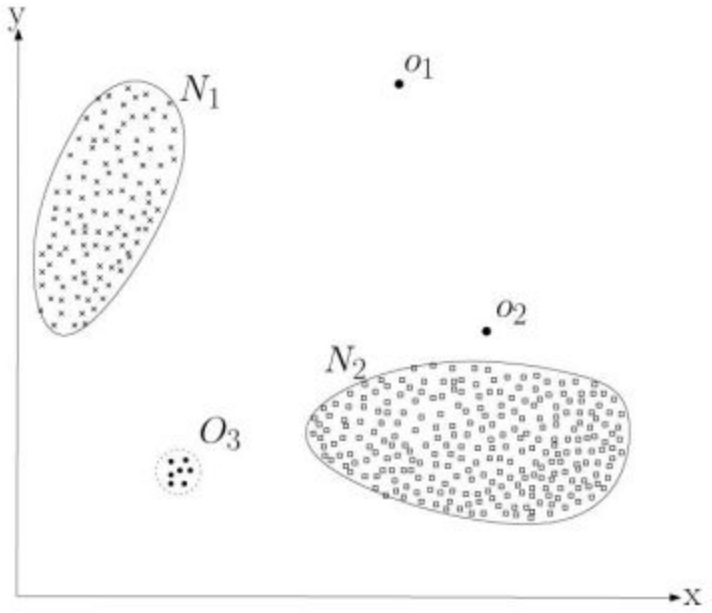


Figure 2.a: Example of anomalous data points[4]

A concrete example is the situation in Figure 2.a.There, two areas N 1 and N 2 (N for normal) are formed of points of a data amount.These data points are composed of values of two different features x and y.
In addition, there is the expectation that newly added points in these two areas accumulate . Data points of these two quantities are thus considered normal Basics. Now there are two points o 1, o 2 and the set of points O 3, all far away from both normal amounts, these may be considered as anomalies.In general, therefore, the anomaly can be described as something that depends on the expectation and deviates to a certain extent[4].

One of the advantages is, the configuration of the system can be unattended as it automatically learns the ‘’behaviour of large number of subjects’’ and can run itself. The drawback is that when the system learns automatically the schedules of the tasks, there is a chance of accepting the abnormal behaviors as normal for a particular user. When it slowly gets adapted to this kind of behaviour, it might consider the attack as normal behaviour[2].

2.1.1.Types of Anomalies:

There are three types of anomalies and their use cases differ from each other[4].

Point anomalies: A single instance of data is anomalous if the data point is far away from the rest of the clustered points.

Business use case: Detecting bank transaction fraud based on "the amount spent."

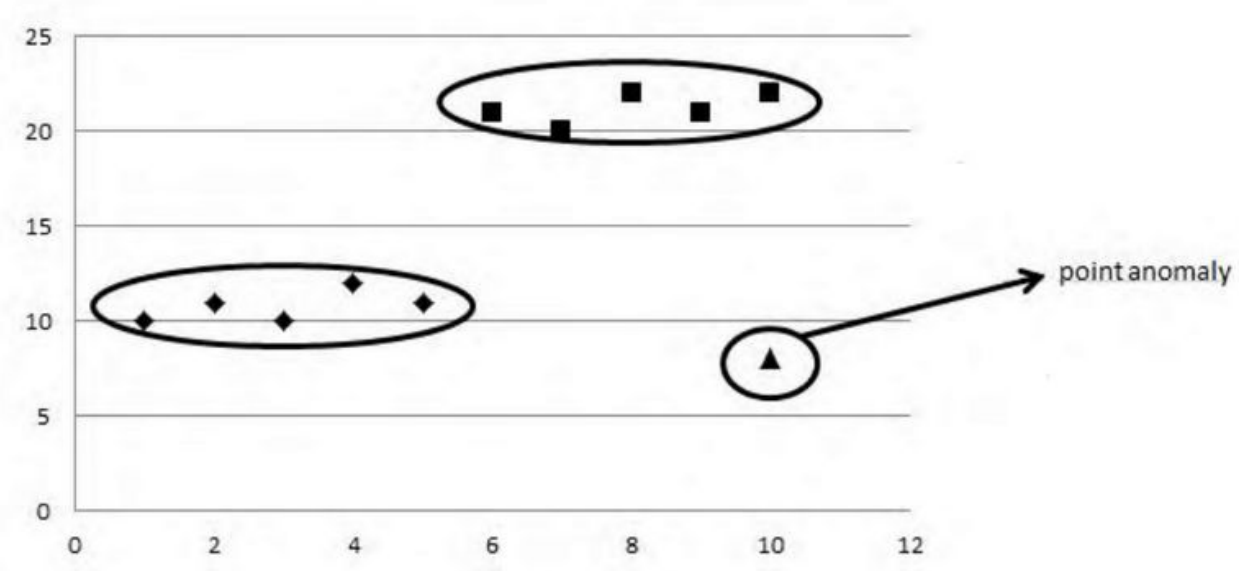


Figure 2.1.1a. An example of point anomaly[33]

Contextual anomalies: This is meant by the kind of abnormality in any kind of contextual situation. This type of anomaly is mostly detected in time-series data.

Business use case: Spending 100 euros on food during weekends or during the holiday season is normal, but may be everyday during weekdays.

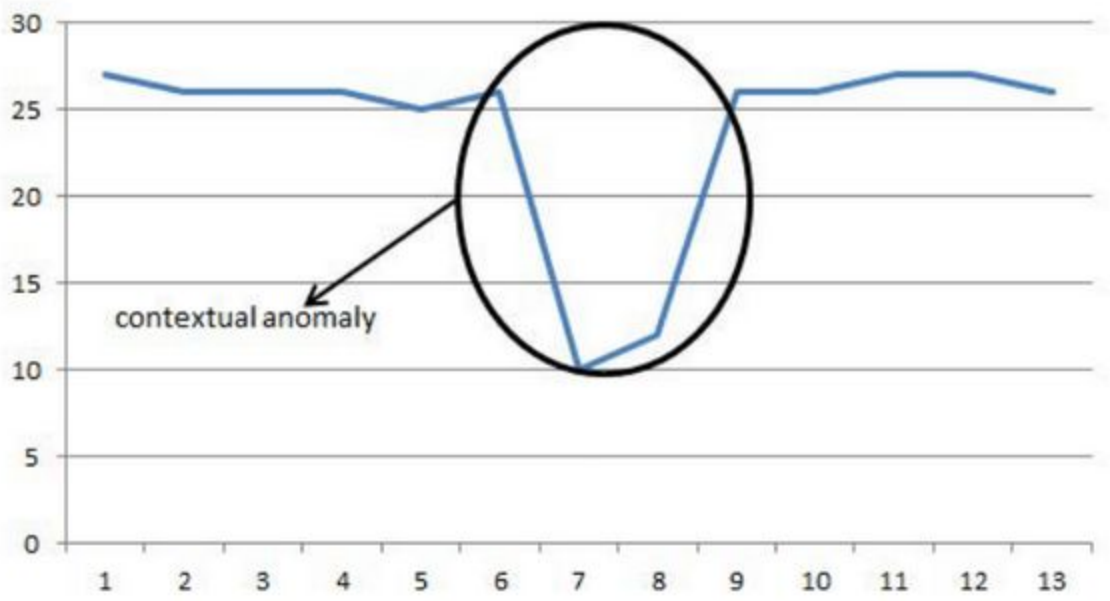


Figure 2.1.1b An example of contextual anomaly[33]

Collective anomalies: A collective set of data instances helps in detecting abnormalities.

Business use case: Every one of the train drivers stand out on the platform to stop all the running trains on the same day is a collective outlier because although it's definitely rare that one or two trains stop due to some reason, but it is very unusual that an entire rail crew is not in duty at the same time.

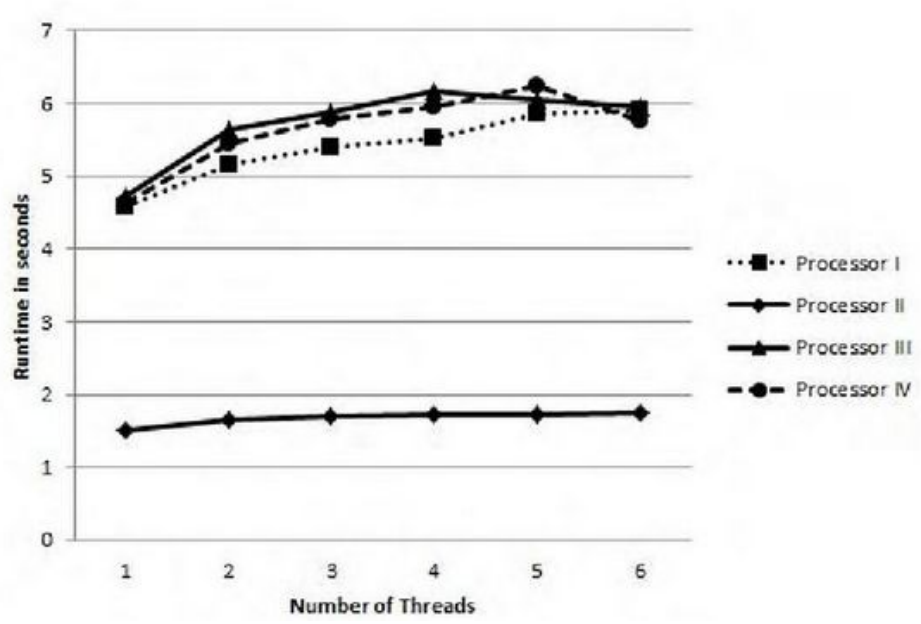


Figure 2.1.1c An example of collective anomaly(anomaly created by processor ii)[33]

2.2 Machine Learning Techniques for AD

There are basically three types of learning methods in the Machine Learning, that are used to detect anomalies depending on the type of data that is available.

2.2.1 Supervised Anomaly Detection

Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs.The data is also known as training data, and consists of a set of training examples. Each training sample has one or more input and the desired output. Theoretically, when comparing it to semi-supervised and unsupervised methods, supervised methods provide better detection rate, since they have labeled data and also have access to more information[5]. The use of this algorithm is not possible to everyone as the labeled data is hard to get, need experts and special devices[9].
According to the figure 2.a. There is a test data that is an input variable let us consider it as(x) and a result that is an output variable, we can consider output variable as (Y) and then use an algorithm to learn the mapping function from the test data to the result.

$$Y = f(X)$$

When the training data set is given,the algorithm iteratively makes predictions on the training data, as the model is trained, it corrects and the learning stops when the algorithm achieves an acceptable level of performance.
Supervised learning model can be further grouped into regression and classification.

- **Classification**: A classification is when the output variable is a category or discrete. For example, if a student gives an exam, thesult shown in classification is pass or fail.
- **Regression**: A regression is when the output variable is a continuous or a real value.For example, in regression if the student gives an exam then the result is shown in percentage i.e in numerical value [9].

2.2.2 Semi-supervised Anomaly Detection

Semi-supervised learning is a learning model that is more bothered of how computers and natural systems such as humans learn in the presence of both labeled and

unlabeled data[9]. Normally the learning has been done either in the unsupervised algorithm, where all the data are unlabeled, or in the supervised algorithm, where all the data are labeled. Semi-supervised learning is used by combining both labeled and unlabeled data, typically a small amount of labeled data and a large amount of unlabeled data[10].

‘’This kind of semi-supervised learning method is of great interest in both machine learning and data mining because it can use instantly available unlabeled data to improve supervised learning tasks when the labeled data are scarce or expensive’’[9].

2.2.3 Unsupervised Anomaly Detection

Unsupervised technique do not need labeled data. As an alternative, they are based on two basic assumptions. First, they presume that only a very small traffic percentage is abnormal and most of the network connections are normal traffic. Secondly, this technique anticipate that malicious traffic is statistically different from normal traffic. According to these two assumptions, data groups are assumed to be normal traffic that are of similar instances which appear frequently , while infrequently instances which considerably various from the majority of the instances are regarded to be harmful [3].Even though theoretically supervised may be better, but in practise unsupervised anomaly detection has many advantages compared to supervised. One of the main advantages is input data, which is unlabeled and easy to obtain[3].

The unsupervised anomaly detection is done by taking the input of unlabeled data. Then the unsupervised algorithms are passed on the unlabeled datasets, where the algorithms detects is there are any anomalous activities that has been occurred in the logs and the result with the normal and as well as anomalies are given as output[26].

2.3 Unsupervised AD

2.3.1.Methods

Unsupervised anomaly detection methods can be categorized as follows[20]:

(i) clustering based methods

Clustering based methods are the entities of each group are similar compared to the other group, similar group of data in the data set forms clusters[22].

(ii)Density based methods
The data points that forms clusters are separated and forms the high point density and low point density‘.Density based methods define anomalies as instances which exist in areas of low density[23].

(iii) Relative density based methods
The difference of density between different clusters should be relative, not absolute. Relative density based method use density ratio between an instance and its neighbourhood, which are introduced to overcome this issue[24].

2.3.2.Related Questions

Why is this method suitable for anomaly detection?

Which unsupervised learning methods are considered?

Properties of the desired recognition results!

How do the respective procedures work?

Are there any special requirements?

3. Possible Attack vectors and Data

3.1. Possible Attack vectors

Here are some of the list of attack vectors based on Feature modelling techniques. The selection of the vector in this thesis is based on the data in hand, the vector that is apt and how important is it for a company.

Vector 1. SQL injections:

SQL injection attacks targets the databases that are accessible through a web front-end, and take advantage of imperfect input of Web components[14].The attackers insert malicious SQL statements and these statements control a database server behind a web application. Attackers can use this vector to go around authorization of a

web application and retrieve the content of the entire SQL database. They can also use SQL Injection to add, modify, and delete records in the database.[27]

Vector 2: DDoS(Distributed Denial of Service) attacks:

The Distributed Denial of Service attack is something that is highlighted in today's cyberworld. It specifically aims to bring the server down, by interrupting the normal server or the network operation[15]. DDos attack is succeed by sending multiple requests from multiple systems at the same time and floods the server, so it gets breakdown. Denial of Service(Dos) attack is nothing but a single system that is attacking the server by sending multiple requests. It is possible to stop the attack as it is only one system that is attacking the server. But it is impossible to avoid DDos attack.

Vector 3. Path traversals

This path traversal is one of the vectors where the attacker concentrates on accessing the file system by using path or a directory. Attackers use HTTP to access restricted directories and execute commands outside of the web server's root directory. Using this vector the attacker accesses the restricted files and gets more information to further have the complete control on the system[28].

Vector 4. Password Brute-Forcing

In password brute forcing vector means the attacker tries to force login using the user-id and password. This can be done manually or automatically as they run the program with all combination of words and numbers at one point they gain access to the user. This kind of vector is used to analyze an organization's network security, analysts implement these techniques to determine the vulnerabilities in their system[29].

