

**NDDOS – TCP SYN FLOODING DETECTION USING SVM**



**UNIVERSITI TEKNIKAL MALAYSIA MELAKA**

## NDDoS – TCP SYN Flooding detection using SVM

AZWAR HAFUZA BIN MOHD NASIR



This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Computer Networking) with Honours.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY  
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

## DECLARATION

I hereby declare that this project report entitled  
**[NDDoS – TCP SYN Flooding detection using SVM]**  
is written by me and is my own effort and that no part has been plagiarized  
without citations.

STUDENT : AZWAR HAFUZA BIN MOHD NASIR

Date : 10/9/2021



اوتنومر سته تیکنیکا ملایسا ملاک  
I hereby declare that I have read this project report and found

this project report is sufficient in term of the scope and quality for the award of  
Bachelor of Computer Science (Computer Networking) with Honours.

SUPERVISOR :  Date : 10/9/2021  
(TS. NOR AZMAN BIN MAT ARIFF)

## DEDICATION

Alhamdulillah. All praise to Allah in providing me a good surrounding upon completing this project. This dedication is for my family, my supportive members and for my supervisors Ts. Nor Azman Bin Mat Ariff for all the inspiration, motivation, support and always give the best to me while completing this project.



## ACKNOWLEDGEMENTS

At first, thank you Allah for all of the blessing and moral guidance while completing this project, and also for providing me a good and supportive surrounding. Because, if I'm doing alone, I will never succeed to complete this project.

Next, greatest thank you for my supervisor. Ts. Nor Azman Bin Mat Ariff. Because always give me moral support and not tired while answering my question. If not have the right guidance and knowledge, I sure that I will never complete this project.

Lastly, thank you for my family, members, classmates and for all who help me direct indirectly upon completing this project.



## ABSTRACT

Leading to Industrial Revolution (IR) 4.0, most of the services are depending on the technology. This changes also will lead to a war named cyber war. The most popular weapon that was used during cyber-attack is Denial of Service (DoS) or Distributed Denial of Service (DDoS). In order to control this problem, a good detection system must be implements in the network architecture. The purpose of doing this project is to propose a DoS TCP SYN flooding detection using machine learning algorithm specifically Support Vector Machine (SVM). In this study, a dataset that was used is gain from Canadian Institute for Cybersecurity named NSL-KDD dataset. In conclusion, hope this project will achieve its goals and can be used for all people as precaution step for securing its network.

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

## ABSTRAK

Menuju Revolusi Perindustrian 4.0, kebanyakan servis bergantung kepada teknologi. Perubahan ini boleh membawa kepada perang disebut sebagai perang siber. Senjata yang paling terkenal yang digunakan semasa perang siber ialah serangan DoS dan juga DDoS. Untuk mengekang masalah ini, system pendeteksi yang berkesan mestilah diwujudkan dan digunakan didalam seni bina rangkaian. Tujuan menjalankan projek ini adalah untuk mencadangkan 'DoS TCP SYN flooding detection' sebagai alat pendeteksi serangan DoS yang menggunakan algoritma pembelajaran mesin lebih spesifik ialah Mesin Sokongan Vektor. Semasa penyelidikan ini, set data yang digunakan ialah daripada Institusi Keselamatan Siber Kanada bernama NSL-KDD. Sebagai konklusi, semoga projek ini akan mencapai matlamatnya dan boleh dimanfaatkan oleh semua orang sebagai langkah awal untuk menyelamatkan rangkaian mereka.

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

## TABLE OF CONTENTS

	<b>PAGE</b>
<b>DECLARATION.....</b>	<b>II</b>
<b>DEDICATION.....</b>	<b>III</b>
<b>ACKNOWLEDGEMENTS.....</b>	<b>IV</b>
<b>ABSTRACT .....</b>	<b>V</b>
<b>ABSTRAK .....</b>	<b>VI</b>
<b>TABLE OF CONTENTS.....</b>	<b>VII</b>
<b>LIST OF TABLES .....</b>	<b>XIII</b>
<b>LIST OF FIGURES .....</b>	<b>XIV</b>
<b>LIST OF ABBREVIATIONS .....</b>	<b>XVII</b>
<b>LIST OF ATTACHMENTS.....</b>	<b>XVIII</b>
<b>CHAPTER 1: INTRODUCTION.....</b>	<b>1</b>
1.1 INTRODUCTION.....	1
1.2 PROBLEM STATEMENT (PS).....	2
1.3 PROJECT QUESTION (PQ).....	4
1.4 PROJECT OBJECTIVE (PO).....	4
1.5 PROJECT SCOPE .....	5
1.6 PROJECT CONTRIBUTION (PC).....	5
1.7 REPORT ORGANISATION.....	6
1.8 CONCLUSION.....	6
<b>CHAPTER 2: LITERATURE REVIEW.....</b>	<b>7</b>



2.1	INTRODUCTION .....	7
2.2	INTRUSION DETECTION SYSTEM (IDS).....	9
2.2.1	IDS DEFINITION .....	9
2.2.2	IDS DETECTION METHOD .....	9
2.2.2.1	SIGNATURE-BASED .....	10
2.2.2.2	ANOMALY-BASED .....	10
2.2.3	TYPE OF IDS .....	10
2.2.3.1	HOST INTRUSION DETECTION SYSTEM (HIDS) .....	11
2.2.3.2	NETWORK INTRUSION DETECTION SYSTEM (NIDS).....	11
2.3	DENIAL-OF-SERVICE (DOS).....	11
2.3.1	DOS DEFINITION.....	11
2.3.2	DISTRIBUTED DENIAL-OF-SERVICE (DDOS) .....	12
2.3.3	CATEGORIES OF ATTACK .....	12
2.3.4	TYPE OF ATTACK.....	13
2.3.4.1	SYN FLOOD ATTACK.....	13
2.3.4.2	ICMP FLOOD ATTACK.....	14
2.3.5	DOS ATTACK TOOLS .....	15
2.3.6	DOS DETECTION TECHNIQUE .....	16
2.3.7	DOS PREVENTION TECHNIQUE .....	16
2.4	MACHINE LEARNING .....	17
2.4.1	MACHINE LEARNING DEFINITION.....	17
2.4.2	DATASET .....	18
2.4.3	FEATURE EXTRACTION.....	18
2.4.4	FEATURE SELECTION.....	19

2.4.5	CLASSIFIER VS MODEL.....	20
2.5	CRITICAL REVIEW.....	20
2.5.1	A STUDY ON NSL-KDD DATASET FOR INTRUSION DETECTION SYSTEM BASED ON CLASSIFICATION ALGORITHMS .....	20
2.5.2	APPLICATION-LAYER DDOS DETECTION BASED ON ONE- CLASS SUPPORT VECTOR MACHINE.....	23
2.5.3	DDOS ATTACK MODELING AND DETECTION USING SMO .....	26
2.5.4	MACHINE LEARNING DDOS DETECTION USING STOCHASTIC GRADIENT BOOSTING .....	28
2.5.5	SYN FLOOD ATTACK DETECTION IN CLOUD COMPUTING USING SUPPORT VECTOR MACHINE.....	30
2.5.6	A MACHINE LEARNING APPROACH FOR DDOS (DISTRIBUTED DENIAL OF SERVICE) ATTACK DETECTION USING MULTIPLE LINEAR REGRESSION .....	31
2.6	PROPOSED SOLUTION .....	35
2.7	CONCLUSION.....	35
<b>CHAPTER 3: DESIGN</b> .....		<b>36</b>
3.1	INTRODUCTION.....	36
3.2	METHODOLOGY .....	37
3.2.1	PREVIOUS RESEARCH.....	38
3.2.2	INFORMATION GATHERING.....	38
3.2.3	DEFINE SCOPE.....	38
3.2.4	DESIGN AND IMPLEMENTATION .....	38
3.2.5	TESTING AND EVALUATION OF MODEL.....	39
3.2.6	DOCUMENTATION .....	39
3.3	PROJECT GANTT CHART .....	39
3.4	PROJECT FLOW CHART.....	40

3.5	PROJECT MILESTONES.....	40
3.6	CONCLUSION.....	43
<b>CHAPTER 4: ANALYSIS AND DESIGN.....</b>		<b>44</b>
4.1	INTRODUCTION .....	44
4.2	PROBLEM ANALYSIS .....	44
4.3	REQUIREMENT ANALYSIS .....	44
4.3.1	SOFTWARE REQUIREMENT .....	44
4.3.2	HARDWARE REQUIREMENT.....	45
4.4	PROJECT DESIGN .....	46
4.4.1	DATASET .....	46
4.4.2	DATA PREPROCESSING .....	48
4.4.3	FEATURE SELECTION.....	49
4.4.4	DATA SPLITTING .....	50
4.4.5	CLASSIFICATION.....	51
4.4.6	CONCLUSION.....	53
<b>CHAPTER 5: IMPLEMENTATION.....</b>		<b>54</b>
5.1	INTRODUCTION .....	54
5.2	SOFTWARE DEVELOPMENT ENVIRONMENT SETUP.....	54
5.3	PROCESS MODULE .....	55
5.3.1	COLLECTION OF DATASET .....	55
5.3.2	DATA PREPROCESSING .....	55
5.3.2.1	REMOVE IRRELEVANT FEATURES .....	56
5.3.2.2	CATEGORICAL ENCODING .....	58
5.3.3	FEATURE SELECTION.....	62

5.3.4	TRAIN AND TEST DATA.....	65
5.3.4.1	SPLITTING DATA .....	65
5.3.4.2	MODEL FILE.....	67
5.3.4.3	PREDICT FILE .....	68
5.4	CLASSIFICATION .....	68
5.4.1	CODE REVIEW .....	69
5.4.2	EXECUTE .....	72
5.5	RESULT .....	72
5.5.1	RESULT OF TESTING MODEL .....	72
5.5.2	ACCURACY TABLE .....	74
5.6	CONCLUSION.....	74
<b>CHAPTER 6: DISCUSSION .....</b>		<b>75</b>
6.1	INTRODUCTION .....	75
6.2	DISCUSSION OF THE PROJECT .....	75
6.3	DISCUSSION ON THE PROPOSED METHOD.....	76
6.4	CONCLUSION.....	77
<b>CHAPTER 7: PROJECT CONCLUSION .....</b>		<b>78</b>
7.1	INTRODUCTION .....	78
7.2	PROJECT SUMMARY .....	78
7.3	PROJECT CONSTRAINT .....	79
7.4	PROJECT CONTRIBUTION.....	79
7.5	PROJECT LIMITATION .....	79
7.6	FUTURE WORK.....	79

7.7	CONCLUSION.....	79
	<b>REFERENCES.....</b>	<b>80</b>
	<b>APPENDIX A – PROJECT GANN CHART .....</b>	<b>82</b>



## LIST OF TABLES

	PAGE
<b>Table 1 – Problem statement.....</b>	<b>3</b>
<b>Table 2 – Project Question .....</b>	<b>4</b>
<b>Table 3 – Project Objective .....</b>	<b>4</b>
<b>Table 4 – Project contribution .....</b>	<b>5</b>
<b>Table 5 – Tools to launch DoS attack.....</b>	<b>15</b>
<b>Table 6 – Shows the list for both type algorithm.....</b>	<b>20</b>
<b>Table 7 – List files in the dataset gained[11]. .....</b>	<b>21</b>
<b>Table 8 – Comparison of all critical review.....</b>	<b>34</b>
<b>Table 9 – Project milestone .....</b>	<b>43</b>
<b>Table 10 – Software requirement .....</b>	<b>45</b>
<b>Table 11 – Hardware requirement .....</b>	<b>45</b>
<b>Table 12 - file include during download the dataset.....</b>	<b>47</b>
<b>Table 13 – brief for NSL-KDD Dataset.....</b>	<b>47</b>
<b>Table 14 – the feature description for the dataset[11].....</b>	<b>48</b>
<b>Table 15 - equation of every kernel in SVM.....</b>	<b>52</b>
<b>Table 16- Shows the summarize of dataset .....</b>	<b>66</b>
<b>Table 17 – Parameter description.....</b>	<b>69</b>
<b>Table 18 – Training model parameter description .....</b>	<b>70</b>
<b>Table 19 – y_predict description.....</b>	<b>71</b>
<b>Table 20 – Accuracy table .....</b>	<b>74</b>
<b>Table 21 – Shows the description of the proposed model.....</b>	<b>77</b>
<b>Table 22 – Project Gann Chart.....</b>	<b>82</b>

## LIST OF FIGURES

	PAGE
<b>Figure 1 – Statistics cybercrime in Malaysia.....</b>	<b>2</b>
<b>Figure 2 – Shows the outline for the next discuss topic. ....</b>	<b>8</b>
<b>Figure 3 – Shows the DoS vs DDoS attack.....</b>	<b>12</b>
<b>Figure 4 – TCP 3-Way Handshake.....</b>	<b>13</b>
<b>Figure 5 – Shows the stages of the ICMP.....</b>	<b>14</b>
<b>Figure 6 – Illustrate the ICMP flood attack. ....</b>	<b>14</b>
<b>Figure 7 – Shows the snort detecting DoS attack.....</b>	<b>16</b>
<b>Figure 8 - Shows the machine learning train and test flow.....</b>	<b>17</b>
<b>Figure 9 – Shows the concept of Bag of Words in Feature Extraction .....</b>	<b>19</b>
<b>Figure 10 – Shows the type of attack in the dataset.[11] .....</b>	<b>22</b>
<b>Figure 11 – Shows the accuracy gain from different algorithm[11].....</b>	<b>22</b>
<b>Figure 12 – Shows the algorithm for training in this project[12].....</b>	<b>25</b>
<b>Figure 13 – Shows the Receiver Operating Characteristics (ROC) curves that illustrated the performance of the proposed model in detecting application-layer DDoS attack in real situation[12].....</b>	<b>25</b>
<b>Figure 14 – Shows the architecture while generating the data[13]. ....</b>	<b>26</b>
<b>Figure 15 – Shows the performance of the SMO algorithm in predicting the attack[13]. ....</b>	<b>27</b>
<b>Figure 16 – Shows the result for test data 1[13]. ....</b>	<b>27</b>
<b>Figure 17 - Shows the dataset used in testing the proposed model[14].....</b>	<b>28</b>
<b>Figure 18 - shows the accuracy result obtain by few popular ml algorithm by using balanced dataset[14]. ....</b>	<b>29</b>
<b>Figure 19 – Indicate the accuracy of predicting SYN flood by using SVM [15]</b>	<b>30</b>