Vector 5. Forced Browsing

Force browsing is a kind of attack technique that is forcing the restricted pages or other resources in a web server by forcing the URL directly. In case there are anything that is not authorised properly then there are more chances of information to be leaked to the attackers. It is important to detect and take proper measurements to avoid anomalous behaviour. The result of this vector attack may lead the attacker to access sensitive information about web applications and operational systems, such as source code, credentials, internal network addressing, and so on[30].

Vector 6:Cross-Site scripting

Cross-site scripting (XSS) is a type of system security vulnerability mainly found in web applications. Cross-site scripting permits attackers to inject client-side scripts into sites viewed by different users. Attackers use this cross-site scripting vulnerability to bypass access controls such as the same-origin policy. XSS effects vary in vary from nuisance to vital security risk, looking on the sensitivity of the information handled by the vulnerable web site and therefore the seriousness of the protection implemented by the site's owner[32].

3.2. Selected Vector: DDos Attack

Denial of Service attack is the most dangerous attack as the attacker brings down the system by sending more HTTP requests which becomes a traffic flood and the process of the system or the controlling of the attacks is slowed and the attacker succeeds by entering the system. But in case of Denial of Service attack as it is only one system, it is possible to control the traffic. When we consider the Distributed Denial of Service, it is the multiple systems sending the flow of requests at the same time and successfully attacking the targeted Victim i.e the server or the browser[15].

Denial of Service is the precursor to the Distributed Denial of Service. The reason of this kind of DDoS attacks are, a victim can be an e-commerce site, a bank, a commercial organization, or an Internet service provider (ISP),the attack leads these users for some financial gains19].A research survey conducted in the first quarter of 2014 by Prolexic highlighted that there is a 47 % increase in total DDoS attacks in comparison to the first quarter of 2013[25].

The attacker can set the date and time and the botnets work according to the set command, as planned the botnets sends requests to the server at the same time to slow down the system. This results to denied service i.e the system is busy dealing with ddos attack, the web pages that the customers want to access are not going to load or keeps on loading very slowly with the spiral loading symbol on the screen.
They can stay for an hour or for days.

Reasons:

Financial reason- when they need to compete with their competitors in the marketplace.
Political reason- when the people don't like the false promises in the site.

Sometimes for Fun- a person can be ddosed by his fellow friends just to win the game they were playing.

How:

They attacker will develop a malware program and send to the other unknown systems which are also known as botnets through email, websites attachments. If these attachments are opened, the attacker has control of these systems without the knowledge of the owner. Later on the attacker programs the date and time a which server to attack, these systems are ready with the malware program and attacks the server on the time set.

By this itself we can imagine how important it is to detect an anomalous behaviour in the company. The main reason to select this particular vector compared to all other vectors is, the company iC Consult accepts the challenging projects involving consumer identity, social media login, IAM managed services and so on. It is very important to keep tracking the anomalous behaviour and prevent it with the possible measures.

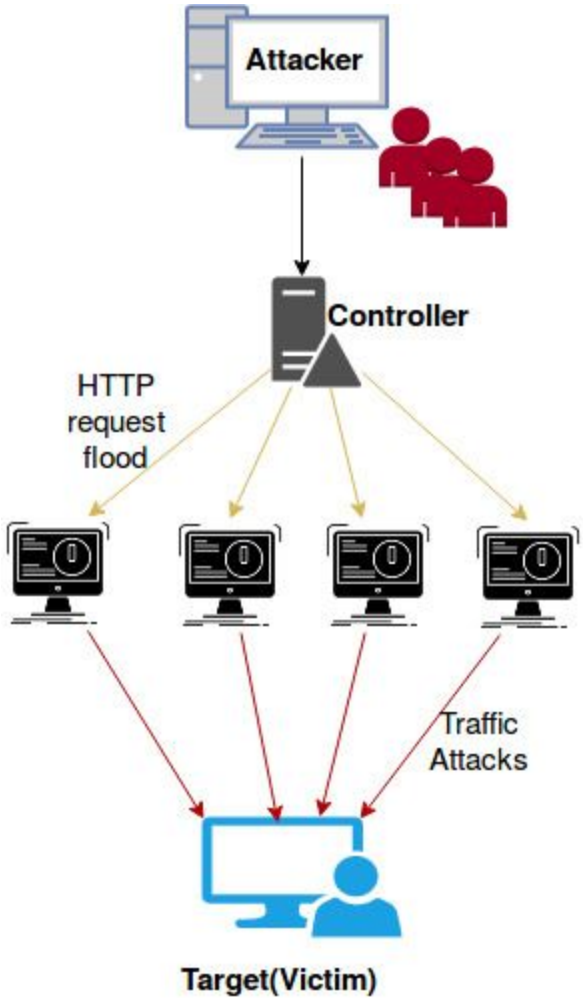


Fig.5.a Distributed Denial of Service attack(add botnet for black systems)

We can detect the anomaly in a system by number of counts of the ip address. If someone is trying to throw as many requests continuously, we can track if the person is really attacking the server by the number of counts that is creating the traffic.
“With the rapid development of network technology in recent years, the attack traffic scale caused by Multiple Identity Attack attacks has been increasing, with the targets including not only internet infrastructures such as firewalls, routers and DNS system as well as network bandwidth, but also business servers, the attack influence sphere has also become broader”[16].

Generally there are two different detection of DDoS attack approaches, signature based and anomaly based that are used by Intrusion Detection Systems [35].In the signature based method, a set of known abnormal activities is fed into the system, some specific features are extracted and used as the signature of those activities. When a new activity is enter the system, its features are compared with the database and any match is

reported as anomaly. In Signature based approach it cannot detect new intrusions which are not in its database [36]. In anomaly based approach, the pattern of normal activity is determined and any structure out of this model is reported as anomaly.The detection rate of this approach is less than that of in signature-based method [35].
Now this thesis is mainly about DDoS attack detection using unsupervised methods of anomaly detection. Here in chapter 5 are the implementations of the Isolation forest and the K-means method for the detection process, with the description of data that is taken into consideration.

In this DDoS attack, detected taking the time series that is generated based on the entropy of HTTP GET request per source IP address.

3.3 Data

Related questions

What data is considered and how is it processed?

What types of recording data are available?

What data is used for anomaly detection?

How are the recording data structured?

Data detection specific to each user or over the whole?

The data that is used in this thesis is obtained from ELK stack, which pulls the data logs from the web application User-svc.The User-svc project is mainly used for end-consumer portals, to register and update user profiles and to give the user access to their applications, e.g. online shops, support portals or car entertainment. It is also used by some clients for internal users (managed by HR). The users who registers their profiles and access every time, their login data is collected into ELK stack automatically and stored for 1 month. Later on it is programmed to be deleted from the Kibana.

The whole data logs that are created, are not in the proper Json format. Therefore a lot of effort is put into this to get the data fields accordingly for the detection of selected vector, so firstly the data is cleaned, filtered and then used.
Here in the Fig.2.b, a clear explanation is given how the data is extracted and been pulled by Elastic search and automatically saved on to the systems. This saved data logs are been used in the thesis to detect anomalies.

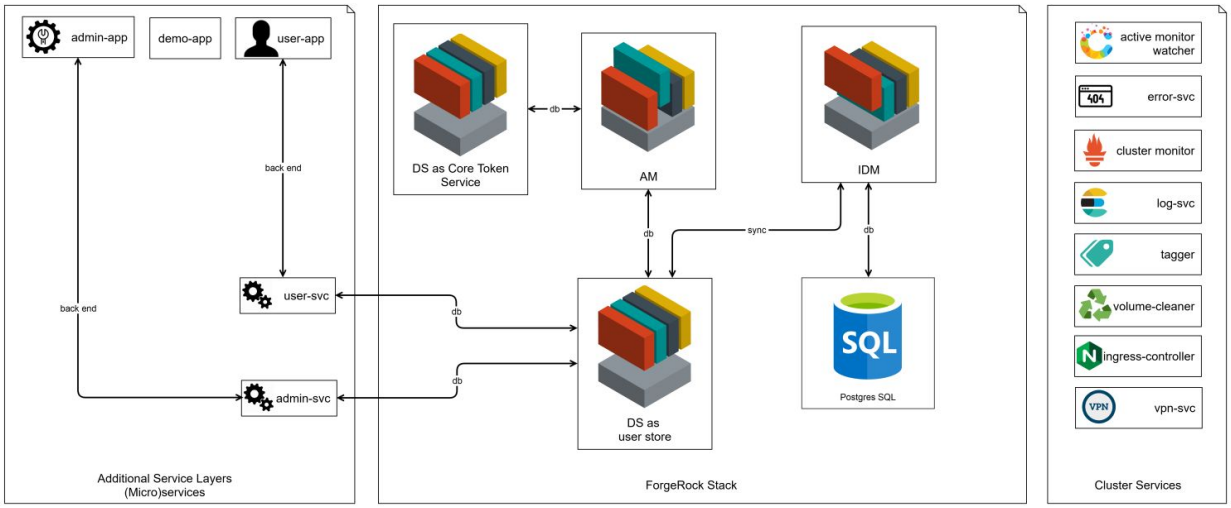


Fig:3.2.a. Cluster Deployment

The Fig:3.2.a represents a 3-tier architecture. The Swagger UI is as the frontend demo app, it is an open source project to visually render documentation for an API defined with the OpenAPI Specification.Scopes are used to grant AN application totally different levels of access to information on behalf of the top user.

Each API may declare one or more scopes. Here the API requires the scopes and we select which ones we want to grant to Swagger UI. Backend for User profile operations is built with the java code and Maven build is used to build and the spring-boot is used to run the project. The data that is given by the user in the user app is sent to the data store. A data store is a repository for storing and managing collections of data which include not just repositories like databases.

ForgeRock is an access management and identity software company, it develops access management products and commercial open source identity for internet of things, customer, cloud, mobile, and enterprise environments.[6][7]. Later from Data Store the AM(Access management) takes charge on granting authorised users the right to use a service, while preventing access to non-authorised users. Parallely IDM (Internet download manager) and data store are synchronised and Postgresql communicates with the IDM. IDM could be a tool to manage and schedule downloads.

The reason behind having IDM is it has recovery and resume capabilities to restore the interrupted downloads due to lost connection, network issues, and power outages. There is another system called Cluster Services. It is nothing but a group of servers are connected on a single system. The moment one of these servers experiences a service

outage, the workload is redistributed to another server before any downtime is experienced by the client. Here we used Clustered servers for applications with frequently updated data with file, print, database and messaging servers ranking as the most commonly used clusters.The data is automatically pulled by ELK stack.

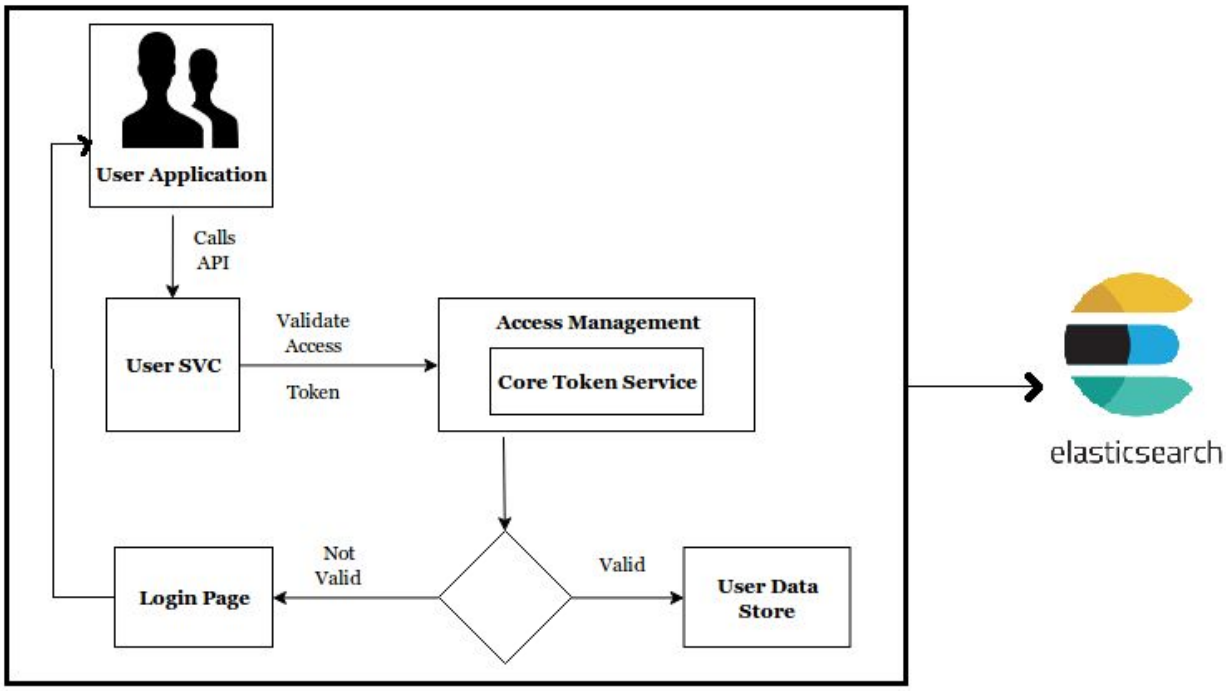


Figure 3.b: UML diagram of the User-app

User-app calls user-svc, which validates the access token against Access Management, which uses the Core Token Service. If the access token is not valid, a login form will be displayed (by the user-app again). If the Access token is valid, it uses the user data store to get and update the user information. Elasticsearch is involved all the time in the process and collects the complete logs automatically from all systems.

4.Approach

4.1 Clustering:

Clustering analysis is a kind of unsupervised study method.Clustering is nothing but a set of similar objects or the same group objects are grouped together as one cluster,similarly other group objects are clustered according to their similarity with each other ,i.e it is the process of partitioning different objects into the same class it belongs to which forms a cluster, some modes are regarded as the same class if they have equal distance of characteristic vector within the scope of certain error margin [11].

Clustering techniques apply when the instances are to be divided into natural groups and there is no class to be predicted .The instances are drawn from the clusters that undoubtedly reflect some mechanism that is at work in the domain, a mechanism that causes some instances to bear a stronger resemblance to each other than they do to the other instances. Clustering requires various techniques to the association and the classification learning methods that we have considered so far[12].

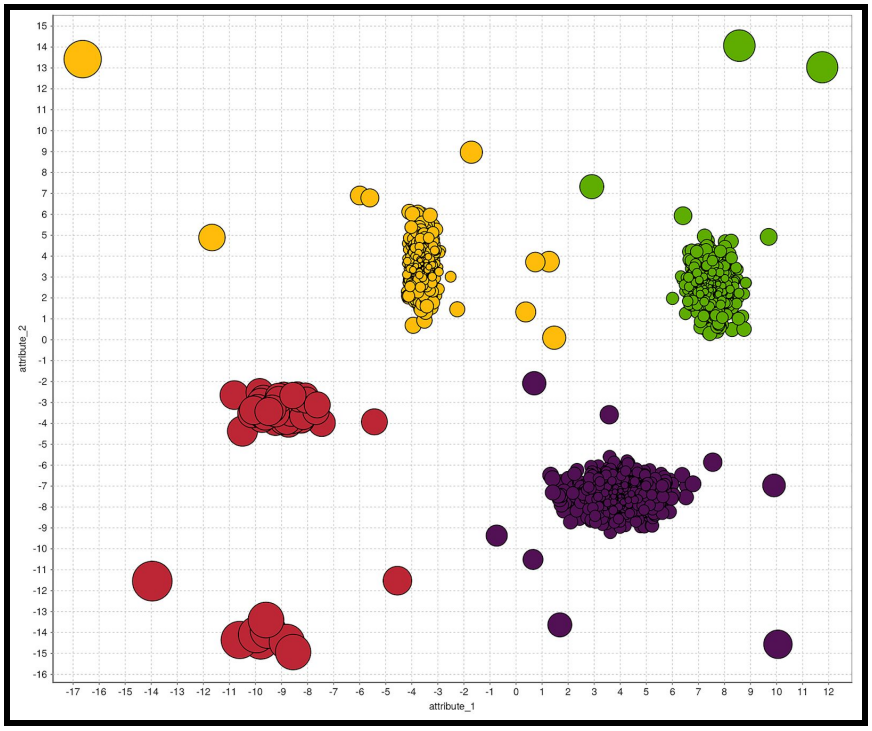


Figure:4.1.a Clustering of data sample[20]

Figure.4.1.a shows the outcome of applying a clustering algorithm.The anomaly detection data in the above clustering algorithm is represented by the points, whereas the color corresponds to the clustering result of one of the clustering algorithms[20].

Here in the figure 4.1.a We can see the different colours of data points which are forming clusters. The whole clustering graph is from a single data sample but with different Fields in that data sample, that are taken into consideration for the anomaly detection. Similar objects are considered as one cluster i.e same colours of points forms one cluster. The points that are far away from the centroid are the outliers i.e the abnormal behaviour occurred in the data.

4.1.1 Higher Dimension Graph

From Figure 5.2.4.b in this thesis there is a graph that is used to understand the relationship between the variables. While most relationships are looking like clusters, the td_max vs is_weekend_ratio is a giant blob.
In case there were an infinite number of ip_addresses and if it was our choice then the graph would have looked smoother with equal distribution.
There are various ways to describe the graph or visualize it using statistical procedures PCA or t-SNE.

PCA- Principal Component Analysis

The basic idea behind PCA is to find the components so that they explain the maximum amount of variance possible by n linearly transformed components.
The basic goal in PCA is to cut back the dimension of information. Indeed, it may be tested that the illustration given by PCA is associate optimum linear dimension reduction technique within the mean-square sense. Such a reduction in dimension has important benefits. First, the machine overhead of the following process stage is reduced. We tried this process in this thesis but did not consider it as a best idea.

The main reason that PCA is not used in this thesis is that, even though it has advantages of dimensionality-reduction method by transforming a large set of variables into a smaller one that also contains most of the data within the massive set. There is some loss of information during the dimensionality reduction[47].

t_SNE - t-Distributed Stochastic Neighbor Embedding

The best method to project higher dimensional data into lower dimensions.The function of kernel is to take data as input and transform it into the required form.These functions can be different types but we use RBF as it has localized and finite response along the entire x-axis.. The Gaussian/RBF kernel employed in the high-dimensional space by t-SNE defines a soft border between the global and local structure of the data. Also for pairs of data points that are close together relative to the standard deviation of the Gaussian, the importance of modeling their separations is almost independent of the magnitude of those separations.
Moreover, t-SNE determines the local neighborhood size for each datapoint separately based on the local density of the data. When there is huge set available then It is

possible to pick a random subset of the data points and display them using t-SNE.

According to [48] there are also three weaknesses of t-SNE, even then useful to visualize in our thesis.
(1) it's unclear however t-SNE performs on general spatial property reduction tasks.
(2) the local nature of t-SNE makes it sensitive to the curse of the intrinsic dimensionality of the data
(3) t-SNE is not guaranteed to intersect to a global optimum of its cost function[48].

4.2 K-Means

k-means is one of the famous unsupervised algorithms that solve the well known clustering problem[5].This algorithm is one of the most popular partitioning unsupervised methods. This K-means clustering is the most used clustering algorithm and is simple compared to others. The intention is to classify the info into k clusters wherever k is that the input parameter laid out in advance through repetitious relocation technique that converges to native minimum[18].

K-means for describing an algorithm that assigns each item to the cluster having the nearest centroid.K is nothing but the hyperparameter of the model. When the value of k is given and tries to build k clusters from samples in the dataset. The value of k depends on the data and the way data has additional features, so it is not easy to determine the right value of k[41].

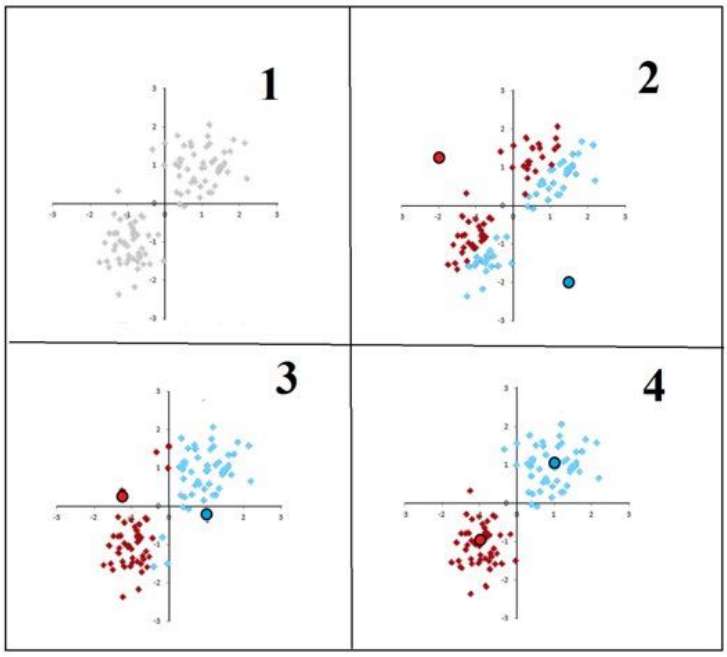


Figure 4.2.a K-means clustering[41]

As already defined in the clustering division in 4.1, a cluster is nothing but a group of points. In k-means the Entity called a centroid has a key role. A centroid is also a point in the data space, it is in the centre of all points that helps to form a cluster.
According to the citation[19], The algorithm is expressed as follows.

k-means Clustering Assigning data points into clusters

Input:
D = {d1, d2,....,dn} //set of n data-points.
C = {c1, c2,.....,ck} // set of k centroids
Output: A set of k

clusters Steps:

1. Compute the distance of each data-point di (1<=i<=n) to all the centroids cj (1<=j<=k) as d(di, cj);
2. For each data-point di , find the closest centroid cj and assign di to cluster j.
3. Set ClusterId[i]=j; // j:Id of the closest cluster
4. Set Nearest_Dist[i]=d(di, cj);
5. For each cluster j (1<=j<=k), recalculate the centroids;
6. **Repeat**
7. For each data-point di

7.1 Compute its distance from the centroid of the present nearest cluster;

7.2 If this distance is less than or equal to the present nearest distance, the data-point

stays in the cluster;

Else

7.2.1 For every centroid cj (1<=j<=k) Compute the distance d(di, cj);

Endfor;

7.2.2 Assign the data-point di to the cluster with the nearest centroid cj

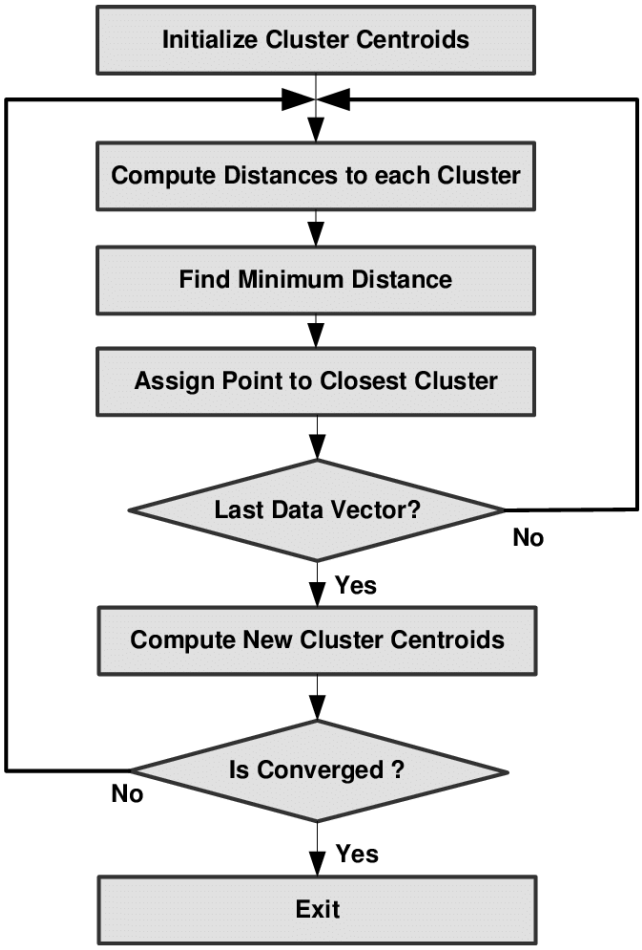
7.2.3 Set ClusterId[i]=j;

7.2.4 Set Nearest_Dist[i]=d(di, cj);

Endfor;

8. For each cluster j (1<=j<=k), recalculate the centroids;

Until the convergence criteria is met[19].



\

Fig:4.2.b K-Means algorithm steps[17]

4.3 Isolation Forest

Isolation forest is one of the latest techniques to detect anomalies and it is unique AD because it utilises an isolation mechanism to detect anomalies[21].

The term isolation means that ‘separating Associate in Nursing instance from the remainder of the instances’.Since anomalies are ‘few and different’ and therefore they are more susceptible to isolation.[42]

According to the citation[21] the algorithm is based on the fact that anomalies are data points that are completely different and very few. As a result of these properties, anomalies are liable to be influenced to a mechanism called isolation.

The main advantage of this algorithm is because of its linear execution time and it works with huge dataset in different dimensions, which makes it extremely efficient when compared to other methods, and thus it is a very interesting option for the challenge of large datasets.
Since algorithmic partitioning is delineated by a tree structure, the quantity of splits needed to isolate a sample is equivalent to the pathlength from length from the root node to the terminating node.This particular path length that is averaged over a forest of such random trees is a normal one and our decision function.

Random partitioning produces shorter paths for anomalies that are noticeable. Hence, those instances which can be easily isolated are likely to be anomalies when a forest of random trees collectively produce shorter path lengths for particular samples[21].

Here are the steps to implement the iForest algorithm and the important step to implement choosing a feature randomly.

Steps

1. Select the point to isolate.
2. For each feature, set the vary to isolate between the minimum and therefore the most.
3. Choose a feature randomly.
4. Again randomly pick a value that’s in the range.
    - Switch the minimum range of the feature to the value, If the chosen value keeps the point above.

- Switch the maximum range of the feature to the value, If the chosen value keeps the point below.

5. Until the point is isolated repeat steps 3 & 4 . the point should only be the one which is inside the range for all features.
6. Count how many times you've had to repeat steps 3 & 4. We call this quantity the isolation number.

The algorithm identifies an outlier if it doesn't have to repeat steps 3 & 4 several times.

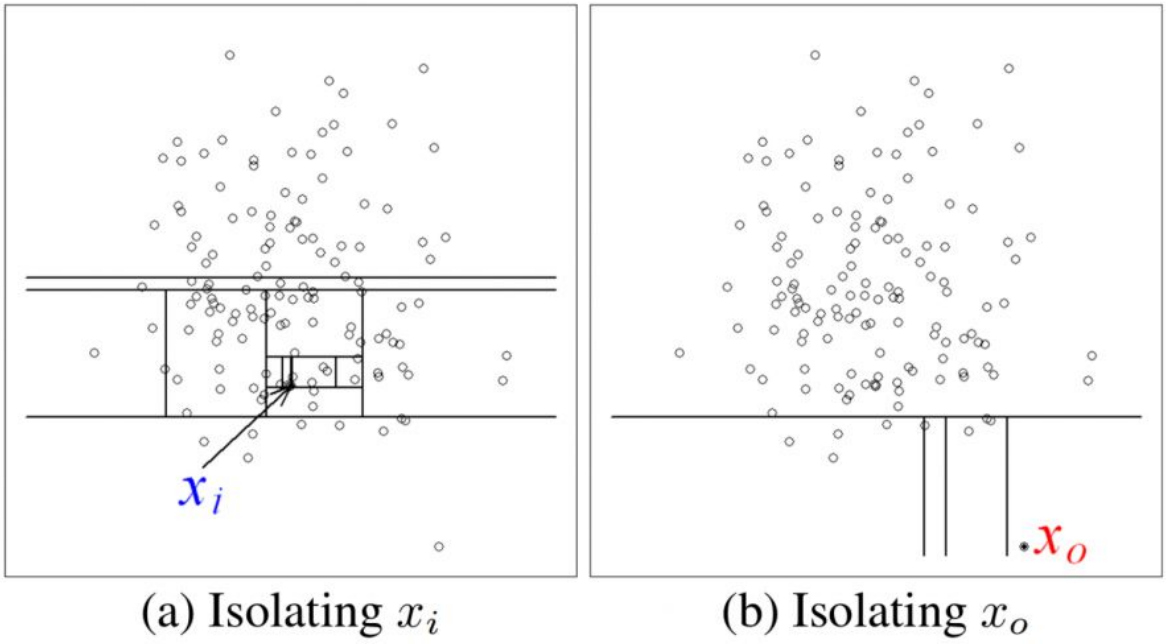


Figure 4.3.a Isolation forest anomaly detection graph[42]

From the figure 4.3.a we can view a data-induced random tree, here partitioning of instances are repeated recursively until all instances are isolated.
This random partitioning gives the shorter paths for anomalies that can be noticed, since the fewer instances of anomalies result in a smaller number of partitions. Shorter paths in a tree structure and instances with distinguishable attribute-values are more likely to be separated in early partitioning[42].

Hence, when a forest of random trees collectively produce shorter path lengths for some particular points, then they are highly likely to be anomalies. We can distinguish anomalies by Xo and normal data points by Xi.

4.4. One class Support Vector Machine

The Support Vector machine is a supervised learning model where as One Class support vector machine is an unsupervised algorithm and is good for detecting anomalies.
According to a citation[39] A support vector machine is composed of two main components. The first component uses quadratic or linear programming to find an optimal separating hyperplane between the two sets of samples in some defined attribute space. One-class SVM is widely adopted in one-class classification fields. However, outliers in the training set negatively influence the classification surface of OC SVM, degrading its performance[43].

The second component is a mapping function, also called a kernel function, which transforms the input space to an arbitrary higher dimensional attribute space which allows a better linear discrimination to be constructed by the first component. From this citation we can see that Tax & Duin (1999) and Schölkopf et al. (2001) proposed one-class or anomaly detector variants of the SVM[39]. The training data should have only one class of data which we are sure are non-anomalies. When predicting, we predict for all classes, hence identifying the anomalies.