**Figure 20 - Machine Learning approach by using Multiple Linear Regression for detecting DDoS attack in the network. [16]..... 31**

**Figure 21 - IG result for all the features [16]..... 32**

**Figure 22 – Flow stages on chosen methodology..... 37**

**Figure 23 – Project Flowchart ..... 40**

**Figure 24 – Project design in this research..... 46**

**Figure 25 - black line that separate class blue and class red. The black line is called as hyperplane..... 51**

**Figure 26 - shows the support vector line to separate the class blue and red. The support line will find the nearest class. The distance between support vector line and hyperplane is called as margin. .... 51**

**Figure 27 – NLSVM example. This type of SVM cannot be done with simple match the hyperplane. So, kernel is needed to class the non-linear type of SVM ..... 52**

**Figure 28 – Process module task..... 55**

**Figure 29 – Data Preprocessing phase ..... 55**

**Figure 30 – Step 1 Phase 1..... 56**

**Figure 31 – Step 2 Phase 1..... 57**

**Figure 32 – Step 3 Phase 1..... 58**

**Figure 33 – Step 1 Phase 2..... 59**

**Figure 34 – Complete categorical encoding..... 60**

**Figure 35 – Step 2 Phase 2..... 61**

**Figure 36 – Step 1 Feature Selection ..... 62**

**Figure 37 – Step 2 Feature Selection ..... 62**

**Figure 38 – Step 1 of splitting data..... 65**

**Figure 39 – Step 2 of splitting data..... 66**

**Figure 40 – Shows the summarize of data after split..... 67**

**Figure 41 – Shows the process of generate the model..... 67**

**Figure 42 – Shows the process of predict the generated train model..... 68**

**Figure 43 – Model code review ..... 69**

**Figure 44 – Train result code review..... 70**

**Figure 45 – Generate the y\_predict for confusion matrix ..... 71**

**Figure 46 – Shows the code for print the report ..... 71**

**Figure 47 – Shows the code for print the confusion matrix ..... 71**



<b>Figure 48 – Shows the process of train model .....</b>	<b>72</b>
<b>Figure 50 – Show the prompt save file name and the average of the training accuracy .....</b>	<b>72</b>
<b>Figure 51 – Shows the report and confusion matrix from test process.....</b>	<b>73</b>
<b>Figure 52 – 10-Fold Cross Validation score.....</b>	<b>76</b>



## LIST OF ABBREVIATIONS

<b>FYP</b>	-	<b>Final Year Project</b>
DDoS	-	Distributed Denial of Service
DoS	-	Denial of Service
HTTP	-	Hypertext Transfer Protocol
UDP	-	User Datagram Protocol
ICMP	-	Internet Control Message Protocol
TCP	-	Transmission Control Protocol
SYN	-	Synchronize
IDS	-	Intrusion Detection System
SVM	-	Support Vector Machine
HIDS	-	Host-Based Intrusion Detection System
NIDS	-	Network Intrusion Detection System
PPS	-	Packets Per Seconds
RPS	-	Request Per Seconds
SYN-ACK	-	Synchronize-Acknowledge
CPU	-	Central Processing Unit
CMD	-	Command Prompt
OS	-	Operating System
RBF	-	Radial Basis Function
CIC	-	Canadian Institute for Cybersecurity

## LIST OF ATTACHMENTS

	<b>PAGE</b>
<b>Appendix A</b>	
<b>Project Gann Chart</b>	<b>82</b>



## CHAPTER 1: INTRODUCTION

### 1.1 INTRODUCTION

Distributed Denial of Service (DDoS) are the extended for Denial of Service (DoS). As knowledge, DoS is the type of attack that require and launched by the single attacker machine. While DDoS are the combination of multiple machine that are set to be an attacker machine that will be used to perform network hack like DDoS. DDoS attack will be the high impact attack for the network because the attack is launch by many devices or machine in one single time.

The problem for DDoS attack is, sometimes owner for the attacker machine does not know that they involved in the attack because the script for DDoS attack is inserted on the malicious software or program that downloaded on the internet. There are few methods to launch the Dos or DDoS attack and the common is by launching the flood attack. Flooding attack can be categorized into several version depending on what protocol packet that attacker used for example HTTP flood, UDP flood, ICMP flood or commonly known as Ping flood or the other name is Ping of Death. The type of flood that will be discuss on this project is TCP flood. TCP flood on DDoS or DoS attack is using handshake in 3-way TCP handshake process[1].

The attack is starting on the first handshake step which is synchronize (SYN) the common name for this type of attack is called TCP SYN flood attack. In general, flow of this attack is the attacker send many requests to the server by using TCP protocol. Server will busy to respond the huge amount of request launched by attacker and it will be denied other real request due to focusing on the SYN attack request. Effect from this attack will be the serious problem because victim might having lost due to their business cannot be operated while the attacked were launched.

## 1.2 PROBLEM STATEMENT (PS)

Hacking nowadays become a trend as method to bringing down someone else. According to (Basyir, 2021) on his article in New Straits Time Malaysia, Malaysia have faced a huge amount of loss since 2017 due to cybercrime frauds and attack. The amount stated is RM 2.23B. The rate statistic of attack is indicated on pie chart below.

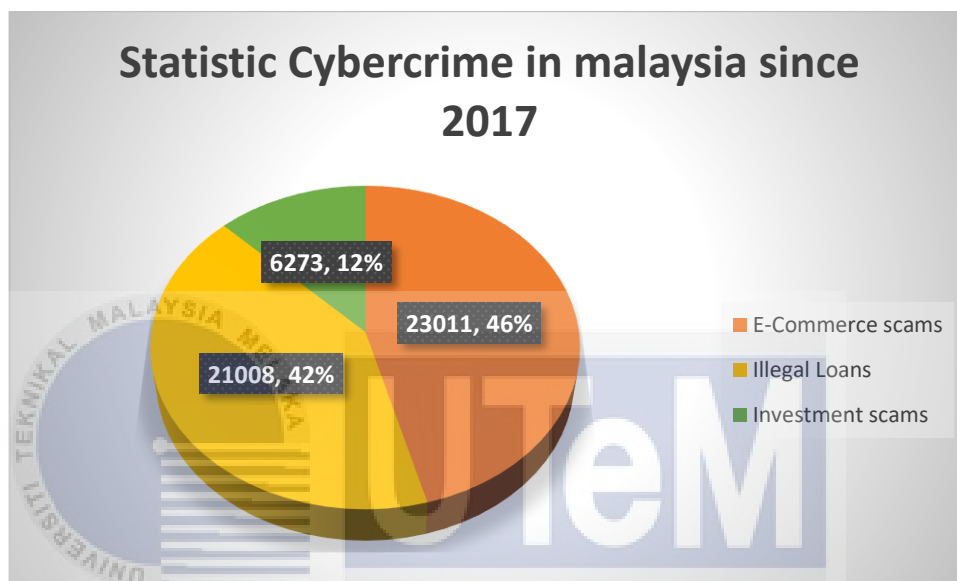


Figure 1 – Statistics cybercrime in Malaysia

This means, cyber-attack is the serious case that need a specialist expertise to solve this problem. According to figure 1. The rate shows the statistic for the scams or cybercrime fraud. But, if we deep inside the cases. In order to launch that attack, attacker also will be involved in cybersecurity threat attack such as data breach, and others network type of attack. This means, they also will have to launch the attack to the networks that relate with the Intrusion Detection System IDS. The IDS is the general terms of network attack.

As mentioned, IDS is only the term. Inside of IDS topic, there are generous of attack that widely use example phishing, deface, denial of service or Distributed denial of Service and others. If someone or some organization get stuck in this attack, they might cause a huge lost especially for a business owner or profit organization.

In general, if the e-commerce website on attack, customer might not reach the website or have been generate to another website that are set by attacker just to dropping the opponent business <sup>(1)</sup>. Besides, the network or traffic of the attacked network might congest and having overflow because there are several attack techniques that generate or transmit huge fake traffic towards attacked network just to prevent the opponent from succeed<sup>(2)</sup>. Regarding to the situation, it is recommended for a server or system have an effective technique to detect and prevent the flow of traffic in the network. Based on the problem, NDDoS – TCP SYN FLOOD DETECTION USING SVM is the best solution to prevent this problem. The problem statement (PS) for this plan is shown in Table 1.1.

PS	PROBLEM STATEMENT
PS1	Business drop cause from bad hacker launching DDOS attack to the network
PS2	Network getting slower due to congested traffic by flooding attack

**Table 1 – Problem statement**

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

### 1.3 PROJECT QUESTION (PQ)

There are four project questions came based on problem statement before this study. The mentioned question is summarized into table 1.2 below:

PS	PQ	PROBLEM STATEMENT
PS1	PQ1	How to classify the traffic towards network?
	PQ2	How to prevent attacker sending huge transmission on the network?
PS2	PQ1	How to detect normal transmission and fake transmission?
	PQ2	Is the machine learning algorithm can be used to measure the accuracy in detecting the spam or attack traffic towards network?

**Table 2 – Project Question**

### 1.4 PROJECT OBJECTIVE (PO)

Project objectives (PO) is the goals for this research. Table 1.3 below are the objectives on this project:

PS	PQ	PO	PROJECT QUESTION
PS1	PQ1	PO1	To study taxonomy of DoS and DDoS attack focuses on TCP SYN Flood.
	PQ2	PO2	To develop a classification system that can detect the attack towards network
PS2	PQ1		PO3
	PQ2		

**Table 3 – Project Objective**

## 1.5 PROJECT SCOPE

Below is the scope that covered in this project:

- I. This research only focuses on TCP SYN flooding attack.
- II. The solution is based on Machine Learning solution
- III. The effectiveness of the proposed model is measured using accuracy.

## 1.6 PROJECT CONTRIBUTION (PC)

Output or contribution for this project is to classify the traffic that transmit to networks whether it is normal traffic or flood attack launch by bad attacker. The overview of the project contribution (PC) is shown in table 1.4 below:

PS	PQ	PO	PC	PROJECT CONTRIBUTION
PS1	PQ1	PO1	PC1	Proposes the Machine learning script that can be used in predicting the anomaly activities in the network.
	PQ2	PO2		
PS2	PQ1	PO2		
	PQ2	PO3		

**Table 4 – Project contribution**

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



## 1.7 REPORT ORGANISATION

This report consists of seven chapter to complete this project. Whole chapters are summarized as below:

### CHAPTER 1: INTRODUCTION

This chapter are introduction or general for this project that developed and analyzed by proposal for the project. This chapter is the guide in pointing the main or important thing that need to be marked as prior for this project.

### CHAPTER 2: LITERATURE REVIEW

This chapter is related based on previous researcher. Which resulting the idea or knowledge that will be used in Chapter 3

### CHAPTER 3: PROJECT METHODOLOGY

This chapter generally is the chosen solution or methodology that solve or answered the whole project objectives that mentioned in Chapter 1

### CHAPTER 4: ANALYSIS AND DESIGN

This chapter is based on result from chapter 3 which is analyzed for the methodology that seem to be used to complete this project.

### CHAPTER 5: IMPLEMENTATION

This chapter is the implementation based on result in chapter 2 to 4 in this project. In this chapter, development of the script or system will be done.

### CHAPTER 6: DISCUSSION

This chapter is the discussion phase that discuss based on implementation of the experiment executed during chapter 6.

### CHAPTER 7: PROJECT CONCLUSION

This chapter is the summarized for the whole project and also the recommendations for future improvement.

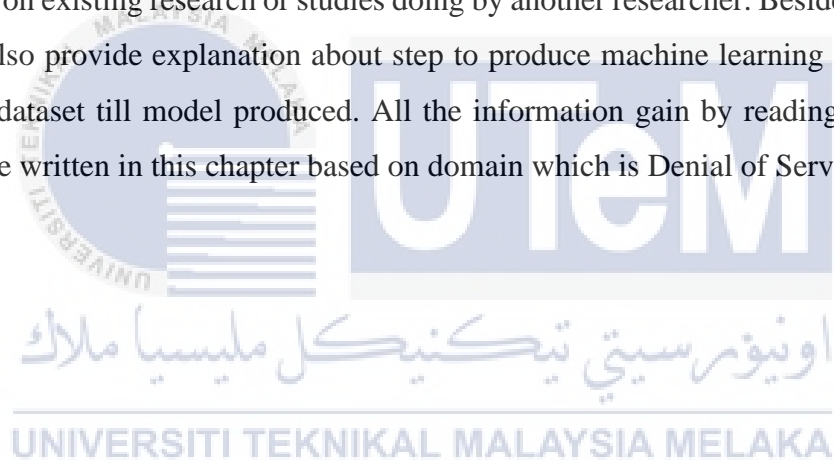
## 1.8 CONCLUSION

The research is carried out to study and gain knowledge about how to classify the traffic that transmitted into network whether it is normal traffic or flood attack traffic by hacker. Result from this study should be able to identify and classify the traffic in network to start the decision whether to prevent or allow the traffic to transmit to the networks. The following chapter discusses the related literature review focused on internet security attack and machine learning.