The one-class SVM algorithm has the two parameters μ and *gamma*. The correct choice of these parameters has a big influence on the quality of the model.[45].

Figure 4.3.a illustrates the basic principle of the one-class SVM. A hyperplane separates most training samples from the origin[39].

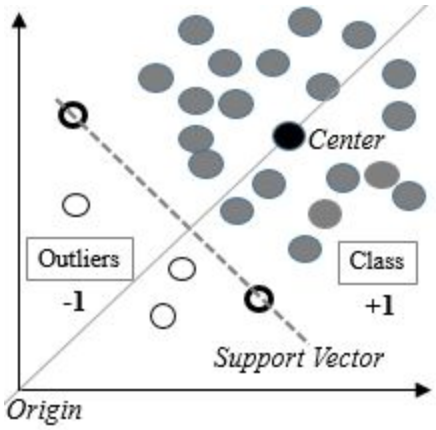


Figure 4.3.a [39] One-Class SVM outlier detection graph

OC SVM summarizes the properties of normal cases and from these, properties can predict which data points are unlike the normal points. This is useful for anomaly detection because the scarcity of training data is what defines anomalies, typically there are very few examples of the network intrusion, fraud, or other anomalous behavior[44].

Training can be achieved by treating a certain number of data points of the positive class as if they belong to the negative class. The idea is to define a boundary between the mximum of the positive data points and outliers. One-class SVMs use the parameter v(Nu) to define the trade-offs between the percentage of data points treated as the positive class and the negative class[34].

Two approaches to get this separating boundary area unit generally available:

The first approach to train a one-class SVM is a classification function to be described that hold to a hypersphere boundary between the normal and the outliers,
based on a density distribution function. The parameter v determines the shape of the boundary.
The second approach fits a hyperplane between the origin and the data points, separating a certain percentage of outliers from the rest of the data points. This approach has been shown to be equivalent to the decision hypersphere(which is formed by SVDD-support vector data description) and is used by many one-class SVM implementations due to its simpler implementation.

In this thesis we use the second approach, even in this approach it is easy to do with SVDD as there is no difference in separating the boundary.The error rate of a one-class SVM is calculable because the fraction of the positive class versus the fraction of outliers that is rejected. The density distribution of the fraction of outliers needs to be estimated to calculate this error measure.
There are many models that is used to identify Z The most simple of these is a Gaussian density model with a probability distribution as:

$$p_N(z;\mu,\Sigma)=\frac{1}{(2\pi)^{d/2}\left|\Sigma^{1/2}\right|}\exp\left\{-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)\right\}$$

where μ is a mean, Σ is covariance matrix, and d the dimension of an object. A notable distinction with boundary ways is that these ways yield multiple boundaries to hide a

given positive category, however the parameters for the quantity of distinct boundaries that need to be supplied[50].

As described previously, the parameter μ controls a trade off between the fraction of data points in the region and the ability of the decision function i.e the value is given based on how much outlier fraction we have, in our case the value is 0.028. The parameter gamma controls the nonlinear characteristics of the decision function and is auto-calculated. Therefore, both the parameters μ and gamma influence the performance of the OCSVM.

The figure 4.3.b is the complete explanation from the above to understand easily.

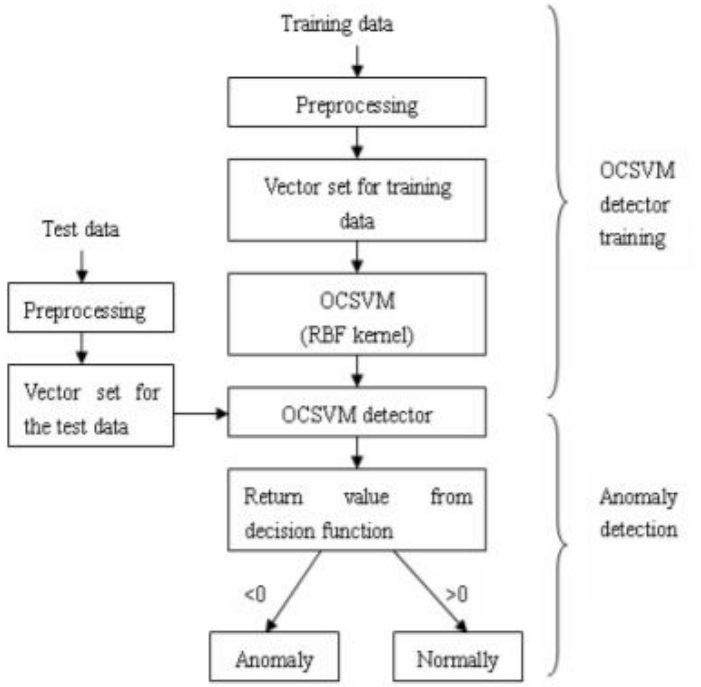


Figure 4.3.b OC SVM anomaly detection steps[46]

It is necessary to transfer the time series ino set of vectors. Different preprocessing procedures are adopted depending on the data.Since the detection results are depending on the counts, the set of vectors are processes and checked to make sure the values are close to the origin.

As RBF kernel is selected for the OC-SVM anomaly detection, we have already mentioned about the parameters nu and gamma, and how important these are because they influence the performance. Both the training set and the test set are preprocessed

to obtain the vector sets according to the various data types. The training vector set is then used to train the OC-SVM detector and the OCSVM detector is later applied on the test vector set. If the return value for the decision function f(x) is negative, an anomaly is detected else the data is normal.

5. Input Data

The input data that is considered in this thesis for the algorithms implementation with all the counts and the other required calculations are explained under this section.
The Input data is a simulated data logs collected in kibana past 1 month that is used in detecting anomalies.

5.1.Selected fields

Here we consider the important fields i.e @timestamp, user_id, client_ip address. The server_ip address is unique, so it can be avoided. The data that is in hand is not sufficient for the detection of anomalies using unsupervised learning, so client ip_address is generated as a new dimension and added in the real access data that is suggested by the company(iC-consultant) for the thesis. We randomly generated IP keeping the timestamp same, as there is only one ip_address available in the whole data logs, so it was easy to generate a fake ip_addresses and add randomly to the logs using python code. This one ip_ address is the source of many login attempts. It is still possible to see patterns of login attempts at an unusual time.

Therefore this client_ip address plays the key role, as the anomalies are detected using the current and history of the timestamp of each user. Later when the anomalies are detected the ip_address that is anomalous is shown as a result. By this process the company can strictly monitor these anomalous actions that is caused by the detected ip_addresses and take precautionary measures in the near future , as the process is adapted to different data sets with large different ip_addresses.

	@timestamp	_id	ip_address
0	July 8th 2019, 14:43:03.000	XswJ0msBoTGddM7vxMDB	10.1.2.312
1	July 8th 2019, 14:43:01.000	dKQJ0msB7mP0GwVzvJjz	10.1.2.271
2	July 8th 2019, 14:42:59.000	CcwJ0msBoTGddM7vtb8y	10.1.2.27
3	July 8th 2019, 14:42:57.000	bKQJ0msB7mP0GwVzrZdT	10.1.2.55
4	July 8th 2019, 14:42:55.000	L6QJ0msB7mP0GwVzpZeI	10.1.2.433

Figure 5.1.a. Filtered and selected data fields

- @timestamp:The timestamp that describes when the login happens.
- _id: The id field gives the unique identity of the users.
- ip_address: These ip_addresses are generated and are alloted to the users randomly.
  Total data that is used to detect anomalies is 721547 rows × 3 columns(timestamp, user_id and ip_address)

5.2.Feature Building

Feature building is the main step to extract new features from the data logs. From the Figure 5.2.a we can see there are new features that are extracted and then used in the future.

Firstly what we see is that the values are sorted according to the ip_address. This building is done for each unique ip_addresses, for eg:the figure 5.2.a gives the details about the ip_address 10.1.1.1 where the user_id's are from different users who are currently using this ip_address.

Secondly timestamp is considered for calculating the time_difference for each user_id by performing the subtraction of the previous observation from the current observation. From the below figure 5.2.a NaT/NaN beams that there was no previous history by the particular user_id.
Day of week(dow) is set for 6 days and sunday is the only day that is considered as weekend.
From the figure 5.2.a is_weekend is declared as "1" as the 9th of june is Sunday.

@timestamp	_id	ip_address	shift_time	time_diff	date	dow	hour	is_weekend	hour
2019-06-09 00:06:09	DBuOOWsB7mP0GwVzhZ9U	10.1.1.1	NaT	NaN	2019-06-09	6	0	1	0
2019-06-09 01:28:39	bB7aOWsB7mP0GwVzDY5G	10.1.1.1	2019-06-09 00:06:09	82.0	2019-06-09	6	1	1	0
2019-06-09 03:12:49	R0w5OmsBoTGddM7vayZT	10.1.1.1	2019-06-09 01:28:39	104.0	2019-06-09	6	3	1	0
2019-06-09 03:13:45	U0w6OmsBoTGddM7vRi8R	10.1.1.1	2019-06-09 03:12:49	0.0	2019-06-09	6	3	1	0
2019-06-09 03:42:39	z01UOmsBoTGddM7vuzyC	10.1.1.1	2019-06-09 03:13:45	28.0	2019-06-09	6	3	1	0

Figure 5.2.a Feature building from data logs

5.3 IP-Profiling

The Median of daily counts are calculated to add some more features i.e the counts are obtained by listing the numbers in the set in ascending order and locating the middle count in the list. These values are considered and counted as the daily_counts of the ip_address.

	ip_address	daily_counts
0	10.1.1.1	40.0
1	10.1.1.100	78.0
2	10.1.1.101	40.0
3	10.1.1.106	35.5
4	10.1.1.109	42.5
5	10.1.1.110	41.0

Figure:5.3.a Calculation of daily_counts

Total Count

Similar to the daily_counts the total counts of each ip_addresses are calculated using the timestamp of all the users who are under that particular ip_address. All these calculations are merged into the feature set in the next session.

	ip_address	total_count
0	10.1.1.1	1446
1	10.1.1.100	2860
2	10.1.1.101	1465
3	10.1.1.106	1408
4	10.1.1.109	1459
5	10.1.1.110	1482

Figure:5.3.b Calculating the total_count of ip_addresses

Average Login time

The average login time of each ip_address is calculated to use it in the future. Here the mean of each ip_address i.e it sums up all the values of the user_id's and divides them by the number of values to get the mean "td_mean".

Also calculated the maximum login time "td_max". These are the values from the data logs that are saved and tested over a month. When it comes to the monitoring of the year data logs then the values would be high in the calculation and the accuracy increases in detecting anomalous activity.

	ip_address	td_mean	td_max
0	10.1.1.1	28.999308	362.0
1	10.1.1.100	14.427072	185.0
2	10.1.1.101	28.520492	211.0
3	10.1.1.106	29.771144	319.0
4	10.1.1.109	28.711934	278.0
5	10.1.1.110	28.249831	240.0
6	10.1.1.114	29.827169	300.0
7	10.1.1.118	28.976471	267.0

Figure:5.3.c Mean and Maximum values of the ip_addresses

5.4 Full Feature set

Here comes the full feature set of the data logs to continue with the main algorithms and the clusters. As mentioned above, the figure 5.4.a shows the complete set of the features that are

considered for the detection process. Now this has all the required featured by merging all the above figures.

	ip_address	total_count	daily_counts	is_weekend_ratio	td_mean	td_max
0	10.1.1.1	1446	40.0	2.070064	28.999308	362.0
1	10.1.1.100	2860	78.0	2.177778	14.427072	185.0
2	10.1.1.101	1465	40.0	2.191721	28.520492	211.0
3	10.1.1.106	1408	35.5	2.229358	29.771144	319.0
4	10.1.1.109	1459	42.5	2.206593	28.711934	278.0
5	10.1.1.110	1482	41.0	2.242888	28.249831	240.0
6	10.1.1.114	1407	37.0	2.140625	29.827169	300.0
7	10.1.1.118	1446	42.0	2.308924	28.976471	267.0
8	10.1.1.119	1440	42.0	2.257919	29.087561	232.0
9	10.1.1.12	1392	40.0	2.200000	30.152408	250.0

Figure:5.4.a Complete features table.

6.Algorithm Implementation

The algorithms are implemented considering the section 5 where the input data is explained in detail. The data that is filtered, selecting the field and calculating the total counts and other features that is done to make the implementation easy and accurate is used in section 6.

6.1 K-means Implementation

Cluster analysis is recognized as an important technique for classifying data, finding clusters of a dataset based on similarities in the same cluster and dissimilarities between different clusters. Putting each point of the dataset to exactly one cluster is the basic of the conventional clustering method whereas clustering algorithm actually separates unlabelled set of data into different groups according to the similarity. Clustering is basically considered as classification of similar objects, it is precisely partitioning of datasets into clusters so that data in each cluster shares some common trait [54]. Now the Kmeans which is considered to be the main clustering algorithm is implemented based on the clusters as per figure 6.1.2.

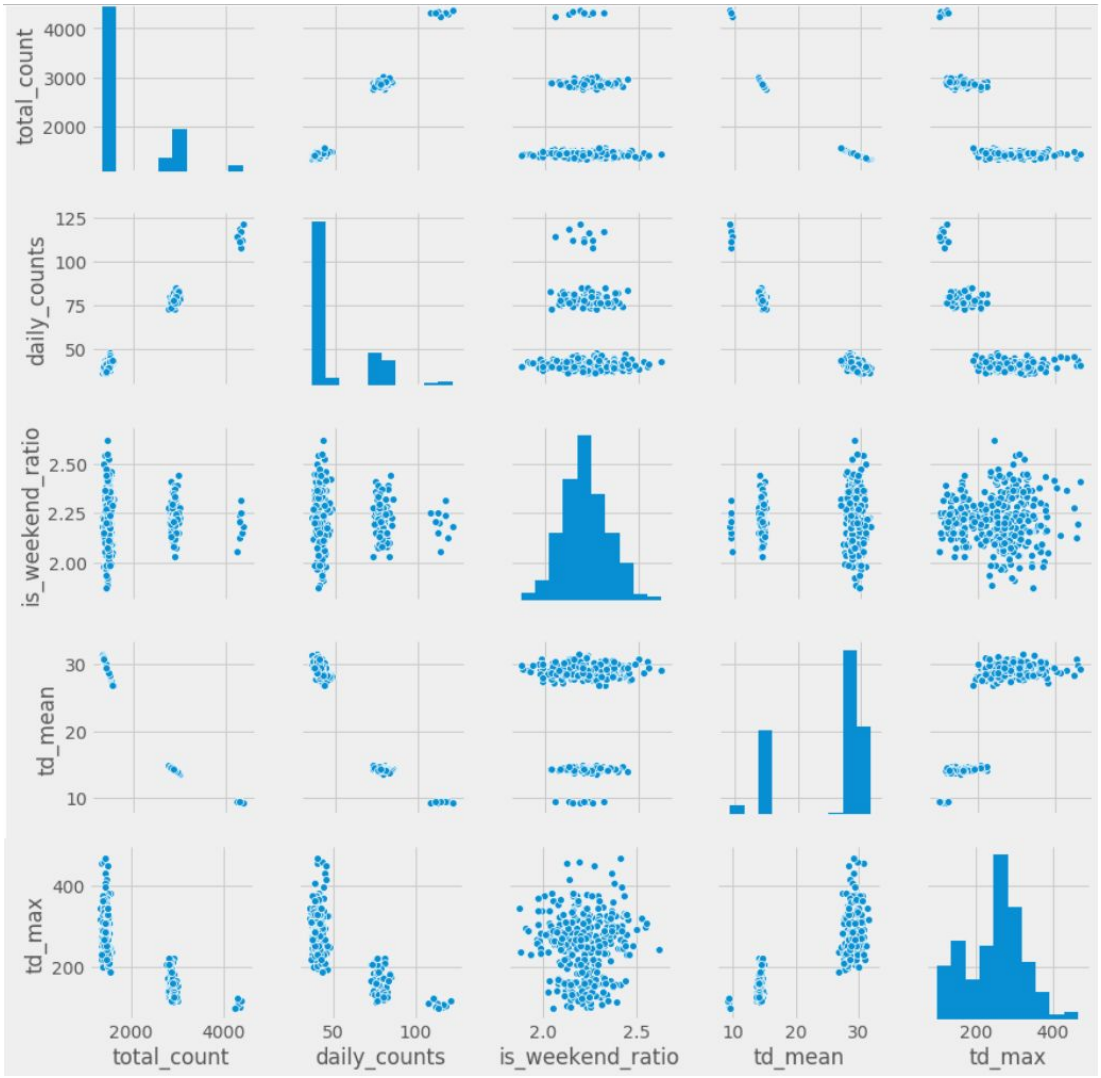


Figure 6.1.a Actual data set graph representation

It is a pairwise plot of the features. This graph is used to understand the relationship between the variables. While most relationships are looking like clusters, the td_max vs is_weekend_ratio is a giant blob. Hence we use t-SNE instead of PCA to explain this graphs, as t-SNE is good at understanding non linear relationships.

This is a normal graph, Ip_addresses have discrete values, so we have different blobs. We are sampling from a set of ip's, so some of the ip_addresses have very high counts, some medium and some very less. That is the reason we have small blobs like a smooth normal graph with long tails. In case there were an infinite number of ip_addresses and if it was our choice then the graph would have looked smoother with equal distribution.

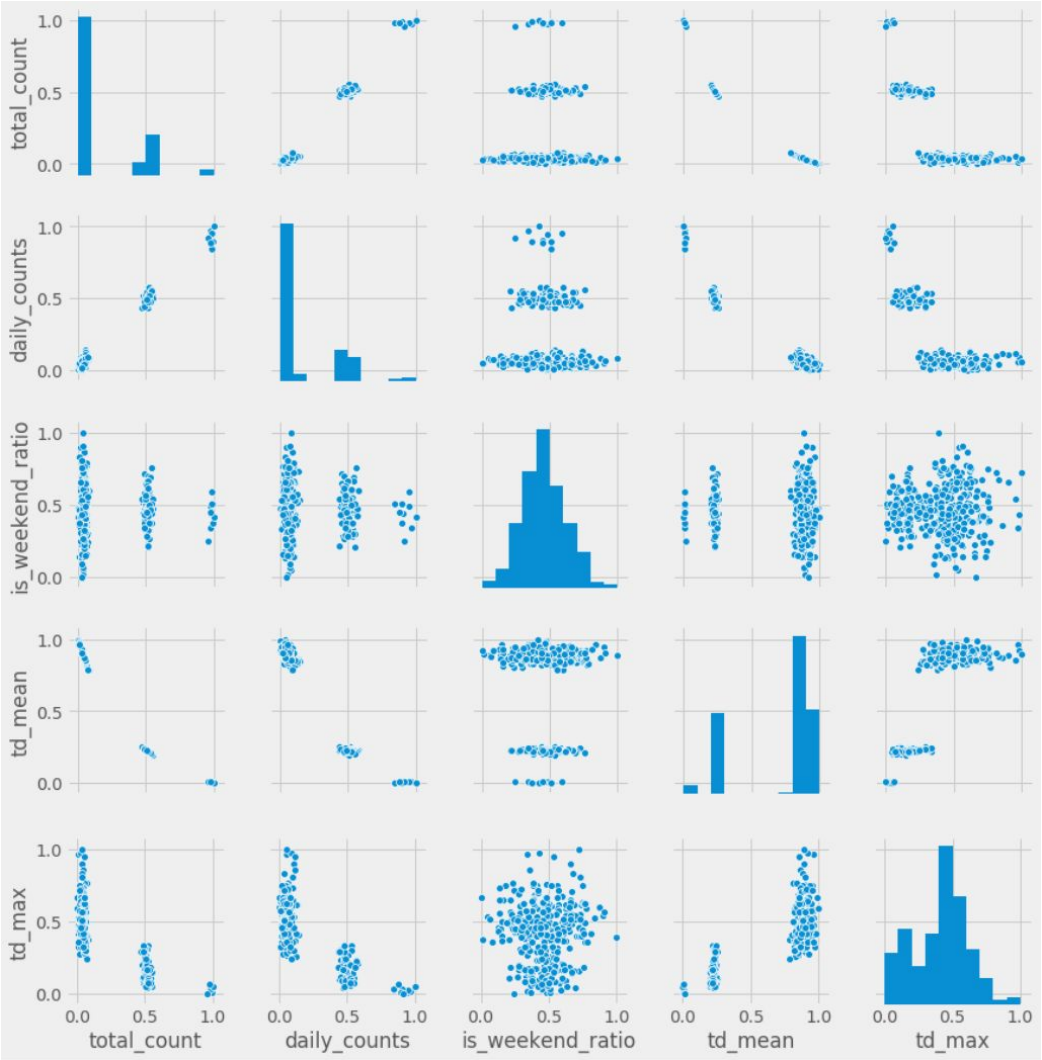


Figure 6.1.b New_data graph showing scaled data.

As shown in Figure 6.1.a, The old data is the actual data. The new_data in figure 6.1.b is the data we get post scaling the data. You can see a MinMaxScaler in the code. It just changes the scale of the data so the calculations are easier. All the clustering was done on actual feature set, but t-SNE was not used here, when the graphs or the blobs need to be shown in order to understand it well we use t-SNE. t-SNE was not used to do any computation, the main purpose was to visualize the blobs.

6.1.1 Elbow Curve

Before we start k-means clustering, we use elbow method to determine the optimal number of clusters. choose a number of clusters so that adding another cluster doesn't give much better modeling of the data[40].

The Elbow curve is something like an interpretation and validation of consistency within cluster analysis, this is mainly used to find the appropriate number of clusters in a dataset.
From the below elbow curve, we see that the graph levels off after 6 clusters, implying that addition of more clusters do not explain much more of the variance in our relevant variable.
The number of clusters chosen is therefore 6[40].

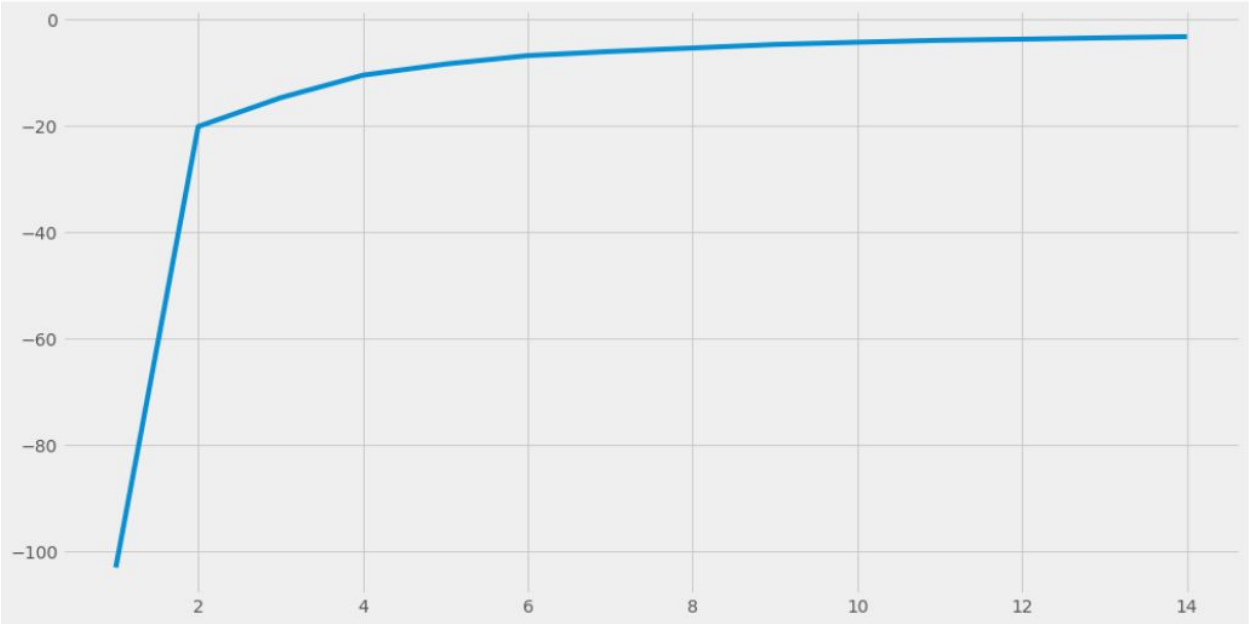


Figure 6.1.1.a: Elbow cure for finding number of clusters

```
1    103
2     94
4     73
3     60
5     46
0     10
Name: cluster, dtype: int64
```

6.1.2 Clustering model with t-SNE

t-SNE- plots are mainly used for creating a graph that has a high dimensional data logs and reduces into low dimensional data points but has a lot of the original information in it. This is just to reduce the same feature points and to convert the correlations in the data logs into a 2-D

graph as shown below in figure 6.1.2.a. Data points that are highly correlated are clustered together.

The axis in the 2-D graph are not named by our choice, but it is done by T-sne according to their importance. That is the reason to leave the axis named as t-SNE-2d-one and t-SNE--2d-two. The differences along the t-sne-2d-one are more important compared to t-sne-2d-two.

Six different colour palettes are used to differentiate between each cluster, as there are 6 clusters which is shown in figure 6.1.2.a. and the values of each cluster in given in the elbow curve, as it is also shown that the cluster 0 is having least of 10 data points forming it into a cluster, so it is the smallest cluster among others. Similarly with the other clusters, cluster 1 with highest of 103 and so on.

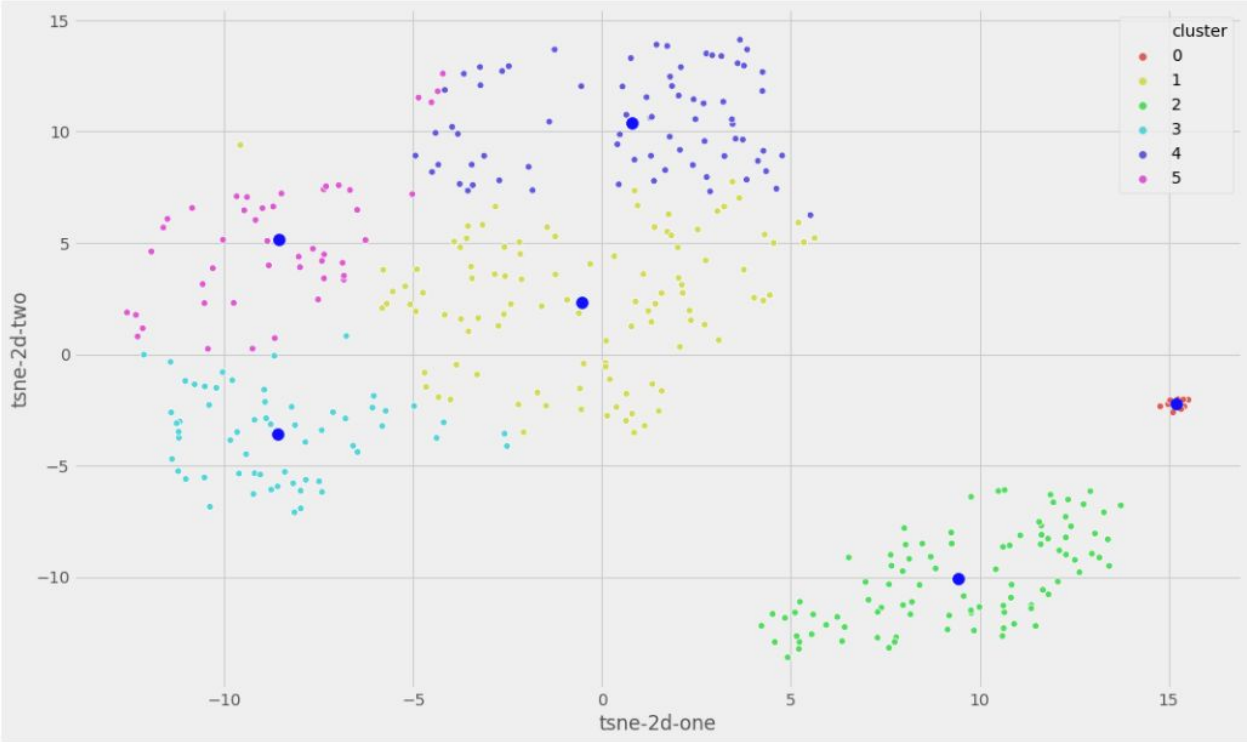


Figure 6.1.2.a t-SNE 2d cluster model.

Now we need to know “WHY” 6 clusters when there is a possibility of 4 when we look into the elbow curve. The important reason is 6 clusters makes more sense than 4 clusters, when we tried with 4 clusters, the distance from each cluster will be closer, the calculation of data points to the centroids and representing it in a bar graph will be screwed to select the appropriate cutoff point.

Kmeans has the below computations before plotting the cluster graph.
Computing 121 nearest neighbors...
Indexed 386 samples in 0.000s...
Computed neighbors for 386 samples in 0.008s...
Computed conditional probabilities for sample 386 / 386
Mean sigma: 0.077308
KL divergence after 250 iterations with early exaggeration: 56.693813
KL divergence after 300 iterations: 0.342785

6.1.3. Bar graph of cluster_model.

In this sub-session the ‘sum of square distance’ (ssd)is calculated and plotted in the graph. Each ip_address’s distance as per timestamp is calculated and the values are added to the graph as shown in figure 6.1.3.a. The distance that is near to the coordinate 0 are considered as the normal behaviour and here far from it are anomalous.

cutoff = 6

The cutoff ratio is considered by knowing the data and depends on how sensitive the algorithm must behave in detecting anomalies. Here I am considering the cutoff point at 6 and applying the algorithm. So considering SSd is greater 6 are anomalies gives better result for Kmeans, when taking cutoff point 8, the result is same as isolated forest. So we want k-means algorithm to be very stringent which is clear distinction.