## CHAPTER 2: LITERATURE REVIEW

### 2.1 INTRODUCTION

This chapter will help to give clear definition and gain better understanding on how Denial of Service attack works and method for detecting or preventing this attack based on existing research or studies doing by another researcher. Besides, this chapter will also provide explanation about step to produce machine learning model starting from dataset till model produced. All the information gain by reading past research will be written in this chapter based on domain which is Denial of Service.



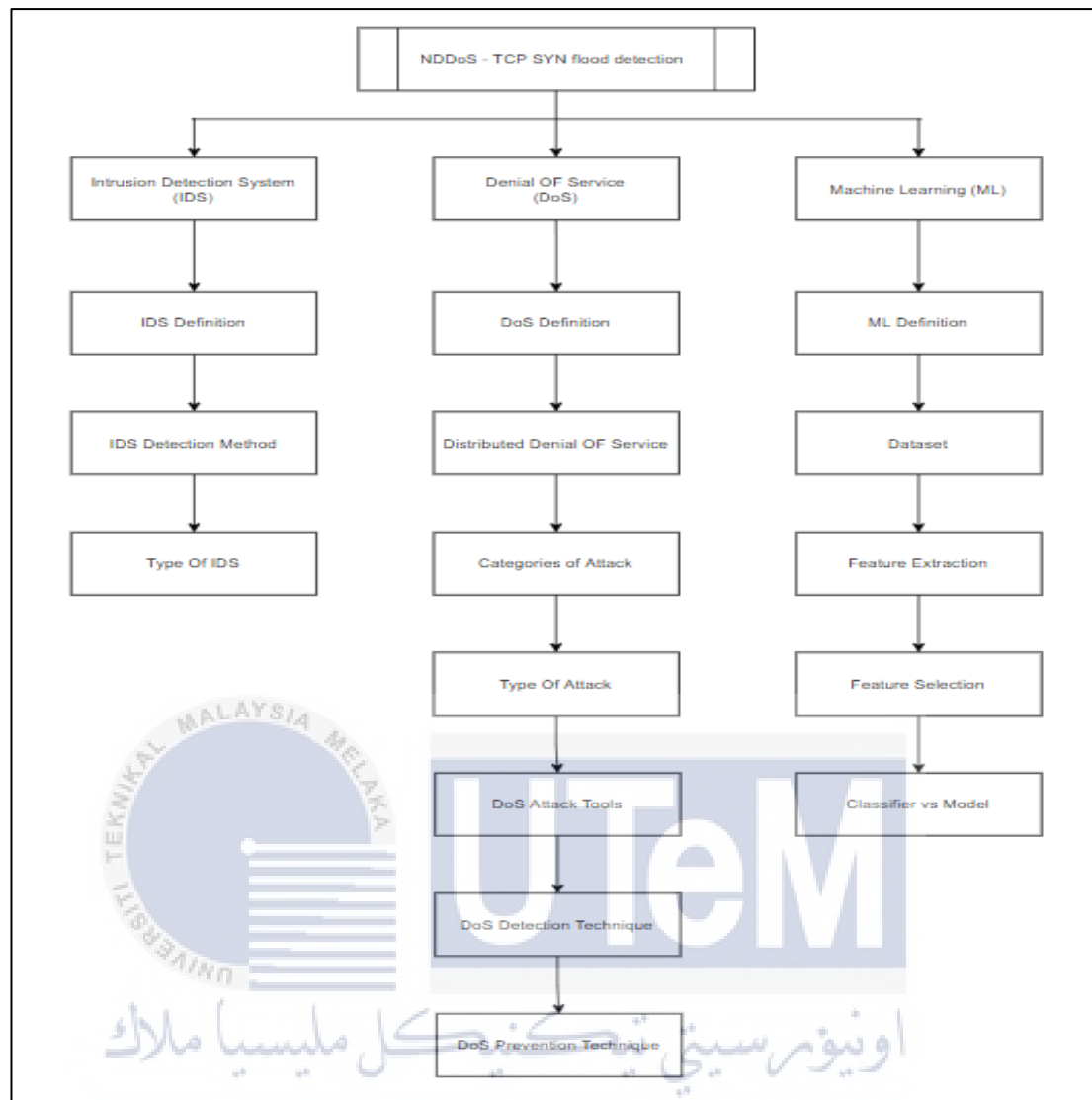


Figure 2 – Shows the outline for the next discuss topic.

## **2.2 INTRUSION DETECTION SYSTEM (IDS)**

### **2.2.1 IDS DEFINITION**

Intrusion Detection System (IDS) is a tools or technique to detect an abnormal activity in the network. The word abnormal included for all type of network threat that always occur in the network current days. Intrusion itself generally referring to the threat or attack while detection system is the way to detect all the abnormal and anomalies activity in the network. Basically, IDS will keep the network user safe because it will detect the abnormal activity and alerting the users and user can proceed to the next preventing the activity from being proceed by the attackers. The flow mechanism of IDS basically it will scan the packet that going to the network[2]. The scanner is act as sensor of all packets. This means, all the packet that enter to the network will be known by the IDS. When there has abnormal packet, IDS will alerting the users of that packet according to the rules that have been set in the configuration of IDS[2]. The rules will act as guideline of filtering the packet, if the packet that entering the network match with the rules that have been configured, IDS will trigger the alarm so that administration can take the action regarding that packet.

### **2.2.2 IDS DETECTION METHOD**

IDS is a detection system. As mentioned earlier, it's only a detection system. Not a preventing system. It's can give a precaution for the admin for the next step of avoiding the attack done in the network. In general, IDS have two detection methods.

### 2.2.2.1 SIGNATURE-BASED

IDS method for signature-based is involving a database which have been set on the IDS server or devices. When the IDS detecting any packet, it will match with the signature that have been created in the database[3]. The example of IDS software using this approach is snort[4]. It has the rules that configured and stored on the system database. When the incoming packet match with the rules, the alarm will be triggered. The drawback for this method of IDS is the database need to be updated from time to time to according to the relevant threat during the time[5].

### 2.2.2.2 ANOMALY-BASED

Anomaly-based sometimes refer to behavior based[5]. This means, Anomaly-Based IDS will learn from the whole packet and will make the decision whether it is a normal or abnormal activities. This also called as an expert system. The benefit of Anomaly-Based detection compared to the Signature-based is this method will keep learn by itself[6] and will be always relevant in detecting the suspicious activity rather than signature that will match the incoming packet with the signature databased that have been set or stored.

### 2.2.3 TYPE OF IDS

Although IDS have it approaching method of IDS. It's also had a type to classify the category of IDS that exist. Type of IDS can be classified as two types which is Host Intrusion Detection System (HIDS) and Network Intrusion Detection System (NIDS).

### 2.2.3.1 HOST INTRUSION DETECTION SYSTEM (HIDS)

HIDS detecting the intrusion or suspicious activity in the host. This type of IDS monitors the whole activity in the single host and HIDS can analyze the whole system such as operating system, logs file and etc[2]. HIDS also can take a snapshot. If the host doing some suspicious activity, HIDS will ring a bell and notified the admin to prevent that activity from being proceed.

### 2.2.3.2 NETWORK INTRUSION DETECTION SYSTEM (NIDS)

NIDS detecting the intrusion in the whole network architecture. Compared to HIDS, it's only focusing on a single host. The workflow of NIDS can be related to port that have been set as scanner port. Whenever the packet enters to that scanner port, the packet will be investigating in further by IDS if the packet contains the abnormal or suspicious activity, an alarm of IDS will be triggered[2]. The advantage of this type is attacker will not know the network have scanner port or mirroring port if they cannot access to the configuration file located in network devices such as routers.

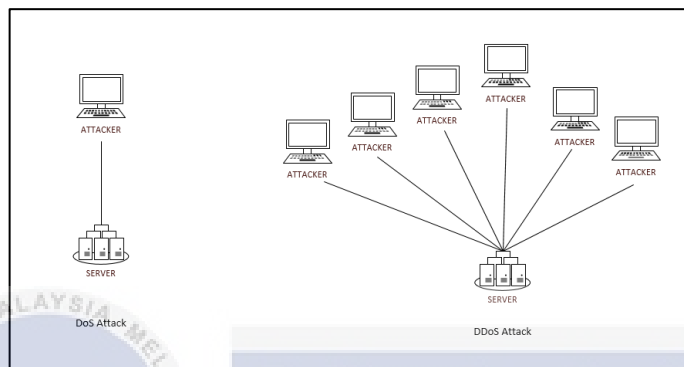
## 2.3 DENIAL-OF-SERVICE (DOS)

### 2.3.1 DOS DEFINITION

Towards Industrial Revolution 4.0, majority of business or organization using technology to run their business. When it comes to technology, server and client will involve in this methodology. Server playing a big role in this era. If server down, all the transaction, process in the system will be effect. This issue might be a big threat for all organization because if any opponent trying to sabotage the operation, they might aim to attack the server. There is variety type of attack while the common attack is Denial of Service or DoS attack. In general, DoS attack is to crash the target machine so that those machines will not be run as the normal machine due to this attack. This attack might be normal or nothing for a small organization or normal people. But if the attacked are launched to the big organization such as aircraft system, automated car system or even bank system. The attack will result a huge lost to the organization and even someone might be killed.

### 2.3.2 DISTRIBUTED DENIAL-OF-SERVICE (DDoS)

Different of DDoS, the term distributed referring to many hosts or machine are set as an attacker to launching the attack within the same time. This type of attack might involve hundred or thousand machines. Those machines normally called as zombie. This because owner of the machine does not now that their machine involved in the attack. The flow mechanism of DoS vs DDoS is simplified in the figure below.



**Figure 3 – Shows the DoS vs DDoS attack.**

### 2.3.3 CATEGORIES OF ATTACK

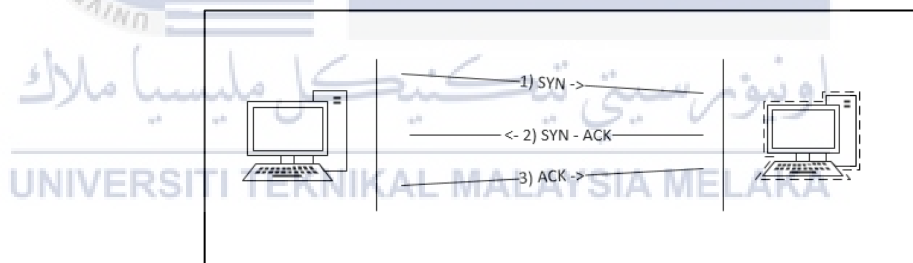
DoS attack can be classified in three categories. The three types of attack are Volume Based Attacks, Protocol Attacks and Application Layer Attacks[7]. And inside these 3 types, there are a lot of ways to implement the attack. In general, Volume Based Attacks is related to DDoS attack. This type of attack involves many attackers' machine. Each machine sending a huge request to the server and the server unable to carry the huge volume of request in one time and turn to server down. The measurement unit of Volume Based Attack is bits per second (bps). Next is Protocol Attack which this attack is aiming to turning down the server resources instead of bandwidth. The example of this attack is SYN flood by using TCP Protocol. Generally, the attacker sending a huge request by using the TCP Protocol and Server was busy responding to the flooding packet that receive and turn to crash. This type is measured in packets per second (pps). Lastly, Application Layer Attacks. This attack mostly focusing on vulnerabilities or issues that will lead to unresponsive application. This type of attack is measured in Request per second (rps).

### 2.3.4 TYPE OF ATTACK

In general, DoS can be categorized into 2 types which is flooding attack or crashing attack. The term flood can be generalized as the attacker send the huge of packet rather than target can process and it will lead to the crashing of the attack. There are few popular attacks that related to the flooding type of attack called SYN Flood attack and ICMP Flood.

#### 2.3.4.1 SYN FLOOD ATTACK

SYN flood attack is the type of DoS attack that using TCP 3-way handshake protocol. In real world, concept of 3-way handshake is client first sending a SYN and when user recognized the request, user will acknowledge the request by transmitting SYN-ACK. When server respond with sending SYN-ACK, then the client will send the ACK which mean the connection between Server and Client are successfully established. Generally, TCP 3-way handshake are illustrated as figure below.



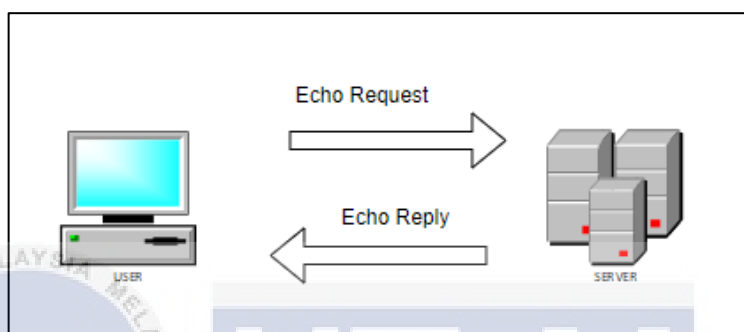
**Figure 4 – TCP 3-Way Handshake**

TCP SYN flood attack using first two layer while launching the attack. Attacker will act as normal client that transmit the huge amount of SYN packet. When server noticed, server will respond to the SYN by using SYN-ACK. Unlike normal connection, SYN flood attack will spamming a huge of SYN request. When server notice there have SYN packet incoming, server will busy to respond the request. The fact of server having crash is because server is busy respond to fake random SYN packet transmitted by attacker machine and unable to respond to any real incoming packet. Target machine also will have a huge of CPU load due to this attack.



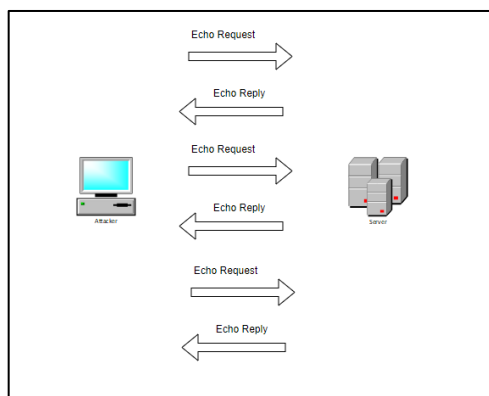
### 2.3.4.2 ICMP FLOOD ATTACK

Internet Control Message Protocol (ICMP) is the well-known protocol in networking field. This related to all the networkers that want to test the connection by using ping command. The ping command is in the ICMP protocol. As knowledge, ICMP contain of two stages which is echo request and echo reply. Figure below illustrate the flow of ICMP protocol.



**Figure 5 – Shows the stages of the ICMP.**

In DoS attack, ICMP flood will create an infinity of echo request and will make the server busy to reply all the request. The common name of this attack is Smurf attack and ping of death attack. This means, the attacker will send an infinity of echo request to the server. When server acknowledge there have incoming echo request, by its nature, server need to reply all of the request. Without knowing this is an attack, the server might be crashed due to reply all the request and ignore other real packet.



**Figure 6 – Illustrate the ICMP flood attack.**

### 2.3.5 DOS ATTACK TOOLS

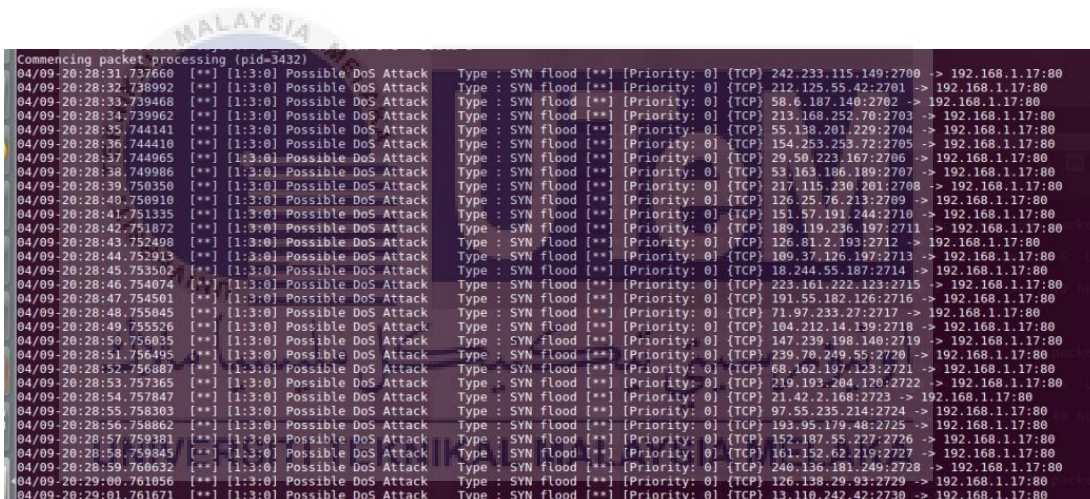
All the attack sometimes used the own script coded by bad programmers. The programmer will code its own script to launch the dos attack such as syn flood attack. But this needs an expert knowledge of manipulating the code to make the code success in launching the attack. But most of the attackers does not have this expertise and even did not know a basic of programming skills. So, most of them choose to launch the attack by common attack tools that widely provided on the internet. Table below list most of the common attack tools that can be used to launch the DoS attack.

TOOLS NAME	DESCRIPTION
Low Orbit ION Cannon (LOIC)	Can perform TCP, UDP and HTTP attack by simply enter the target IP address
High Orbit ION Cannon (HOIC)	Can perform HTTP attack by generates the huge GET and POST request to the targeted server
RUDY	Can perform HTTP attack by identify the webpage that have a form and send the POST request.
Hping3	Can perform TCP and ICMP attack
Slowloris	Can perform HTTP attack
XOIC	Can perform TCP, HTTP, UDP and ICMP attack
DDoS Simulator (DDOSIM)	Can perform HTTP attack by both valid and invalid requests.

**Table 5 – Tools to launch DoS attack.**

### 2.3.6 DOS DETECTION TECHNIQUE

In network architecture, there have several techniques that can be used to detect the attack towards network. The technique that can be used to detect the attack is using IDS. There are several IDS that can be used in this technique such as Suricata and snort. Snort is the open-source IDS that based in signature-based and categorized into NIDS category[8]. Snort will notice the admin or engineer about the suspicious activity is coming to the network if the packet match with the rules that have been configured in the configuration directory. Snort is just an IDS. To get the incoming packet noticed by snort, the port needs to be configured as mirroring port or scanner port. The function of mirror port is to make IDS analyzed in deep of the incoming packet. If the packet matches the rules as mentioned earlier, it will trigger an alarm. Below shows the example of the snort alert regarding DoS attack.



```

Commencing packet processing (pid=3432)
04/09-20:28:31.737660 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 242.233.115.149:2700 -> 192.168.1.17:80
04/09-20:28:32.738992 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 212.125.55.42:2701 -> 192.168.1.17:80
04/09-20:28:33.739468 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 58.6.187.140:2702 -> 192.168.1.17:80
04/09-20:28:34.739962 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 213.160.252.70:2703 -> 192.168.1.17:80
04/09-20:28:35.744141 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 55.138.201.229:2704 -> 192.168.1.17:80
04/09-20:28:36.744410 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 154.253.253.72:2705 -> 192.168.1.17:80
04/09-20:28:37.744965 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 29.50.223.167:2706 -> 192.168.1.17:80
04/09-20:28:38.749886 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 53.163.186.189:2707 -> 192.168.1.17:80
04/09-20:28:39.750350 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 217.115.230.201:2708 -> 192.168.1.17:80
04/09-20:28:40.750910 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 126.25.76.213:2709 -> 192.168.1.17:80
04/09-20:28:41.751335 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 151.157.191.244:2710 -> 192.168.1.17:80
04/09-20:28:42.751872 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 189.119.216.197:2711 -> 192.168.1.17:80
04/09-20:28:43.752498 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 126.81.2.193:2712 -> 192.168.1.17:80
04/09-20:28:44.752913 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 189.37.126.197:2713 -> 192.168.1.17:80
04/09-20:28:45.753502 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 18.244.55.187:2714 -> 192.168.1.17:80
04/09-20:28:46.754074 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 223.161.222.123:2715 -> 192.168.1.17:80
04/09-20:28:47.754501 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 191.55.182.126:2716 -> 192.168.1.17:80
04/09-20:28:48.755045 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 71.97.233.27:2717 -> 192.168.1.17:80
04/09-20:28:49.755526 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 104.212.14.139:2718 -> 192.168.1.17:80
04/09-20:28:50.756035 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 147.239.198.140:2719 -> 192.168.1.17:80
04/09-20:28:51.756495 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 239.70.249.55:2720 -> 192.168.1.17:80
04/09-20:28:52.756887 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 68.162.197.123:2721 -> 192.168.1.17:80
04/09-20:28:53.757365 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 219.193.204.120:2722 -> 192.168.1.17:80
04/09-20:28:54.757847 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 21.42.2.168:2723 -> 192.168.1.17:80
04/09-20:28:55.758303 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 97.55.235.214:2724 -> 192.168.1.17:80
04/09-20:28:56.758862 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 193.95.179.48:2725 -> 192.168.1.17:80
04/09-20:28:57.759381 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 152.187.95.227:2726 -> 192.168.1.17:80
04/09-20:28:58.759945 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 161.152.62.219:2727 -> 192.168.1.17:80
04/09-20:28:59.760632 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 240.136.181.249:2728 -> 192.168.1.17:80
04/09-20:29:00.761056 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 126.138.29.93:2729 -> 192.168.1.17:80
04/09-20:29:01.761671 [**] [1:3:0] Possible DoS Attack Type: SYN flood [**] [Priority: 0] [TCP] 13.110.242.42:2730 -> 192.168.1.17:80

```

Figure 7 – Shows the snort detecting DoS attack.

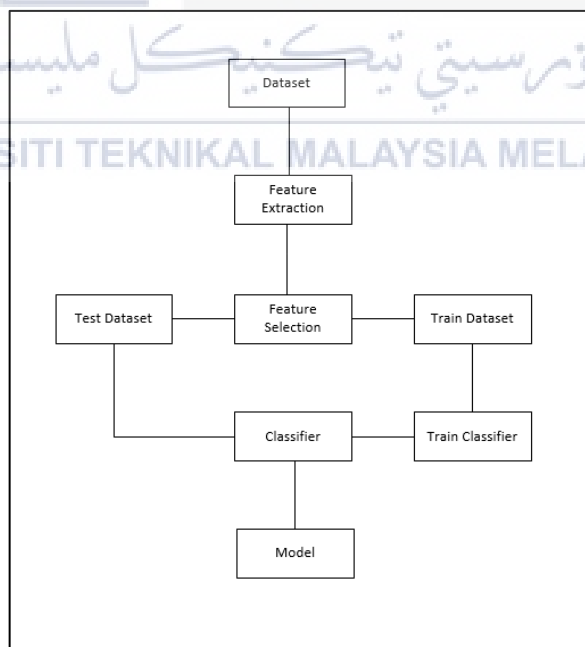
### 2.3.7 DOS PREVENTION TECHNIQUE

DoS attack can be preventing by several technique. This leads to the network and computer security field. There are several method or technique that can be used to prevent this attack towards network. The quite easy and cheaper is to add access control list (ACL)[9] in the router configuration or blocking some protocol in firewall. By this, all the packet that match with the list configured will be drop immediately by the router. This method can be implemented for both small and big organization.

## 2.4 MACHINE LEARNING

### 2.4.1 MACHINE LEARNING DEFINITION

Artificial Intelligence is the one of field that came from problem to solve or fulfil the request of human. Artificial Intelligence usually link with robot industry. This because, the robot is programmed as the clone human that can do majority of normal human task. Robot or machine also widely used in industrial field because with this technology, they do not need the human to work. Machine Learning is the one of field or part in Artificial Intelligence. Generally, machine learning is the algorithm or programmed that link with human brain. This algorithm will produce a system or product that will have a same idea, knowledge, think same just human do. This technology even more accurate than normal human because the machine is do what have been programmed and do not have careless mistake in doing their task unlike normal human did. To gain the perfect result, Machine learning required a phase that called training and testing that using complete dataset as the data to train and testing to produce a new model or sometimes called as classifier that can fulfil the requirement in real life. The flow of these phase is as figure below.



**Figure 8 - Shows the machine learning train and test flow.**

## 2.4.2 DATASET

Artificial Intelligence fundamental actually is the data. This because, to produce the perfect model of Artificial Intelligence, model need to be trained by a big data to make sure that AI model having a same knowledge as human do. Referring back to produce a machine learning model, a set of data needed to obtain the best model for the situation. Dataset can be categorized into two parts which is raw data and a data that have been transform into a set during Feature Extraction. In this domain of research, there are a lot of datasets that have been used in a same domain of study which to study and test machine learning algorithm for predicting the syn flood attack. The example of dataset that have for dos and DDoS attack is CICDDoS2019, CICDDoS2017, NSL-KDD and etc.

## 2.4.3 FEATURE EXTRACTION

Feature extraction or FE is the early step in developing machine learning model. Every produced model that using or gaining the raw dataset required to pass this feature. Generally, this features definition is to extract the raw data into a set of data. In machine learning, the raw data cannot be used to train the model. So, the first step is to extract the raw data so that the machine learning can processing the set of data or called as dataset to produce the model that can be used to solve the problem. There is example of raw data as mentioned above is text, image, geospatial data and others. There are several common models to complete the feature extraction in machine learning such as Bag of Words[10].

Bag of Words (BoW) can be imagined like put plenty of word into a bag. The word that stored into a bag is random mixed and does not sorting into order anymore. Furthermore, Bag of Words model is measured using frequency of use for every word as the result[10].The concept of BoW can be illustrated as the figure below.



**Figure 9 – Shows the concept of Bag of Words in Feature Extraction**

#### 2.4.4 FEATURE SELECTION

To produce a machine learning model, the model needs to be trained and test first just to simulate the model before it can be used in a real world. The initial step of having the test and train dataset, feature selection must be done. Feature selection can be classified into two section which is supervised and unsupervised learning depending on the type of dataset.

(Julianna Delua, 2015) defined that supervised learning approach in machine learning is used on labeled datasets. By using labeled datasets, the model can measure its accuracy comparing to the labeled input and output and this approach can make the model consistent in learning over a time. Supervised learning can be divided into two types which is classification and regression. The difference between these two is based on the type of labeled datasets or value. Regression algorithm is used to predict accuracy in continuous values such as price, time, etc. Meanwhile, Classification is used in predicting the accuracy for discrete values example gender. Furthermore, there are a few algorithms in both classification and regression algorithm. Example algorithm for both classification and regression are listed in the table below.