This is by viewing the Figure 5.3.c and its distance from the coordinate.

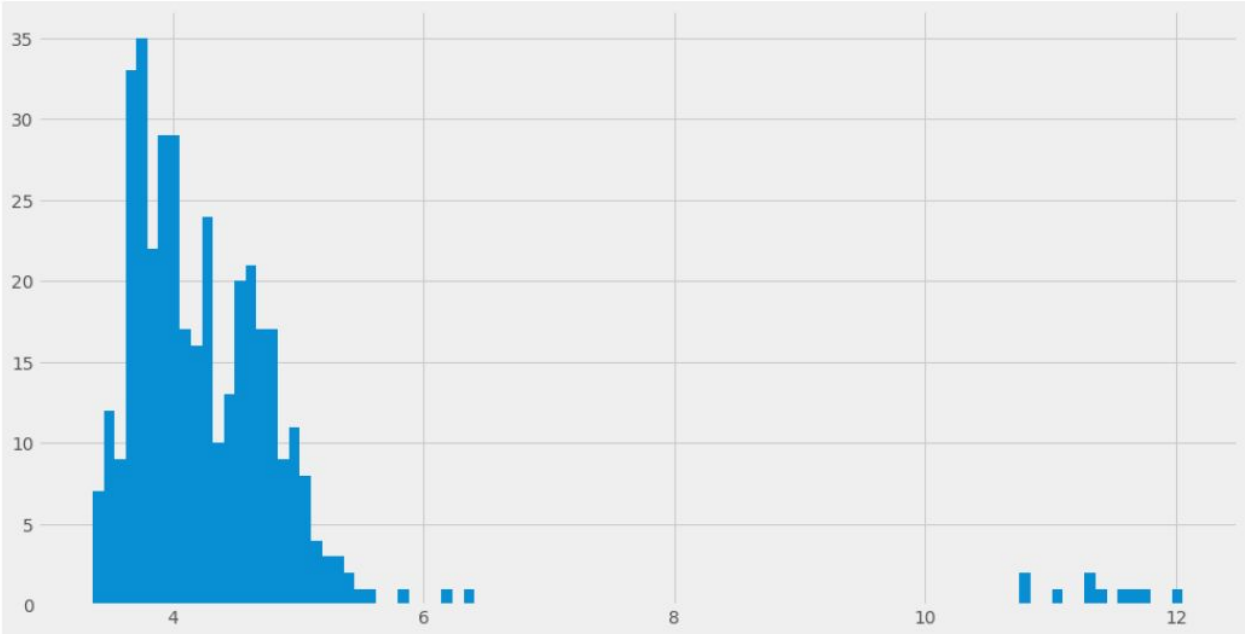


Figure 6.1.3.a: SSD cluster_model and its cutoff point.

6.1.4 Kmeans Agorithm

After analysing and visualizing all the data logs we step into the main process of implementing the algorithm. As we know the above, having elbow curve which was used to determine the number of clusters, clustering graph to represent the clusters with its centroids which was an important step that shows how to calculate SSD and the bar graph that is important for selecting the cutoff point is the basic and very important steps to implement Kmeans algorithm.

K-mean clustering algorithm in order to differentiate time intervals with normal and anomalous data points in the training dataset. The resulting cluster centroids are then used for fast anomaly detection in new monitoring data.In the next subsection, we describe the raw data and the extracted features that serve as input for the unsupervised algorithm. Then we explain the K-mean clustering algorithm and the resultings in a t-SNE graph.

A distance function is necessary to compute the distance between two objects. The most commonly used distance function is the Euclidean one which is defined as[37]:

SSD(i) = $\sqrt{\sum_{i=1}^{k} (xi - Cx(k))^2 + (yi - Cy(k))^2}$

x = (x1,...,xm) and y = (y1,...,ym) are two input parameters with m quantitative features. In the Euclidean distance function, all features contribute equally to the function value. However, since different features are usually measured with different metrics or at different scales, they must be normalized before applying the distance function[38].

The underlying assumption in the clustering based anomaly detection is that if we cluster the data, normal data will belong to clusters while anomalies belong to small clusters. We use the following steps to find and visualize anomalies.

- Choose the number of clusters
- Then the initialization of the centroid
- Each data point is assigned to a cluster with closest centroid
- Means of each cluster is calculated to be its new centroid
- After recomputing the centroid, Sum of square distance is calculated
- considering the cutoff point at 6 and applying the algorithm. So considering SSD that is greater than 6 are anomalies gives better result for Kmeans

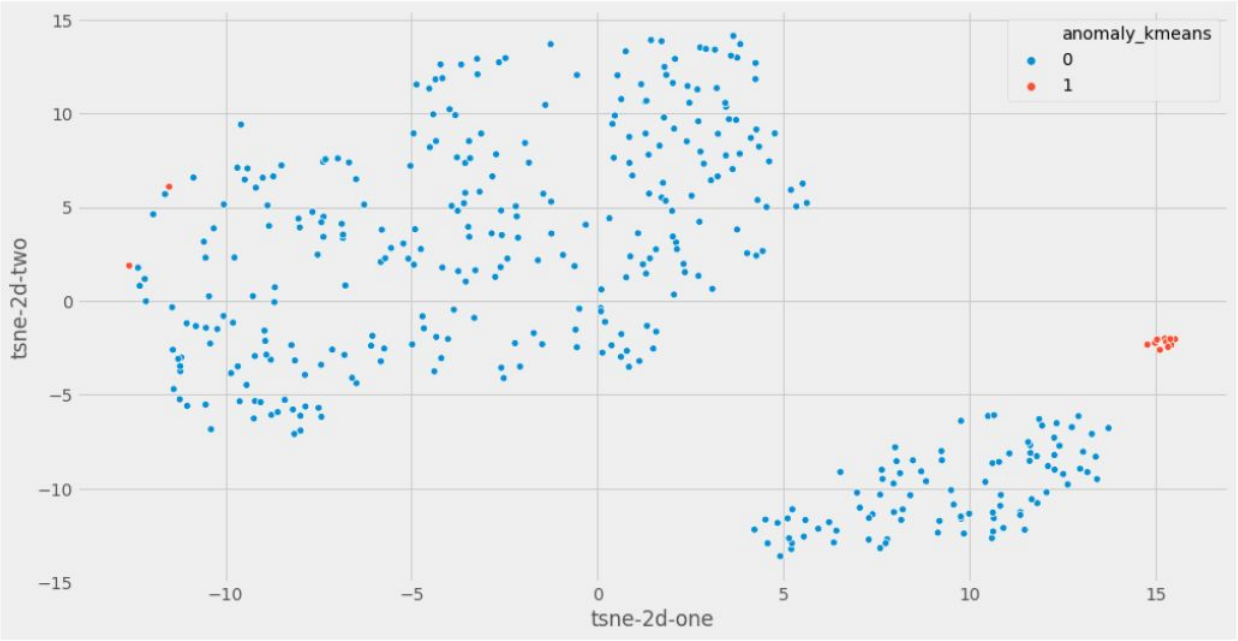


Figure 6.1.4.a K-means 2d graph showing anomalies

Finally, we see from figure 6.1.4.a how the pattern scan be used for classification and outlier detection.

The k-means result of anomaly contains the above method Cluster (0:normal, 1:anomaly).

For the given data, 375 data points are returned as normal and 12 data points are determined to be anomalies.

	ip_address	total_count	daily_counts	is_weekend_ratio	td_mean	td_max	cluster	tsne-2d-one	tsne-2d-two	ssd	anomaly_kmeans	anomaly_isolated	anomaly_svm	anomaly_manual
42	10.1.1.199	1365	40.5	2.123570	30.801320	455.0	5	-11.512229	6.081614	6.164888	1	0	1	0
62	10.1.1.249	4301	116.5	2.236268	9.459535	101.0	0	15.251910	-2.014084	11.541074	1	1	1	1
118	10.1.1.386	4300	118.5	2.127273	9.453361	104.0	0	15.528737	-2.041709	11.779482	1	1	1	1
164	10.1.1.486	4317	117.0	2.315668	9.417285	108.0	0	14.781444	-2.334249	11.659370	1	1	1	1
177	10.1.1.63	4339	112.0	2.148766	9.368142	101.0	0	15.287605	-2.182364	11.319226	1	1	1	1
188	10.1.1.86	4293	113.0	2.203731	9.456897	110.0	0	15.423254	-2.342858	11.090154	1	1	1	1
255	10.1.2.249	4353	112.0	2.250934	9.332721	102.0	0	14.986474	-2.238024	11.318310	1	1	1	1
311	10.1.2.386	4326	108.0	2.250188	9.392370	110.0	0	15.119220	-2.613225	10.791042	1	1	1	1
331	10.1.2.432	1437	40.5	2.413302	29.176880	466.0	5	-12.583876	1.878566	6.357696	1	0	1	0
357	10.1.2.486	4251	114.0	2.056075	9.571059	99.0	0	15.041411	-2.061416	11.438757	1	1	1	1
370	10.1.2.63	4372	121.0	2.184268	9.268588	118.0	0	15.395945	-2.032638	12.054720	1	1	1	1
381	10.1.2.86	4307	111.0	2.209389	9.441013	122.0	0	15.335781	-2.458637	10.825228	1	1	1	1

Figure 6.1.4.b Ip_addresses that are anomalous from k-means implementation

From the figure 6.1.4.b we can see all the ip_address that are anomalous in behaviour and are detected by applying the k-means algorithm to the selected and filtered data.

6.2. Isolation Forest implementation:

The detection of anomalies in isolation is implemented without employing any distance or density measure. This method is fundamentally different from clustering based or distance based algorithms as described in section 4.2.

Isolation forest is implemented in this thesis using the scikit-learn (sklearn) python libraries. Given the input data set and certain parameters, the sklearn ensemble method IsolatedForest can be used to isolate the outliers, which are in our case the anomalies. The method fit() is used to fit a model to the normalised input data set. The parameter number of estimators is set to 200 and the contamination is set to 0.028, based on initial outlier guess in the data. Further, the predict() method is used to detect if a given data is an outlier or not.

When applying an IsolationForest model, we set contamination = outliers_fraction, that is telling the model that the proportion of outliers in the data set is 0.028.

fit and predict(data) performs outlier detection on data, and returns 0 for normal, 1 for anomaly.

Finally, we visualize anomalies with t-SNE 2d graph, from figure 5.4.a how the pattern scan be used for classification and outlier detection.

The method returns 0 for normal and 1 in case of outlier or anomaly. For the given data, 375 data points are returned as normal and 11 data points are determined to be anomalies.

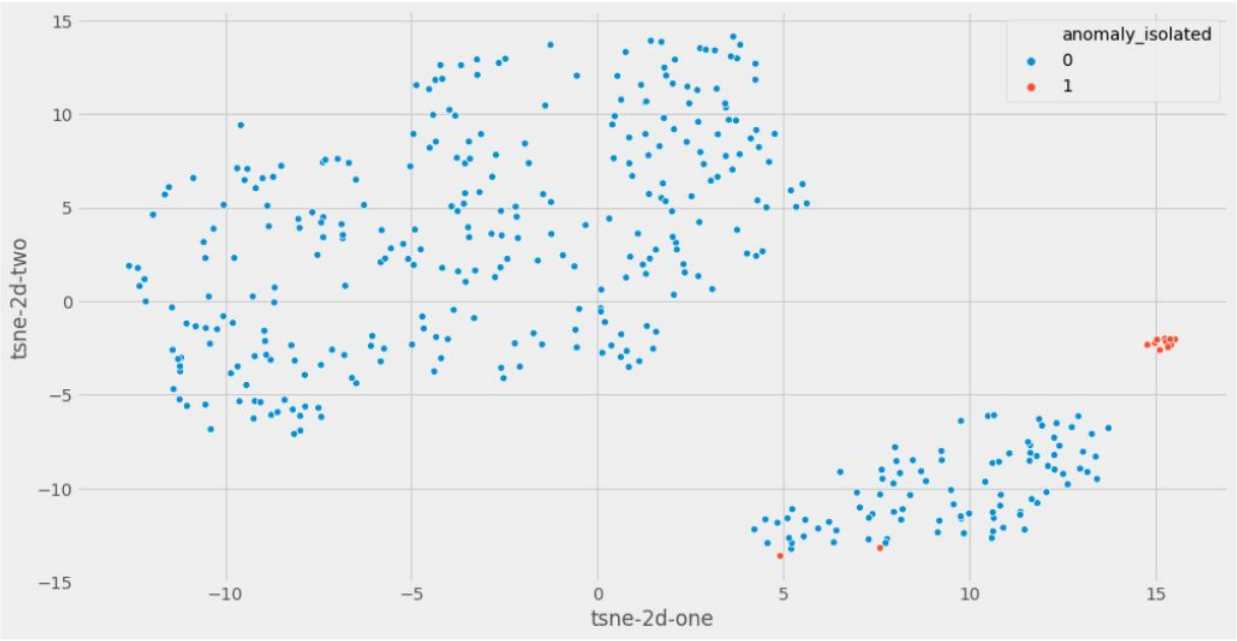


Figure 6.2.a Anomalies detected using isolated forest

From the figure 6.2.b we can see all the ip_address that are anomalous in behaviour and are detected by applying the isolated forest algorithm to the selected and filtered data.

	ip_address	total_count	daily_counts	is_weekend_ratio	td_mean	td_max	cluster	tsne-2d-one	tsne-2d-two	ssd	anomaly_kmeans	anomaly_isolated	anomaly_svm	anomaly_manual
62	10.1.1.249	4301	116.5	2.236268	9.459535	101.0	0	15.251910	-2.014084	11.541074	1	1	1	1
118	10.1.1.386	4300	118.5	2.127273	9.453361	104.0	0	15.528737	-2.041709	11.779482	1	1	1	1
164	10.1.1.486	4317	117.0	2.315668	9.417285	108.0	0	14.781444	-2.334249	11.659370	1	1	1	1
177	10.1.1.63	4339	112.0	2.148766	9.368142	101.0	0	15.287605	-2.182364	11.319226	1	1	1	1
188	10.1.1.86	4293	113.0	2.203731	9.456897	110.0	0	15.423254	-2.342858	11.090154	1	1	1	1
255	10.1.2.249	4353	112.0	2.250934	9.332721	102.0	0	14.986474	-2.238024	11.318310	1	1	1	1
311	10.1.2.386	4326	108.0	2.250188	9.392370	110.0	0	15.119220	-2.613225	10.791042	1	1	1	1
357	10.1.2.486	4251	114.0	2.056075	9.571059	99.0	0	15.041411	-2.061416	11.438757	1	1	1	1
361	10.1.2.497	2973	83.5	2.440972	13.869785	165.0	2	4.914829	-13.606808	5.452700	0	1	1	0
370	10.1.2.63	4372	121.0	2.184268	9.268588	118.0	0	15.395945	-2.032638	12.054720	1	1	1	1
381	10.1.2.86	4307	111.0	2.209389	9.441013	122.0	0	15.335781	-2.458637	10.825228	1	1	1	1

Figure 6.2.b Ip_addresses that are anomalous from isolation forest implementation

6.3 OC-SVM implementation:

The idea of SVM for anomaly detection is to find a function that is positive for regions with high density of points, and negative for small densities.