Classification Algorithms	Regression Algorithm
Logistic Regression	Simple Linear Regression
K-Nearest Neighbors	Multiple Linear Regression
Support Vector Machines	Polynomial Regression
Kernel SVM	Support Vector Regression
Naïve Bayes	Decision Tree Regression



Decision Tree Classification	Random Forest Regression
Random Forest Classifications	

**Table 6 – Shows the list for both type algorithm.**

#### 2.4.5 CLASSIFIER VS MODEL

Classifier and Model is depending on each other during machine learning model development. Classifier defined as algorithm that used while testing and training the data. Meanwhile, model is the result of which classifier produced a high accuracy and that classifier will be call as propose model.

#### 2.5 CRITICAL REVIEW

##### 2.5.1 A STUDY ON NSL-KDD DATASET FOR INTRUSION DETECTION SYSTEM BASED ON CLASSIFICATION ALGORITHMS

Based on[11]. This research is to study about different classification algorithms in detecting the abnormal activities in network. This research is conduct by using NSL-KDD dataset. The domain for this study is based on Intrusion Detecting System with various type of attack include DoS attack. NSL-KDD dataset is the clean version of the original dataset produced by KDD. It can be called as refine product because on the newest dataset, all the unwanted or noise features have been removed. This action will make the researchers after becoming easiest and will make the data look clean and neat. In this research, writers used a data mining software name WEKA to select or study the classification algorithm. All the algorithm used are available in WEKA software. Writers said that while conducting this study, they have exposed many relationships between protocols and network attacks. NSL-KDD dataset have provided the main for train and test data. Besides, the dataset also has the version of data splitting. All the details as in table below.

No.	Name of the file	Description
1	KDDTrain+.ARFF	Full NSL-KDD dataset for train in ARFF format

2	KDDTrain+.TXT	Full NSL-KDD dataset for train in txt format if the researchers want to export to the csv format
3	KDDTrain+_20Percent.ARFF	Split to 20% in format ARFF
4	KDDTrain+_20Percent.TXT	Split to 20% in format txt
5	KDDTest+.ARFF	Full NSL-KDD dataset for test in ARFF format
6	KDDTest+.TXT	Full NSL-KDD dataset for test in txt format
7	KDDTest-21.ARFF	Subset of the test dataset in ARFF file format
8	KDDTest-21.TXT	Subset of the test dataset in txt file format

**Table 7 – List files in the dataset gained[11].**

The reason of dataset authors split the data is to make the testing phase is not biased because of the instances have been read first while train the model. So, the result of test might be different from the train but it's still relevant and can be document because the model has been trained and tested using different instances while developed. The dataset has total 41 attributes and also have the class for differentiate each instance whether it is normal or abnormal network. This dataset has 4 type of attack which is DoS, Probe, R2L and U2R. DoS or Denial of service is a type of attack that aim the target resources and make the target machine become unresponsive. Probe is the type of attack that want gain information about the target machine such as port scanning. Next is R2L means unauthorized access from the remote machine. Or it can be called as session hijack and gain access to the target devices or machine. Lastly is U2R. U2R means unauthorized access to root, and it have all the privileges like root or admin do. Table below shows the details of dataset and its class.



Data Set Type	Total No. of					
	Records	Normal Class	DoS Class	Probe Class	U2R Class	R2L Class
KDD Train+ 20%	25192	13449	9234	2289	11	209
		53.39%	36.65%	9.09%	0.04%	0.83%
KDD Train+	125973	67343	45927	11656	52	995
		53.46%	36.46%	9.25%	0.04%	0.79%
KDD Test+	22544	9711	7458	2421	200	2754
		43.08%	33.08%	10.74%	0.89%	12.22%

**Figure 10 – Shows the type of attack in the dataset.[11]**

In study of this dataset, authors have used 3 different algorithms for all type of attack contain in the dataset. The algorithm that was used is J48, SVM and Naïve Bayes. Table below shows the accuracy gain by using 3 different algorithms for each type of attacks.

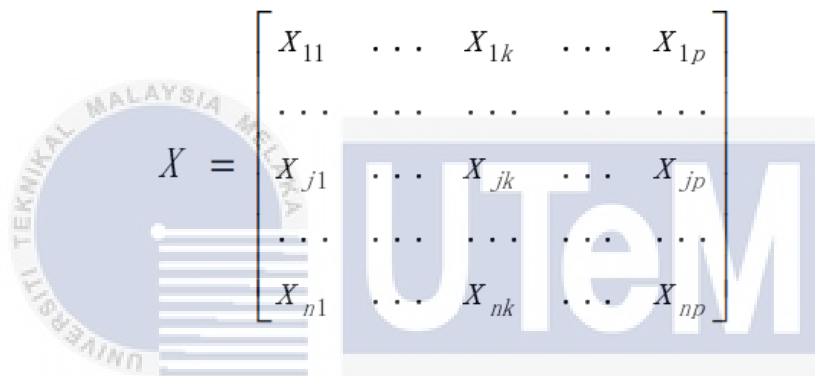
Classification Algorithm	Class Name	Test Accuracy with 6 features
J48	Normal	99.8
	DoS	99.1
	Probe	98.9
	U2R	98.7
	R2L	97.9
SVM	Normal	98.8
	DoS	98.7
	Probe	91.4
	U2R	94.6
	R2L	92.5
Naïve Bayes	Normal	74.9
	DoS	75.2
	Probe	74.1
	U2R	72.3
	R2L	70.1

**Figure 11 – Shows the accuracy gain from different algorithm[11].**

As conclusion, the NSL-KDD dataset is the good dataset to conduct the research in choosing the algorithm for classifying the attack in the networks especially in IDS domain. Writers have used correlation-based feature selection (CFS) in feature selection and then used feature reduction or dimensionality reduction as to remove the data noise and make the prediction process become faster because the model only looked on selected features instead of all the features that have in the dataset. Lastly, based on the table above, it shows that the dataset is a good dataset to conduct a study, the advantage of this dataset is it have multiple type of attack in IDS domain and all the data given is good for predicting the accuracy.

## 2.5.2 APPLICATION-LAYER DDOS DETECTION BASED ON ONE-CLASS SUPPORT VECTOR MACHINE

Based on [12], Researchers investigate the machine learning model that can be used to detect the dos attack that based on Application Layer. Researchers said that the chosen algorithm in this project is One-Class Support Vector Machine (OC-SVM). OC-SVM is the variant of normal SVM. In train and test the data, writer extract total 7 features from normal user's session and then writers build the OC-SVM model to detect application-layer DDoS attacks. The dataset extract is going into data pre-processing phase to convert a data into a structure data named data matrix. This because OC-SVM required a data structure data to load the dataset.



$$X = \begin{bmatrix} X_{11} & \dots & X_{1k} & \dots & X_{1p} \\ \dots & \dots & \dots & \dots & \dots \\ X_{j1} & \dots & X_{jk} & \dots & X_{jp} \\ \dots & \dots & \dots & \dots & \dots \\ X_{n1} & \dots & X_{nk} & \dots & X_{np} \end{bmatrix}$$

Equation 1 – Matrix form.[12]

The raw data have many features, and all of this feature might not be useful in predicting the attack. So, the writer needs to do feature selection so that only relevant features will be train by the model. Before selecting the features, the data need to be classify using several methods. The method that writer used is Resource Popularity. The formulae of the Resource Popularity are as below.

$$POP_i = \frac{AC_i}{AC_{all}}$$

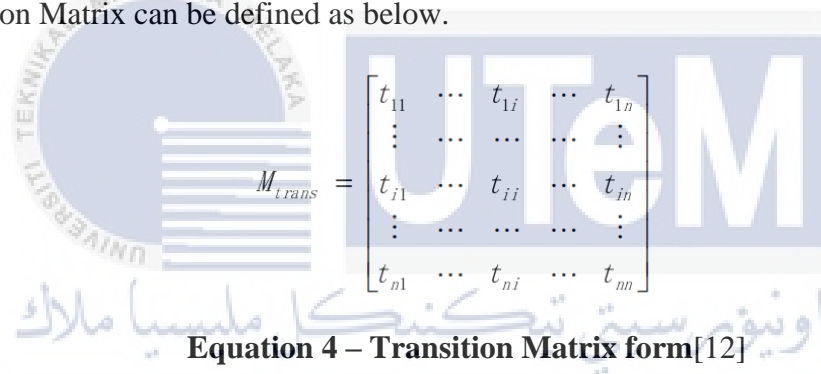
Equation 2 – Resources Popularity equation[12]

$AC_i$  refers to the number of resources accessed in a time. While  $AC_{all}$  is the total resources that access in a time. From the formula above, writer can conclude that Resources A's gain higher popularity than B's if resources A access frequency higher than Resources B. Next, Transition Probability. Writers conclude that on each website, the transition probability is different and not same for every 2 resources. Formula below shows the way to calculate Transition Probability.

$$P_{ij} = \frac{Trans_{i \rightarrow j}}{Trans_{i \rightarrow all}}$$

**Equation 3 – Transition Probability equation[12]**

$Trans_{i \rightarrow j}$  indicates the transition time between resources I to j while  $Trans_{I \rightarrow all}$  is the transition time between I to other resources. Lastly, History Transition Matrix. This actually is a record of the transition times between one resource to all resources. The Transition Matrix can be defined as below.



$$M_{trans} = \begin{bmatrix} t_{11} & \dots & t_{1i} & \dots & t_{1n} \\ \vdots & \dots & \dots & \dots & \vdots \\ t_{i1} & \dots & t_{ii} & \dots & t_{in} \\ \vdots & \dots & \dots & \dots & \vdots \\ t_{n1} & \dots & t_{ni} & \dots & t_{nn} \end{bmatrix}$$

**Equation 4 – Transition Matrix form[12]**

From all the features that have been selected, writer success to get a feature vectors for the project based on the extracted data. The vector is like below.

$$Features = (N_{session}, POP_{session}, \dots, dynamic)$$

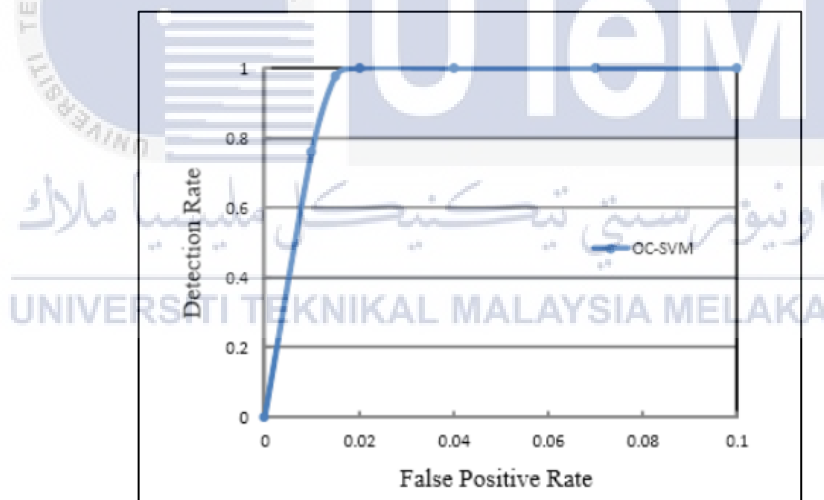
**Equation 5 – Shows features equation for this study[12].**

Figure below shows the writer used OC-SVM as the model to train the normal behavior model.

Algorithm 1 training the normal sessions	
1:	<b>Input:</b> Sessions
2:	<b>Output:</b> boundary
3:	<b>Method:</b>
4:	Step1: extract features from the set of normal sessions.
5:	Step2: normalize feature vectors and get the training set $\{x_i, i = 1, 2, \dots, n\}$ .
6:	Step3: use OC-SVM algorithm to get the boundary of the training set.
7:	(1) use the kernel function $K(x_i, x_j) = \exp(-\ x_i - x_j\ ^2 / (2\sigma^2))$ .
8:	(2) solve the corresponding OC-SVM optimization problem.
9:	(3) get the optimal boundary of the training set.

**Figure 12 – Shows the algorithm for training in this project[12].**

As conclusion, this journal is training the model differ to other researchers. This research trains the model using normal activities and then match it to the abnormal activity in detecting the attack. Figure below shows the detection result of using OC-SVM.

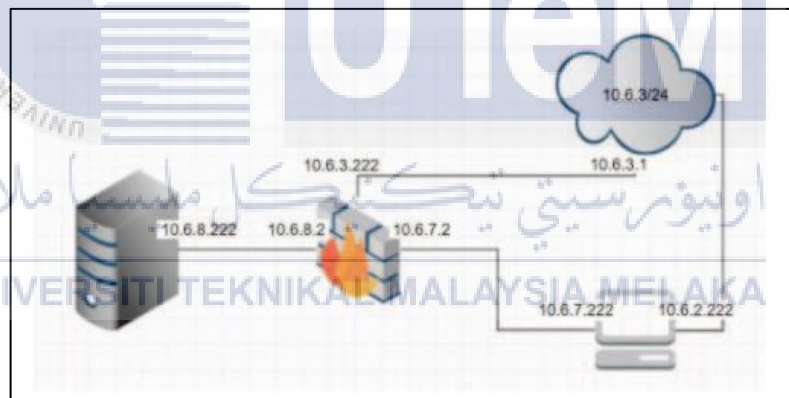


**Figure 13 – Shows the Receiver Operating Characteristics (ROC) curves that illustrated the performance of the proposed model in detecting application-layer DDoS attack in real situation[12].**

### 2.5.3 DDOS ATTACK MODELING AND DETECTION USING SMO

Based on [13]. Authors said that inspiration of doing this research is based on the cyber war that always occur since last decade. This attack not only aim the e-business field but also, they aim to attack for the government website. From this problem. Writer proposed a machine learning model to detect the DDoS attack in the network. The proposed model is using Sequential Minimal Optimization (SMO) with PolyKernel. In this study, writer gain the dataset by self-simulated the attack. The incoming packet is generating from firewall logs form the Security Information and Event Management (SEIM). From the extracted data, writers train the data by using SMO Algorithm with PolyKernel and running on 10-fold cross-validation. From the run, writers said that SMO did not give any false alarm in detecting the attack.

To generate the data that needed in this research, writers did a DoS attack towards the server. The attack consists of 3 types of attack which is TCP SYN flood, UDP flood and Smurf attack. The target machine that has been set in this testing is Windows Server 2012 while attack machine is Kali Linux.



**Figure 14 – Shows the architecture while generating the data[13].**

In this study, the firewall is used to capture the data by accessing the logfile. Writers state in the journal that on this project, they combined the attack and normal file with creating 3 data 1 for training that contain 27,7090 total logs while another two is for testing the model. Writers said they have used Weka as a machine learning software. The train and testing model also have used Weka. Table below shows the performance metrics of using SMO algorithm.

Performance Metrics <sup>a</sup>	Normal	Attack
TP Rate	1.00	1.00
FP Rate	0.00	0.00
Precision	1.00	1.00
Recall	1.00	1.00
F-Measure	1.00	1.00
MCC	1.00	1.00
ROC Area	1.00	1.00
PRC Area	1.00	1.00

**Figure 15 – Shows the performance of the SMO algorithm in predicting the attack[13].**

Based on the table above, this can be concluded that SMO can give high accuracy in determining the attack. Table below shows the results of both test data that run using the train model.

Negative	Positive	Class
290215	0	Normal
0	3	Attack

**Figure 16 – Shows the result for test data 1[13].**

In conclusion, the writer stated that SMO can be used in predicting the attack towards network. By having an experiment on testing the real live data that they generate by themselves, the SMO can give high accuracy and less false data.

## 2.5.4 MACHINE LEARNING DDoS DETECTION USING STOCHASTIC GRADIENT BOOSTING

Based on [14]. Author proposed a Stochastic Gradient Boosting (SGB) algorithm as model for detecting DDoS attacks. Writer said that they are using hybrid dataset which extract from three open dataset in the training process of the proposed model. The dataset gained is by raw data from pcap file. Writers mentioned the feature get extract by using CICFlowMeter v4 software and the software will generate the .csv file. Before proposed SGB as model, writers have trained the data by using other popular machine learning algorithm which is K-NN, Decision trees, Random Forest and Naïve Bayes and comparing the prediction accuracy result with SGB. Surprisingly, SGB algorithm produce better result than other by achieving 100% of performance metrics. Writer also said that SGB have more relevant features compared to RF algorithm. Besides, the proposed methodology also has been trained over imbalanced dataset just to simulate real-time traffic. In this case, SGB does not produce any misclassifications and also perfect in evaluating the metrics.

Data Set	File name	Tools/Attack type	File Size (GB)
CSE-CIC-IDS2018-AWS	Friday-16-02-2018/(all pcaps)	DoS-SlowHTTPTest DoS-Hulk	47
	Thursday-15-02-2018/(all pcaps)	DoS-GoldenEye DoS-Slowloris	46
	Wed-21-02-2018/(all pcaps)	DDoS-LOIC-UDP DDoS-HOIC	76
	Tues-20-02-2018/(all pcaps)	DDoS attacks-LOIC-HTTP DDoS-LOIC-UDP	52
	Thursday-20-02-2018_TrafficForML_CICFlowMeter.csv		
CICIDS2017	Monday-WorkingHours.pcap	Benign Traffic	10.8
	Friday-WorkingHours.pcap	DDoS-LOIC Port scan	8.8
CIC DoS dataset(2016)	AppDDos.pcap	slowbody2 ddosim goldeneye slow headers hulk slowloris rudy slowread	4.6

Figure 17 - Shows the dataset used in testing the proposed model[14].

Metric	SGB	RF	DT	K-NN	NB
Hyper Parameters	max_depth=5, learning_rate=0.2, subsample=0.4, colsample_bytree=1.0, colsample_bylevel=0.1, n_estimators=200 n_jobs=30	max_depth=5, n_jobs=30 n_estimators=200	max_depth=5,	k=6, n_jobs=25	Bernoulli
Accuracy (%)	100	99.95	99.94	99.94	91.81
F1-score(%)	100	99.95	99.94	99.95	91.73
Precision(%)	100	99.96	99.92	99.95	91.37
Recall(%)	100	99.95	99.96	99.95	89.93
Confusion Matrix	$\begin{bmatrix} 2086461 & 0 \\ 0 & 2135766 \end{bmatrix}$	$\begin{bmatrix} 2085725 & 736 \\ 978 & 2134788 \end{bmatrix}$	$\begin{bmatrix} 2084882 & 1579 \\ 756 & 2135010 \end{bmatrix}$	$\begin{bmatrix} 2085311 & 1150 \\ 1145 & 2134621 \end{bmatrix}$	$\begin{bmatrix} 1958145 & 128316 \\ 217195 & 1918571 \end{bmatrix}$
Total Mis-classifications	0	1714	2335	2295	345511
Execution Time	10 min	2 min	4 min	5 hrs	20 sec

**Figure 18 - shows the accuracy result obtain by few popular ml algorithm by using balanced dataset[14].**

From the analysis table shown above, it is proven that SGB have better prediction accuracy in detecting the traffic in the networks. From the both test for balanced and imbalanced dataset, SGB achieve full accuracy which is 100% compared to other algorithm which is Random Forest, Naïve Bayes, Decision trees, and also K-Nearest neighbor machine learning algorithm.



### 2.5.5 SYN FLOOD ATTACK DETECTION IN CLOUD COMPUTING USING SUPPORT VECTOR MACHINE

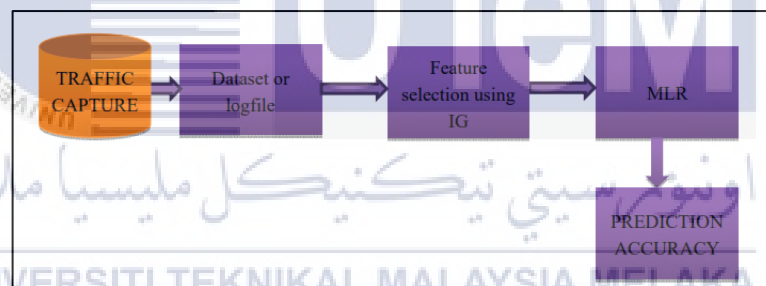
Based on [15] study, writer proposed the classification system for detecting the DoS attack in the network. Writer said that the data is obtain by themselves by simulating the cloud environment and the data is captured by using Wireshark application. After completing the feature selection phase, writer proposed the Support Vector Machine Algorithm as the model. Writer claimed that, by using SVM, the accuracy achieves up to 100%. Writer also claimed that based on the result obtain, SVM is recommended for detecting network attack.

Authors	Methodology	Data	Attack	Accuracy (%)	TP (%)	FP (%)
Khoshdel, Ali and Wasimi [9]	C4.5 decision tree	Performance data: CPU, memory, disk and network usage	RDoS SYN flood HTTP flood	Overall 93.47	-	-
Khoshdel et al [9]	Random forest	Performance data: CPU, memory, disk and network usage	SYN flood HTTP flood ICMP flood	Overall 93.9	98	0.1
Chen et al. [10]	K-means clustering Naive Bayes	Network traffic data: source and dest. IP, source and dest. Port, package length and timestamp	DDoS	-	90	0.5
Pandeewari and Kumar [11]	Hybrid of Fuzzy C-Means clustering and Artificial Neural Networks (ANN)	Network traffic data: KDD 99 dataset	DoS	99.96	97.2	5.33
Amiri et al. [12]	Least Square Support Vector Machine (LSSVM)	Network traffic data: KDD 99 dataset	DoS	84.11	78.6	0.73
Kumar, Lal and Sharma [13]	One-class Support Vector Machine	Network traffic data: source and dest. IP, bytes of data transferred, protocol	ICMP flood Ping-of-Death UDP flood SYN flood TCP Land DNS flood SYN flood	100 94 97 96 98 99 100	100 100 97 100 100 100 100	0 12 2.85 7.69 4.55 2.85 0
<b>Our proposed model</b>	<b>Support Vector Machine</b>	<b>Network traffic features: source and dest. port, sequence and acknowledgement numbers, ack and syn flags, ip ttl.</b>				

Figure 19 – Indicate the accuracy of predicting SYN flood by using SVM [15]

### 2.5.6 A MACHINE LEARNING APPROACH FOR DDOS (DISTRIBUTED DENIAL OF SERVICE) ATTACK DETECTION USING MULTIPLE LINEAR REGRESSION

According to [16], researchers used Multiple Linear Regression as the method to detect the attack. Referring to the study, dataset used is from CICIDS 2017 dataset. The objective of doing this study is to develop a machine learning model from feature selection by using regression analysis. In the study, the only data that were tested is Friday logfile or dataset. Which in this test, researchers separate the test into two session of dataset which is morning and afternoon. Researchers state that to get the relevant features in predicting the accuracy, Information Gain (IG) is used as feature selection method. IG will give the rank to all the features and the highest or nearest to 1 consider as the most relevant features. The higher prediction accuracy came from morning logfile which is gain up to 97.86%. The second session which is Friday afternoon logfile, prediction accuracy gain is 73.79%. In this research, researcher have limited the dataset for one day logfile which is Friday as mentioned earlier.



**Figure 20 - Machine Learning approach by using Multiple Linear Regression for detecting DDoS attack in the network. [16]**

S.NO	ATTRIBUTE NAME	INFORMATION GAIN	ATTRIBUTE DIMENSION IN CICIDS2017 DATASET
1	TotalLengthofFwdPackets	0.939343	5
2	SubflowFwdBytes	0.939343	64
3	AveragePacketSize	0.80995	53
4	TotalLengthofBwdPackets	0.782456	6
5	SubflowBwdBytes	0.782456	66
6	BwdPacketLengthMean	0.781841	13
7	AvgBwdSegmentSize	0.781841	55
8	forwardheaderLength	0.778016	56
9	forwardheaderLength1	0.778016	35
10	DestinationPort	0.77582	1
11	BwdPacketLengthMax	0.760317	11
12	Init_Win_bytes_forward	0.708411	67
13	AvgFwdSegmentSize	0.706064	54
14	FwdPacketLengthMean	0.706064	9
15	FwdPacketLengthMax	0.701009	7
16	BwdHeaderLength	0.682524	36

**Figure 21 - IG result for all the features [16].**

Formula and Calculation of Multiple Linear Regression

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip} + \epsilon$$

where, for  $i = n$  observations:

- $y_i$  = dependent variable
- $x_i$  = explanatory variables
- $\beta_0$  = y-intercept (constant term)
- $\beta_p$  = slope coefficients for each explanatory variable
- $\epsilon$  = the model's error term (also known as the residuals)

**Equation 6 – Formula for the MLR algorithm [16]**

Journal	Algorithm Used	Dataset Used	Accuracy Gain
<p>A MACHINE LEARNING APPROACH FOR DDOS (DISTRIBUTED DENIAL OF SERVICE) ATTACK DETECTION USING MULTIPLE LINEAR REGRESSION (S.Sambangi and L.Gondi, 2020),</p>	<p>Multiple Linear Regression</p>	<p>CICIDS 2017</p>	<p>I. 97.86% II. 73.79%</p>
<p>SYN FLOOD ATTACK DETECTION IN CLOUD COMPUTING USING SUPPORT VECTOR MACHINE (Z.Mašetić, D. Kečo, N. Dođru and K. Hajdarević, 2017)</p>	<p>Support Vector Machine</p>	<p>Self-obtain Dataset</p>	<p>100%</p>

<p>MACHINE LEARNING DDOS DETECTION USING STOCHASTIC GRADIENT BOOSTING</p> <p>(M Devendra Prasad, Prasanta Babu, C Amarnath, 2017)</p>	<p>Stochastic Gradient Boosting</p>	<p>I. CSE-CIC-IDS2018-AWS II. CICIDS2017 III. CIC DoS dataset(2016)</p>	<p>100%</p>
<p>A STUDY ON NSL-KDD DATASET FOR INTRUSION DETECTION SYSTEM BASED ON CLASSIFICATION ALGORITHMS</p> <p>(L.Dhanabal, Dr. S.P. Shantharajah, 2015)</p>	<p>K48, NAÏVE BAYES, SVM</p>	<p>NSL-KDD Dataset</p>	<p>Prediction exceeded 90% for J48 and SVM algorithm while Naïve Bayes gain more than 70% in all cases</p>
<p>APPLICATION-LAYER DDOS DETECTION BASED ON ONE-CLASS SUPPORT VECTOR MACHINE</p>	<p>ONE-CLASS SUPPORT VECTOR MACHINE</p>	<p>Self-obtain Dataset</p>	<p>High percent prediction</p>
<p>DDOS ATTACK MODELING AND DETECTION USING SMO</p>	<p>Sequential Minimal Optimization</p>	<p>Self-obtain Dataset</p>	<p>High percent prediction</p>

**Table 8 – Comparison of all critical review.**

## 2.6 PROPOSED SOLUTION

The main aim for this project is to propose the machine learning algorithm as methodology in detecting the DoS attack specifically TCP-SYN attack in the networks. The proposed model hopefully will give higher accuracy in predicting the attack in the network. Based on critical review. The selected methodology that will be used in this project is Support Vector Machine algorithm. Based [15] in study named SYN flood attack detection in cloud computing using support vector machine, writer obtain high accuracy which is 100% while testing the SVM methodology. According to [15] said that SVM Classifier is precious in detecting the attack towards network. Lastly, the dataset that will be choose in conducting this study is from NSL-KDD dataset[11] because based on this chapter, it can be classify as a good dataset.