According to the one class svm we select one class to implement this algorithm and detect anomalies.

- When fitting OneClassSVM model, we set nu=outliers_fraction, which is an upper bound on the fraction of training errors and a lower bound of the fraction of support vectors, and must be between 0 and 1. Basically this means the proportion of outliers we expect in our data.
- Specifies the kernel type to be used in the algorithm as in our case its rbf. This will enable OC-SVM to use a nonlinear function to project the hyperspace to higher dimension.
- Gamma is a parameter of the RBF kernel type and controls the influence of individual training samples - this affects the "smoothness" of the model. Through experimentation, I did not find any significant difference.
- predict() perform classification on data, and because our model is an one-class model, +1 or 0 is returned, and 1 is anomaly, 0 is normal.

The default value of gamma will change from 'auto' to 'scale' in version 0.22 to account better for unscaled features. Set gamma explicitly to 'auto' or 'scale'.

From the below code snippet take all the data points where total count is less than 1-outlier fraction. From our point of view outlier fraction is nu=0.028, which gives the result of 0.97. So taking the 97th quantile of total count and all data points below it are normal. Assuming this as one class we fit out model on this and predict all the classes. This will identify the anomaly class.

```
data_new_clean = data_new.loc[data_new.total_count <=data_new.total_count.quantile(1-outlier_fraction)]
```

The method returns 0 for normal and 1 in case of outlier or anomaly.For the given data, 365 data points are returned as normal and 21 data points are determined to be anomalies.

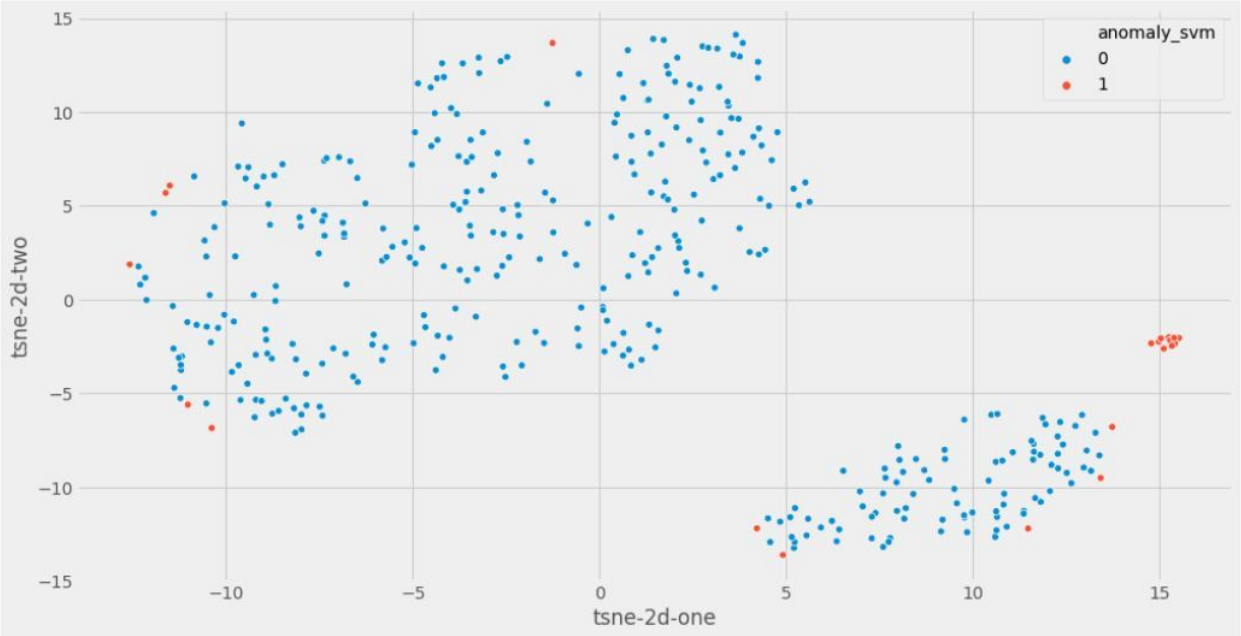


Figure 6.3.a Anomalies detected using One Class SVM

From the below figure 6.3.b we can see all the ip_address(21 ip_addresses) that are anomalous in behaviour and are detected by applying the OC SVM algorithm to the selected and filtered data.

	ip_address	total_count	daily_counts	is_weekend_ratio	td_mean	td_max	cluster	tsne-2d-one	tsne-2d-two	ssd	anomaly_kmeans	anomaly_isolated	anomaly_svm	anomaly_manual
14	10.1.1.138	1383	39.5	2.546154	30.306078	297.0	3	-11.028182	-5.598881	5.249748	0	0	1	0
42	10.1.1.199	1365	40.5	2.123570	30.801320	455.0	5	-11.512229	6.081614	6.164888	1	0	1	0
62	10.1.1.249	4301	116.5	2.236268	9.459535	101.0	0	15.251910	-2.014084	11.541074	1	1	1	1
70	10.1.1.264	1445	42.5	2.621554	29.042936	241.0	3	-10.385866	-6.843124	5.357686	0	0	1	0
118	10.1.1.386	4300	118.5	2.127273	9.453361	104.0	0	15.528737	-2.041709	11.779482	1	1	1	1
134	10.1.1.424	2990	80.0	2.225458	13.777852	117.0	2	11.483550	-12.199755	5.263552	0	0	1	0
164	10.1.1.486	4317	117.0	2.315668	9.417285	108.0	0	14.781444	-2.334249	11.659370	1	1	1	1
171	10.1.1.500	2946	80.0	2.147436	13.989134	120.0	2	13.428407	-9.499339	5.189349	0	0	1	0
177	10.1.1.63	4339	112.0	2.148766	9.368142	101.0	0	15.287605	-2.182364	11.319226	1	1	1	1
188	10.1.1.86	4293	113.0	2.203731	9.456897	110.0	0	15.423254	-2.342858	11.090154	1	1	1	1
189	10.1.1.89	2895	83.0	2.028243	14.259157	137.0	2	13.737984	-6.781340	5.349382	0	0	1	0
208	10.1.2.139	2898	76.0	2.393443	14.231619	115.0	2	4.219205	-12.188558	5.293503	0	0	1	0
253	10.1.2.243	1407	39.5	1.871429	29.747511	343.0	4	-1.257572	13.681190	5.593116	0	0	1	0
255	10.1.2.249	4353	112.0	2.250934	9.332721	102.0	0	14.986474	-2.238024	11.318310	1	1	1	1
286	10.1.2.323	1408	43.5	2.192744	29.719261	458.0	5	-11.622698	5.688149	5.825046	0	0	1	0
311	10.1.2.386	4326	108.0	2.250188	9.392370	110.0	0	15.119220	-2.613225	10.791042	1	1	1	1
331	10.1.2.432	1437	40.5	2.413302	29.176880	466.0	5	-12.583876	1.878566	6.357696	1	0	1	0
357	10.1.2.486	4251	114.0	2.056075	9.571059	99.0	0	15.041411	-2.061416	11.438757	1	1	1	1
361	10.1.2.497	2973	83.5	2.440972	13.869785	165.0	2	4.914829	-13.606808	5.452700	0	1	1	0
370	10.1.2.63	4372	121.0	2.184268	9.268588	118.0	0	15.395945	-2.032638	12.054720	1	1	1	1
381	10.1.2.86	4307	111.0	2.209389	9.441013	122.0	0	15.335781	-2.458637	10.825228	1	1	1	1

Figure 6.3.b Ip_addresses that are anomalous from OC-SVM implementation

7.Comparision and Result

As we are using the unsupervised learning and need to compare the chosen algorithms to identify the best algorithm that is accurate in detecting anomalies in the data logs. The first step that is done for the calculation purpose is that we need to manually label the data and see the performance of the individual algorithms.

- Criteria for manual labels is strictly based on counts as we want the algorithms to be robust on count anomalies

Two conditions were taken to of total counts and daily counts is greater than quantile(0.98), if either of them is satisfied it is an anomaly. Where ever it is true mark it as 0 or 1. From the figure 6.a we can see the anomaly coming from kmeans,Isolated Forest,OC-SVM and manual(what we manually identify). Given these are the labels and the right ones we calculate the scores.

	ip_address	total_count	daily_counts	is_weekend_ratio	td_mean	td_max	cluster	tsne-2d-one	tsne-2d-two	ssd	anomaly_kmeans	anomaly_isolated	anomaly_svm	anomaly_manual
0	10.1.1.1	1446	40.0	2.070064	28.999308	362.0	5	-4.352579	11.806170	4.570390	0	0	0	0
1	10.1.1.100	2860	78.0	2.177778	14.427072	185.0	2	9.245516	-8.002912	4.393522	0	0	0	0
2	10.1.1.101	1465	40.0	2.191721	28.520492	211.0	1	4.270270	2.411105	3.658624	0	0	0	0
3	10.1.1.106	1408	35.5	2.229358	29.771144	319.0	5	-7.361582	4.483963	4.265576	0	0	0	0
4	10.1.1.109	1459	42.5	2.206593	28.711934	278.0	1	-2.134053	3.368442	3.626661	0	0	0	0

Figure 7.a Data for comparison

Manual anomaly is nothing but used for performance check, to understand which algorithm is doing better. This is an unsupervised learning classification problem

The very first step is to understand the confusion matrix. From the Figure 7.b if we know what stands where then the calculations becomes easy.

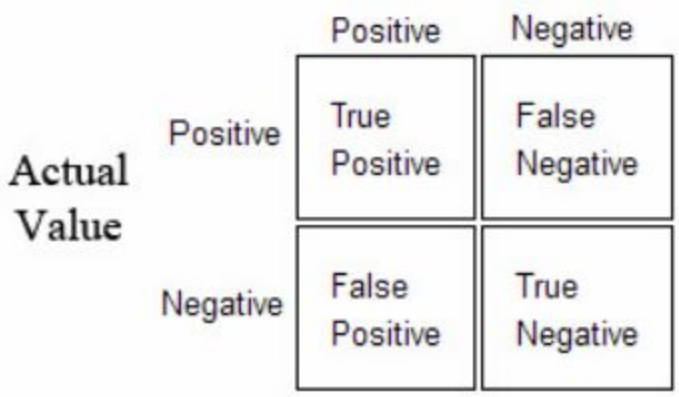


Figure 7.b Confusion matrix visualization[51]

A confusion matrix is a table used to describe the performance of a classification model on a set of data for which the true values are known.

True Positives (TP) - The value of actual class and the value of predicted class is true.
True Negative (TN) - That the value of actual class and value of predicted class is also Negative.
False Positives (FP) – When actual class is Negative and predicted class is Positive.
False Negatives (FN) – When actual class is Positive but predicted class as Negative.

	isolated_forest	kmeans	svm
accuracy	0.997409	0.994819	0.971503
f1_score	0.952381	0.909091	0.645161
roc_score	0.998670	0.997340	0.985372
sensitivity	1.000000	1.000000	1.000000
specificity	0.909091	0.833333	0.476190

Figure 7.c. The comparison scores of algorithms

Accuracy :

Accuracy is the most instinctive performance measure and a ratio of correctly predicted value to the total values. As we know, if we have high accuracy then our model is best. Accuracy is a good measure but only when you have symmetric datasets where values of FP and FN are almost the same. Therefore, in this thesis there are other parameters calculated to evaluate the performance of your model.

Accuracy = (TP+TN) / Total number of samples

Accuracy score:
The accuracy of the Isolated forest is 0.9974 then follows the Kmeans with very minute difference of 0.9948, where as OC_SVM is 0.97 which gives us the result with false positives i.e the normal behaviour which is identified as anomalous.

Roc_score:
Receiver Operating characteristic Curve (ROC) balance achieved between the true positive rate and false positive rate for a predictive model as x-axis and y-axis.
In general it is a plot of the false positive rate on the x-axis versus the true positive rate on the y-axis for a number of different threshold values between 0.0 and 1.0. In another way, it plots the false alarm value versus the actual value.

True Positive Rate = TP / (TP + FN)

False Positive Rate = FP / (FP + TN)

ROC score	Evaluation
$0.9 < AUC \leq 1.0$	Excellent
$0.8 < AUC \leq 0.9$	Good
$0.7 < AUC \leq 0.8$	Fair
$0.6 < AUC \leq 0.7$	Poor
$AUC \leq 0.6$	Fail

Figure 6.b ROC score and its Evaluation[55]

From the figure 7.c,As we have the evaluation of the ROC we can easily compare our scores in this.Iforest as usual stands with the high value of 0.9986, with very minute difference is the kmeans with value of 0.9973. With the unexpected roc score is the OC-SVM giving the value of 0.9853. Only in ROC the value of OC-SVM is very near to the other algorithm scores.

Sensitivity/Recall:

Sensitivity is correctly predicted positive observations to the total predicted positive values[52].

Sensitivity= TP/(TP+FN)

Sensitivity is 1.00 for all algorithms, the predicted value and the actual value are equal in our case so it is 1.

Specificity:
Specificity is nothing but a true negative value[52].

Specificity=TN/(TN+FP)

If the specificity score is high then identified anomalies are correct, there is no normal data that is considered as anomalous data. The specificity of the Isolated algorithm is high with the score of 0.909, then follows the kmeans with 0.833 and there are a lot of false predictions in oc-svm as the specificity score is 0.476 which is not good.

F1 score:

F1 score means that you have low false positives and low false negatives in the detection. When there is small doubt with other measures calculated,F1 stands the best calculator which balances the precision and recall(sensitivity).

Now precision calculation is important for F1 score
Precision=TP/(TP+FP)

F1=2* ((precision*recall) / (precision+recall))

Difference between F1 and accuracy is accuracy can be largely contributed by a large number of True Negative whereas F1 Score might be a better measure to use if there is a large number of Actual Negatives.[53]

F1 score for iForest is 0.952 which is pretty good score and then follows the Kmeans with 0.909 and then the least is the OC-SVM with 0.645.

From the above scores, we can clearly see that the isolated forest serves the best at detecting anomalies where all the measures gives the clear view with high calculated scores.

Then in the list is the Kmeans with minute difference in the scores. Where as the OC-svm which is known for its anomaly detection is showing the false prediction which means the normal data is falsely predicted as the anomaly.

8.Summary and Outlook

The goal of this master thesis was to detect the anomalies using machine learning techniques (learning method selection depending on the data that is available in hand). The thesis was broadly divided into research and the implementation. This final chapter is concluding the thesis, highlight some of the useful lessons learned throughout the process and future improvements that are possible with the extension of this thesis.