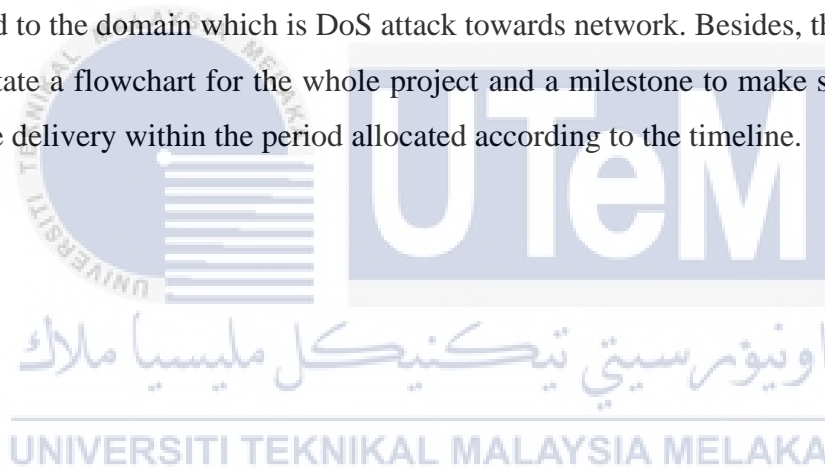
## 2.7 CONCLUSION

In conclusion, this chapter clearly brief the domain, and the important thing in this chapter is on critical review. Because in critical review subtopic, it will give an idea for proposing the suitable methodology that can be used to complete this project. All the information gain is based on reviewing past study, journal and research. In the next chapter, which is in methodology, it will discuss further into the methodology used to complete this project as mentioned in subtopic 2.6 in this chapter.

## CHAPTER 3: DESIGN

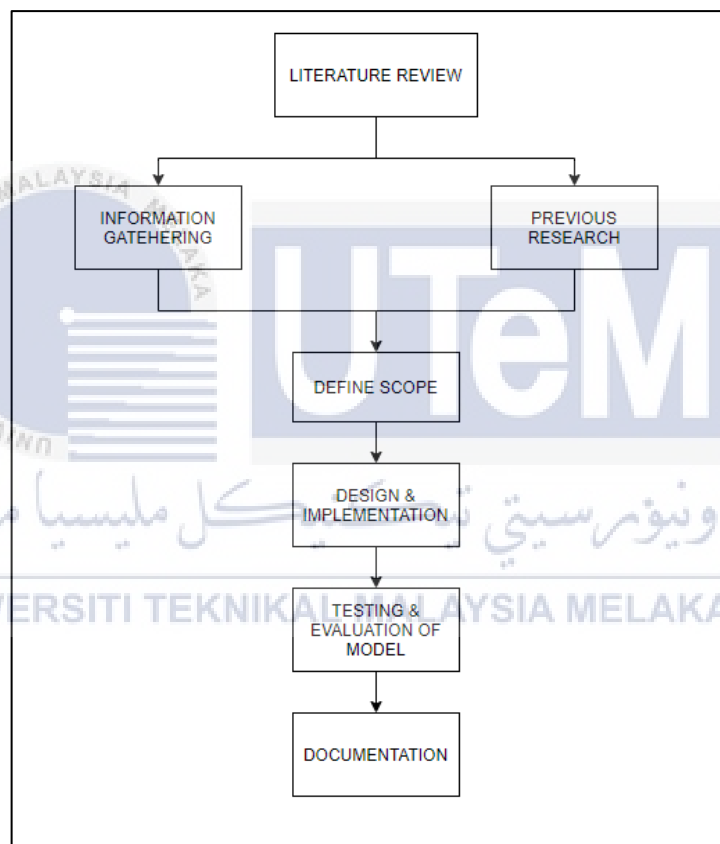
### 3.1 INTRODUCTION

This chapter is to clearly discuss about methodology that used to complete this project. The methodology gained from the chapter 2 which is literature review. Inside literature review, there have a subtopic named critical review. On this subtopic, it reviews all the past paper regarding methodology that was used to solve the problem related to the domain which is DoS attack towards network. Besides, this chapter also will state a flowchart for the whole project and a milestone to make sure the project can be delivery within the period allocated according to the timeline.



### 3.2 METHODOLOGY

Methodology is a bunch of approaches that needed to complete the project. Generally, methodology is must to make sure that proposed project is running in the right flow and achieve the whole objectives in this project. Besides, methodology also will act framework which is the guide or phase or stages to complete the project. Stages as mentioned consist of 6 stages which is previous research, information gathering, define scope, design and implementation, testing and evaluation of model and lastly documentation. All the stages flow are illustrated as figure below.



**Figure 22 – Flow stages on chosen methodology.**



### 3.2.1 PREVIOUS RESEARCH

This stage has been completed on chapter 2 which is Literature Review. This stage also will give a better understanding about the domain, and the methodology that will be used to complete the project. The domain or field that chose in this project is Denial of Service (DoS) specifically in TCP SYN Flood attack detection. Besides, this project also using machine learning model to classify whether the incoming packet is a normal packet or attack packet based on dataset that were chosen.

### 3.2.2 INFORMATION GATHERING

Based on previous research stage, all the information is extracted from the previous research paper that chosen during previous research phase. This chapter also will give an analysis or idea of which machine learning model will be choose for this project and the chosen model is Support Vector Machine (SVM).

### 3.2.3 DEFINE SCOPE

The scope for this project is limited to TCP SYN flooding attack only. This project is only to analyze or to define whether the incoming packet is an attack or normal packet based on selected dataset. Besides, the scope is only focusing on the machine learning model as the method for detecting in this project.

### 3.2.4 DESIGN AND IMPLEMENTATION

This stage is an extract for previous stage. On this stage, all the knowledge, information and understanding gained from previous research will be implement. The understanding included till first phase of machine learning model development which is gaining dataset to the feature selection phase. The dataset that will be used to complete this project is from NSL-KDD dataset[11]. Lastly, the machine learning classifier chosen is Support Vector Machine that coded using python programming language because based on understanding, SVM can be used in most cases, and it also produces high accuracy in determining or classifying the data.

### 3.2.5 TESTING AND EVALUATION OF MODEL

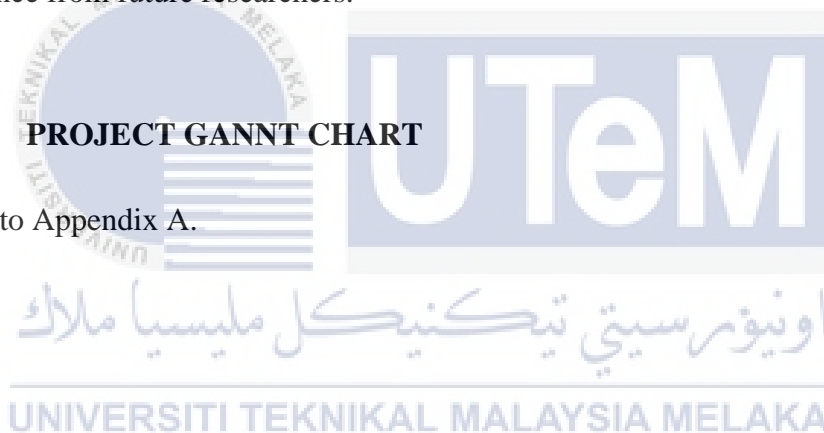
This stage will be conducted soon as implementation stage finished. On this stage, the developed machine learning model will be used to run the dataset and will determine whether the model can be used to meet the requirement for this project. It will give a prediction accuracy regarding the domain from dataset inserted which is to identify the packet is a normal packet or attack packet that limited to SYN flood attack as mentioned earlier.

### 3.2.6 DOCUMENTATION

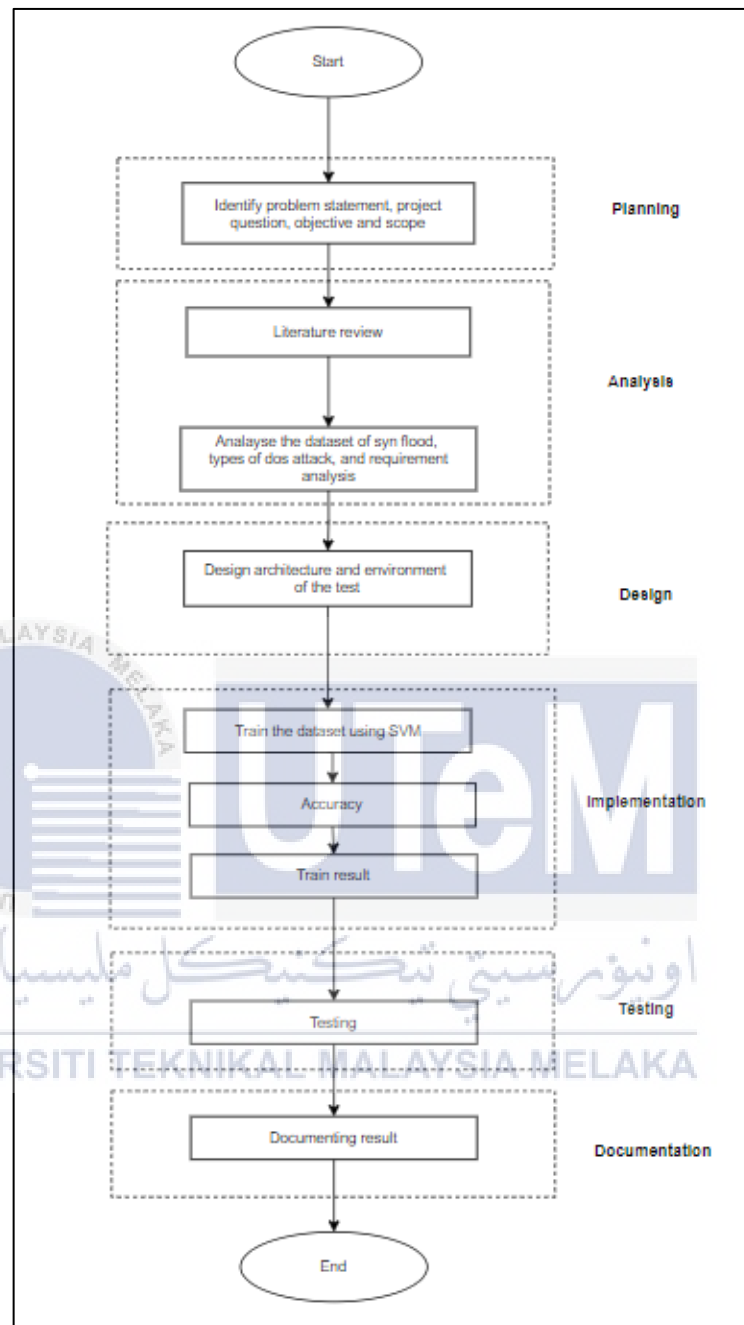
Once the model has been tested and evaluated, all the result either it is successful or even not meet the project requirements and objectives will be recorded as references to be refer in future as guidance incased this project will be made as reference from future researchers.

## 3.3 PROJECT GANTT CHART

Refer to Appendix A.



### 3.4 PROJECT FLOW CHART



**Figure 23 – Project Flowchart**

### 3.5 PROJECT MILESTONES

Project Milestones is the knowledge from project management subject. In this topic, all the work and phase will be divided into a period category. This is important to ensure that the conducted project will follow the time and keep in on track. Table below will show the project milestone that will be referred to complete this project.