8.1 Theory Side

The thesis started with understanding and research of the web application. This included inserting audit logs and collecting data logs in kibana. After the keen observation of how the application and with collecting the information of the company,

the decision of DDoS attack vector was carried out. A clear understanding of the technologies and the relation between them was a very important phase to implement and get the best results.
Use of these anomaly detection algorithms in any kind of companies that needs to detect attacks in advance can be a game-changer, especially in environments where a huge dataset is involved.

There are different approaches to anomaly detection. The approaches depends on the type of data that is in hand, whether its a labeled data or an unlabeled data, and also on the real time system or the simulated data. This resulted in this thesis to select an unsupervised method with the particular above selected algorithms.

But they lack in providing the spontaneous result to the user as this is using the filtered simulated data logs. The same code base can be run on multiple platforms. Current algorithms are written in Python, there had been a small research done on this selection and as mentioned Python is considered to be the best language when compared to R in detecting anomalies.

Bibliography

[1].https://www.ic-consult.com/en-UK/access-management.html
[2]S. Axelsson. Research in intrusion-detection systems: a survey. Department of Computer Engineering, Chalmers University of Technology,
Goteborg, Sweden, Technical Report. , 1998.
[3] Eleazar Eskin, Leonid Portnoy and Salvatoer J. Stolfo. Intrusion Detection with Unlabeled data using clustering. In proceedings of ACM CSS Workshop on Data Mining applied to Security(DMSA-2001), Philadelphia, PA,2001.
[4] Chandola, V., Banerjee, A. and Kumar, V.,. Anomaly detection: A

survey. ACM computing surveys (CSUR), 2009.
[5] Guobing,Z.,Cuixia,Z.and Shanshan,s.A Mixed Unsupervised Clustering-based Intrusion Detection Model. Third International Conference on Genetic and Evolutionary Computing. 2009.
[6]"ForgeRock Redefines Digital Identity Management with Enhanced Capabilities for The Internet of Things". MarketWired. April 3, 2017.
[7]."Company Overview of ForgeRock AS". Bloomberg. August 15, 2016. Retrieved August 15, 2016.
[8]https://www.volico.com/understanding-clustering-servers-capabilities
[9]Introduction to Semi-Supervised Learning, XiaojinZhu and Andrew B.Goldberg Synthesis Lectures on Artificial Intelligence and Machine Learning, 2009, Vol. 3, No. 1 , Pages 1-130 (https://doi.org/10.2200/S00196ED1V01Y200906AIM006)
[10]Fazakis, Nikos; Karlos, Stamatis; Kotsiantis, Sotiris; Sgarbas, Kyriakos (2015-12-29). "Self-Trained LMT for Semisupervised Learning". *Computational Intelligence and Neuroscience*. 2016: 3057481. doi:10.1155/2016/3057481. PMC 4709606. PMID 26839531.
[11]Alexander hinneburg and Daniel A.Kim.Clustering methods for large databases.from the past to the future.In Alex delis,Christosn falousos,and shahram ghandeharizadeh editors.sIGMOD 1999,proceeding ACM SIGMOD International conference on management of data,june1-3,1999.
[12]Data Mining: Practical Machine Learning Tools and Techniques. Authors:Ian H. Witten, Eibe Frank, Mark A. Hall. Publication: · Book Data Mining: Practical Machine Learning Tools and TechniquesISBN:0123748569 9780123748560

[13]Hussain, Hanaa. (2012). Dynamically and Partially Reconfigurable Hardware Architectures for High Performance Microarray Bioinformatics Data Analysis.
[14]Boyd S.W., Keromytis A.D. (2004) SQLrand: Preventing SQL Injection Attacks. In: Jakobsson M., Yung M., Zhou J. (eds) Applied Cryptography and Network Security. ACNS 2004. Lecture Notes in Computer Science, vol 3089. Springer, Berlin, Heidelberg
[15] Mahjabin, T., Xiao, Y., Sun, G., & Jiang, W. (2017). A survey of distributed denial-of-service attack, prevention, and mitigation techniques. International Journal of Distributed Sensor Networks. https://doi.org/10.1177/1550147717741463
[16]Journal, I. J. C. S. M. C., & Tandon, G. (2019). Comparative Analysis of Detection of DDOS Attack in WSN. IJCSMC, 8(5), 155–163.
[17]Hussain, Hanaa & Benkrid, Khaled & Ebrahim, Ali & Erdogan, Ahmet & Seker, Huseyin. (2012). Novel Dynamic Partial Reconfiguration Implementation of K-Means Clustering on FPGAs: Comparative Results with GPPs and GPUs. International Journal of Reconfigurable Computing. 2012. 10.1155/2012/135926.

[18]Modified Centroid Selection Method of K-Means Clustering,farit afendi,rose mawati,Bogor University,IOSR Journal of Mathematics (IOSR-JM) e-ISSN: 2278-3008, p-ISSN:2319-7676. Volume 10, Issue 2 Ver. III (Mar-Apr. 2014), PP 49-53 jhttps://www.academia.edu/25798184/Modified_Centroid_Selection_Method_of_K-Means_Clustering
[19] Mahjabin, T., Xiao, Y., Sun, G., & Jiang, W. (2017). A survey of distributed denial-of-service attack, prevention, and mitigation techniques. International Journal of Distributed Sensor Networks. https://doi.org/10.1177/1550147717741463
[20] Kai Ming Ting,Tharindu R. Bandaragoda, David Albrecht, Fei Tony Liu,and Jonathan R. Wells, Efficient Anomaly Detection byIsolation Using Nearest Neighbour Ensemble[2014],School of Information Technology, Monash University, Victoria, Australia.
[21]F. T. Liu, K. M. Ting, and Z. H. Zhou, “Isolation Forest,” in Proceedings of the 2008 8th IEEE International Conference on Data Mining. IEEEComputer Society, 2008, pp.413–422.
[22]M. Ester, H. P. Kriegel, J. S, and X. Xu, “A density-based algorithmfor discovering clusters in large spatial databases with noise,” in Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining. AAAI Press, 1996, pp. 226–231.
[23]F. Angiulli and C. Pizzuti, “Fast Outlier Detection in High DimensionalSpaces,” in Proceedings of the 6th European Conference on Principlesof Data Mining and Knowledge Discovery. Springer-Verlag, 2002, pp.15–26.
[24]S. Papadimitriou, H. Kitagawa, P. B. Gibbons, and C. Faloutsos, “LOCI:Fast Outlier Detection Using the Local Correlation Integral,” in Data Engineering, 2003. Proceedings. 19th International Conference on,March 2003, pp. 315–326.
[25]Gulshan Kumar (2016) Denial of service attacks – an updated perspective, Systems Science & Control Engineering, 4:1, 285-294, DOI: 10.1080/21642583.2016.1241193
[26]Goldstein M, Uchida S (2016) A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data. PLoS ONE 11(4): e0152173. https://doi.org/10.1371/journal.pone.0152173
[27]Advanced Computational Intelligence: An International Journal (ACII), Vol.2, No.2, April 2015 27 Comparison and Implementation Of Random4 Algorithm And Hirschberg Algorithm Using OpenSource Software For Prevention Of SQL Injection Attack, Mohammed Ahmed,Sayyed Saima,Shaikh Shagufta,Shaikh Tazreen (Department of Computer Engineering) M.H. Saboo Siddik College of Engineering, Clare Road, Byculla, Mumbai-400008, India,Vol.2, No.2, April 2015
[28]Ming-Syan Chen, Jong Soo Park and P. S. Yu, "Efficient data mining for path traversal patterns," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 10, no. 2, pp. 209-221, March-April 1998.doi: 10.1109/69.683753

URL:
http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=683753&isnumber=15039
[29]Vaithyasubramanian, Subramanian & Christy, A & Saravanan, Dhanavel. (2014). An analysis of Markov password against brute force attack for effective web applications. Applied Mathematical Sciences. 8. 5823-5830. 10.12988/ams.2014.47579.
[30]Padmaja K., Nageswara Rao K., Murthy J.V.R. (2014) Defending Approach against Forceful Browsing in Web Applications. In: Satapathy S., Avadhani P., Udgata S., Lakshminarayana S. (eds) ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India- Vol II. Advances in Intelligent Systems and Computing, vol 249. Springer, Cham.
[31] Information and Software Technology Volume 91, November 2017, Pages 186-197 An anomaly detection system based on variable N-gram features and one-class SVM WaelKhreich ,babakKhosravifar, AbdelwahabHamou-Lhadj, ChamseddineTalhi
https://doi.org/10.1016/j.infsof.2017.07.009
[32]During the second half of 2007, 11,253 site-specific cross-site vulnerabilities were documented by XSSed, compared to 2,134 "traditional" vulnerabilities documented by Symantec, in "Symantec Internet Security Threat Report: Trends for July–December 2007 (Executive Summary)" (PDF). **XIII**.
[33]Baddar, Sherenaz & Merlo, Alessio & Migliardi, Mauro. (2014). Anomaly Detection in Computer Networks: A State-of-the-Art Review. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA). 5. 29-64.
[34]Schölkopf, Bernhard & Platt, John & Shawe-Taylor, John & Smola, Alexander & C. Williamson, Robert. (2001). Estimating Support of a High-Dimensional Distribution. Neural Computation. 13. 1443-1471. 10.1162/089976601750264965.
[35]Liao, H.; Lin, C.R.; Lin, Y.; Tung, K. Intrusion detection system: A comprehensive review. J. Netw. Comput. Appl. 2013, 36, 16–24.
[36]Modi, C.; Patel, D.; Borisaniya, B.; Patel, H.; Patel, A.; Rajarajan, M. A survey of intrusion detection techniques in cloud. J. Netw. Comput. Appl. 2013, 36, 42–57.
[37] Traffic Anomaly Detection Using K-MeansClustering
Gerhard M¨unz, Sa Li, Georg Carle Computer Networks and InternetWilhelm Schickard Institute for Computer ScienceUniversity of Tuebingen, Germany
[38].J. MacQueen, “Some methods for classification and analysis of mul-tivariate observations,” in Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press,1967, pp. 281–297.
[39]Optimising a one-class SVM for geographic object-based novelty detection Christoff Fourie 1, Adriaan van Niekerk .2, Ladislav Mucina. Department of Geography and Environmental Studies, Stellenbosch University, Stellenbosch, South Africa, fourie.christoff@gmail.com .Centre for Geographical Analysis, Stellenbosch University,

Stellenbosch, South Africa, avn@sun.ac.za.Department of Environment and Agriculture, Curtin University, Perth, Australia,
[40]Ketchen, Jr, David J.; Shook, Christopher L. (1996). "The application of cluster analysis in Strategic Management Research: An analysis and critique". *Strategic Management Journal*. **17** (6): 441–458.
[41]Analysis of an Event Forecasting Method for Wireless Sensor Networks András KALMÁR
, Gergely ÖLLÖS, Rolland VIDA High Speed Networks Laboratory (HSNLab), Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary e-mail:{kalmar;ollos; vida}@tmit.bme.huManuscript received November 10, 2011; revised December 15, 2011
(PDF) Analysis of an Event Forecasting Method for Wireless Sensor Networks. Available from: https://www.researchgate.net/publication/321051036_Analysis_of_an_Event_Forecasting_Method_for_Wireless_Sensor_Networks [accessed Jul 23 2019].
[42]Liu, Fei Tony, Kai Ming Ting and Zhi-Hua Zhou. “Isolation-Based Anomaly Detection.” *TKDD* 6 (2012): 3:1-3:39.
[43]Lili Yin , Huangang Wang , Wenhui Fan, Active learning based support vector data description method for robust novelty detection, Knowledge-Based Systems, v.153 n.C, p.40-52, August 2018
[44]Hamideh Hajiabadi , Reza Monsefi , Hadi Sadoghi Yazdi, relf: robust regression extended with ensemble loss function, Applied Intelligence, v.49 n.4, p.1437-1450, April 2019
[45] Hanley JA, McNeil BJ. The meaning and use of the area under the receiver operating characteristic (ROC) curve. Radiology. 1982;143:29–36.
[46]Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol I IMECS 2008, 19-21 March, 2008, Hong Kong,Network Anomaly Detection Using One Class Support Vector Machine Rui Zhang, Shaoyan Zhang, Yang Lan, Jianmin Jiang.
[47]New Principle Component Analysis Based Colorizing Method,Arash Abadpour Sharif University of Technology abadpour@math.sharif.edu. Shohreh Kasaei Sharif University of Technology skasaei@sharif.edu
[48] Visualizing Data using t-SNE, Laurens van der Maaten L.VANDERMAATEN@MICC.UNIMAAS. NL MICC-IKAT Maastricht UniversityP. Box 616, 6200 MD Maastricht,The Netherlands. Geoffrey Hinton HINTON@CS.TORONTO.EDU Department of Computer ScienceUniversity of Toronto6 King’s College Road, M5S 3G4 Toronto, ON, Canada
[49]Tax, David M. J.: One-class classification-concept learning in the absence of

counterexamples. Ph.D. Dissertation, Delft University of Technology. ASCI Dissertation Series. Vol. 65. (2001) 1-190
[50]Comparison of One-Class SVM and Two-Class SVM for Fold Recognition Alexander Senf, Xue-wen Chen, and Anne ZhangThe University of Kansas, Lawrence KS 66045 USA,ajsenf@ku.edu, xwchen@ku.edu, yazhang@eecs.ku.edu
[51]Banda, Juan & Angryk, Rafal & Martens, Petrus. (2013). Steps Toward a Large-Scale Solar Image Data Analysis to Differentiate Solar Phenomena. Solar Physics. 288. 10.1007/s11207-013-0304-x.
[52]An evaluation, comparison, and accurate benchmarking of several publicly available MS/MS search algorithms: Sensitivity and specificity analysis Eugene A. Kapp1*, Frédéric Schütz1*, Lisa M. Connolly1 , John A. Chakel2, Jose E. Meza2,Christine A. Miller2, David Fenyo3, Jimmy K. Eng4, Joshua N. Adkins5, Gilbert S. Omenn6 and Richard J. Simpson1
[53]Goutte, Cyril and Éric Gaussier. “A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation.” *ECIR* (2005).
[54]Analysis of clustering techniques, Priyanka jadhav, professor, Rasika Patil Professor, Bharti Vidyapeeth's institute of management and information Technology, CBD Belapur, Navi mumbai, Maharashtra, India.
[55]Jeong, Min-Su & Jang, Ho-Won & Kim, Jong-Suk & Lee, Joo-Heon. (2016). ROC evaluation for MLP ANN drought forecasting model. Journal of Korea Water Resources Association. 49. 10.3741/JKWRA.2016.49.10.877.

Acronyms

AM-Access Management
API-Application Programming Interface
AD- Anomaly Detection
DoS- Denial of Service
DDos-Distributed Denial of Service
DOW-Day Of Week
HTTP-Hypertext Transfer Protocol

IAM- Identity Access Management
IDM-Internet Download Manager
ELK-Elasticsearch,Logstash,Kibana
PCA- Principle component Analysis
SSD-Sum of Square Distance
t-SNE- T-distributed Stochastic Neighbor Embedding
UI-User Interface
URL- Uniform Resource Locator