WEEK	PHASE	Activity	Deliverables
< W0 (< 21/3)	Planning	Preparing the project proposal and searching for supervisor	Project Proposal
W1 (15/3 → 21/3)		Proposal correction and improvement	
W2 (22/3 → 28/3)		Waiting for proposal approval by committee	
W3 (29/3 → 4/4)		Chapter 1 :- Introduction, problem statement, project question, project objective, project contribution	Chapter 1 - Introduction
W4 (5/4 → 11/4)			
W5 (12/4 → 18/4)	Analysis	Chapter 2 :- Research on related work regarding chosen domain and doing critical review about model used by past researchers	Chapter 2 – Literature Review
W6 (19/4 → 25/4)			
W7 (26/4 → 2/5)			
W8 (3/5 → 9/5)	Analysis	Chapter 3 :- Research the methodology that will be used in this project	Chapter 3 – Project Methodology
W9 (10/5 → 16/5)		Mid Semester Break	
W10 (17/5 → 23/5)	Design	Chapter 4:-	Chapter 4 – Analysis And Design

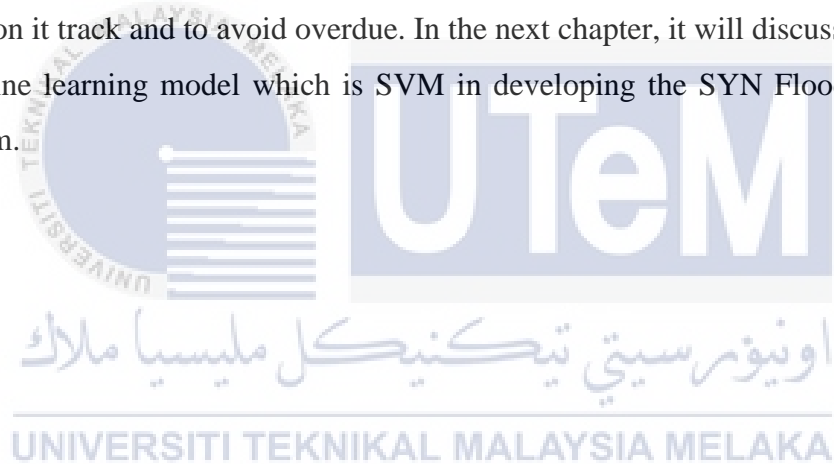
		Analyze the project that will be conducted based on previous chapter and start choosing the tools that will be used during implementation phase and design the project	
W11 (24/5 → 30/5)			
W12 (31/5 → 6/6)			
W13 (7/6 → 13/6)			
W14 (14/6 → 20/6)			
W15 (21/6 → 27/6)	Final Presentation		
W16 (28/6 → 4/7)	Implementation	Chapter 5:- Developing the machine learning model	Progress Chapter 5
W17 & W18 (5/7 → 18/7)	Final Examination Weeks		
W19 (19/7→25/7)	Implementation	Developing the machine learning model using python programming language and WEKA data mining software	Chapter 5 - Implementation
W20 (26/7→1/8)			
W21 (2/8→8/8)			
W22 (9/8→15/8)	Discussion	Discussion and analyze result that produced by implementation phase in Chapter 5	Chapter 6 – Discussion
W23 (16/8→22/8)			
W24 (23/8→29/8)	Conclusion	Conclude the project from beginning to end of the project	Chapter 7 – Project Conclusion

W25 (30/8→5/9)	Final Presentation & Project Demonstration
W26 (6/9→12/9)	Final Examination Weeks
W27 (13/9→19/9)	Inter-Semester Break

**Table 9 – Project milestone**

### 3.6 CONCLUSION

In conclusion, this chapter is the important for this project because this will act as guide on what to do based on chosen methodology stages. Besides, it also clearly briefs the milestone for this project and this very important to make sure project still keep on it track and to avoid overdue. In the next chapter, it will discuss about chosen machine learning model which is SVM in developing the SYN Flooding detection system.



## **CHAPTER 4: ANALYSIS AND DESIGN**

### **4.1 INTRODUCTION**

This chapter will specifically explain the analysis and design of the implementation method for completing this project. This chapter is related to the previous chapter which is literature review and methodology because it will describe in further what have been explain during methodology chapter with focusing on the selected machine learning algorithm Support Vector Machine. Topic that will discuss in this chapter is problem analysis and project design which contain the whole machine learning model development flow.

### **4.2 PROBLEM ANALYSIS**

The main objective of doing this research or studies is to categorize the incoming packet in the network whether it is a legit packet or just a random spam packet. Although it is the human research, the categorizing process is done with machine learning technology. To complete this, there have a dataset that need to be used to make sure that proposed machine learning model can be used to identify the incoming packet. Dataset chosen is from Canadian Institute for cybersecurity. This dataset produced in 2019 and the data have been separated according to the type of attack. The chosen dataset is according to this research domain which is SYN flood detection. The selected machine learning model is Support Vector Machine. The reason for selecting this model is because based on previous reading and study, SVM can give a higher accuracy rate in their prediction in almost field and problems.

### **4.3 REQUIREMENT ANALYSIS**

#### **4.3.1 SOFTWARE REQUIREMENT**

The machine learning model need to be coded using programming language that have integration with Machine Learning attribute or package. The chosen

language that will be used to code the model is python programming language. Table below shows the software detail that will be used in this project.

Software	Description
Python 3.9.5	Used to execute the python script
PyCharm 2021.1.3	Used to code the python script
Microsoft Windows 10	Operating system to run the software
Microsoft Word 2016	Used in documentation
Microsoft Excel 2016	Used in writing the analysis during training and testing the model
Draw.io online	To sketch the flowchart and diagram

**Table 10 – Software requirement**

#### 4.3.2 HARDWARE REQUIREMENT

The devices that will be used to complete this project is a laptop. Luckily, the laptop is meet the minimum requirement to run the software listed in software requirement above. Table below will list the specification for the laptop that used in this project.

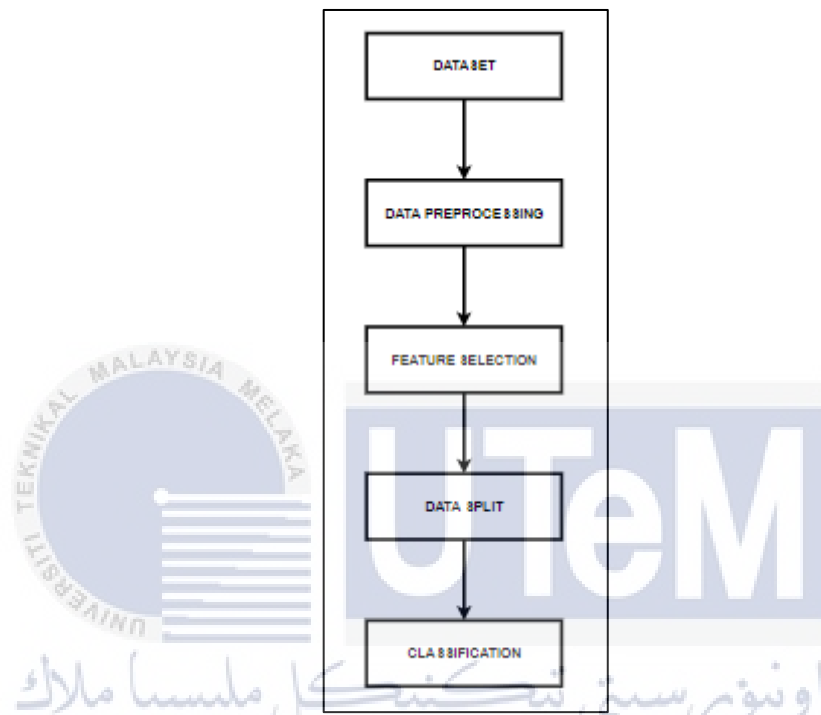
Specification	Description
CPU	Intel Core i5 6500u : 2 core, 4 thread
GPU	Nvidia GTX 930M ; 2GB GDDR4
RAM	4GB DDR3L 1600MHZ
OS	WINDOWS 10 64BIT
DISPLAY	14"
STORAGE	512GB SSD

**Table 11 – Hardware requirement**



#### 4.4 PROJECT DESIGN

To develop a machine learning model. There have a step or phase that need to follow to make sure that the model can be used to solve the problem or not. In this topic. All the phase that mentioned earlier will be describe and have further explanation. The phase will be started with dataset, feature selection, data split and then classification.



**Figure 24 – Project design in this research.**

##### 4.4.1 DATASET

Dataset is the raw of data that have been captured by previous researcher during their studies or research. This became a main reason of which model have been successfully developed. Because, in producing a machine learning model, dataset is required to train and testing the model. In this field of project. The popular way to produce the dataset is using the live network packet capture like Wireshark. To complete this project, the dataset that will be used is gained from Canadian Institute for Cybersecurity website[17]. In that page, there have listed a NSL-KDD dataset and also give a permission to download the dataset. Table below shows the dataset brief.

No	File Name
1	KDDTest+.arff
2	KDDTest+.txt
3	KDDTest-21.arff
4	KDDTest-21.txt
5	KDDTrain+.arff
6	KDDTrain+.txt
7	KDDTrain+20%.arff
8	KDDTrain+20%.txt

**Table 12 - file include during download the dataset.**

NSL-KDDTest+ DATASET	
Total Features	42
Total Instances	125973
Total Attack	58630
Total normal	67343

**Table 13 – brief for NSL-KDD Dataset**

Features	Description	Features	Description
duration	Duration of connection	su_attempted	su command entered
protocol_type	Protocol used	num_root	Switch to root user
service	Destination network service	num_file_creations	Total file creation in connection
flag	Status of connection	num_shells	Total shell entered
src_bytes	Number of bytes transfer from source in each session	num_access_files	Total operation on access files
dst_bytes	Number of bytes transfer from destination in each session	num_outbound_cmds	Outbound command in ftp connection
land	Simplified Ip address and port numbers if same for source and destination	is_host_login	If login as host
wrong_fragment	Total wrong fragment	is_guest_login	If login as guest

urgent	Urgent packet in connection	count	Total connection to the same host
hot	Indicators in the content	srv_count	Total connection to the same service
num_failed_logins	Login failed counts	error_rate	Flag activated percentage
logged_in	Status of login	srv_error_rate	Flag activated percentage
num_compromised	Total compromised condition	error_rate	Flag activated percentage
root_shell	Total gained root shell	srv_error_rate	Flag activated percentage
same_srv_rate	Percentage of same service	srv_diff_host_rate	Percentage of different destination devices
diff_srv_rate	Percentage of different service	dst_host_count	Total same destination IP address
dst_host_srv_count	Total same port number	dst_host_srv_diff_host_rate	Percentage of different destination devices
dst_host_same_srv_rate	Percentage of same service	dst_host_error_rate	Flag activated percentage
dst_host_diff_srv_rate	Percentage of different service	dst_host_srv_error_rate	Flag activated percentage
dst_host_same_src_port_rate	Percentage of same source port number	dst_host_srv_error_rate	Flag activated percentage
dst_host_error_rate	Flag activated percentage	class	Label for instances

**Table 14 – the feature description for the dataset[11].**

#### 4.4.2 DATA PREPROCESSING

The dataset that used in this research is the dataset for multi class of IDS. Based on literature review, all of researchers that used NSL-KDD dataset was using multiclass of model. Meanwhile, this research only focusses on single class of SVM which is to predict the SYN Flood DoS attack in the network. Based on [18], researchers said that for SYN flood, there have 10 relevant features. By using WEKA software, the irrelevant features for SYN flood attack have been removed and the remaining features

will be saved as new dataset. Besides, in this dataset, there are one feature that need to do categorical encoding. The type of categorical encoding that will be use in this research is One-Hot encoding. One-Hot encoding basically is the type of data transformation from a nominal to a binary form transformation. This process is needed because to train the data with model, the feature data type must be numeric except for label or class feature.

#### 4.4.3 FEATURE SELECTION

The extract features gained from feature extraction is contain an irrelevant and redundant feature because all the features from data were recorded in dataset. All the extracted features cannot be used in predicting the accuracy using machine learning model. So, from this problem, feature selection will used to overcome the problem. In this research, the Information gained (IG) algorithm is used as feature selection technique[19–21]. Generally, information gain concept is analyzing by ranking. This mean, all the feature and attribute will be ranked by IG, and it will give the result of each attribute with the ranked. Ranked mean the relation with the class attribute. The data will be selected in descending order which the attribute that give high rank to a lowest rank. The lowest rank might be zero value because it does not contribute at all in giving the prediction. Information gain is from decision tree method which consist of two algorithm that depending on each other. First is entropy and second is information gain. Information gain is the detail from entropy. The value for information gain is range from 0 to 1. This means, the feature or attribute that gain closed to 1 is consider as the high rank of attribute. Below show the formula of entropy and information gain.

$$\text{ENTROPY: } H = - \sum_{i=1}^k P_k \log_2 P_k$$

**Equation 7 – Entropy formula**

$$\text{INFORMATION GAIN: } G = H - \frac{m_L}{m} H_L - \frac{m_R}{m} H_R$$

**Equation 8 – Information gain formula**

Where: k=Class, H=Number of inputs, P\_k=Dataset, M=total number of instances

#### 4.4.4 DATA SPLITTING

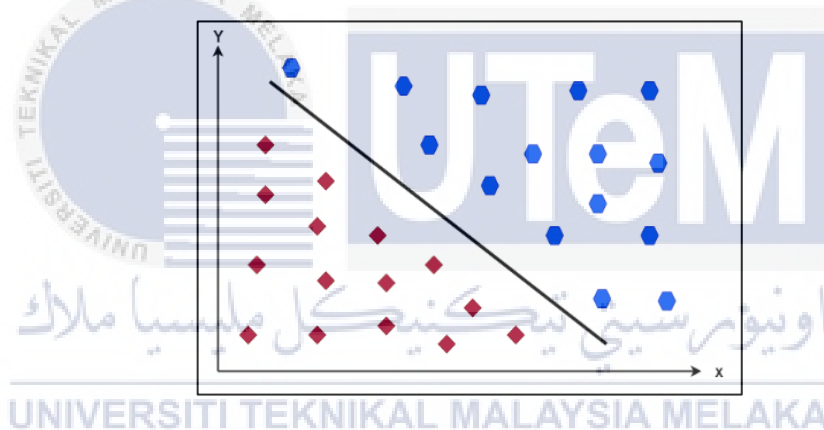
Data splitting is needed to separate the data into two category or set which is testing and training dataset. There is popular ratio that always used in data splitting which is 80-20 ratio[22]. This mean, the 80% of the data is used for training the model while another 20% is used to test the built model. The function of doing this is to avoid bias in testing the model because if the same instances is used while testing the model, it will give a bias since the model already known all the instance in dataset. The caution step of doing data splitting is by doing data randomize. Data randomize will reorder the position of instances to a random so it will avoid the same data position as original data.



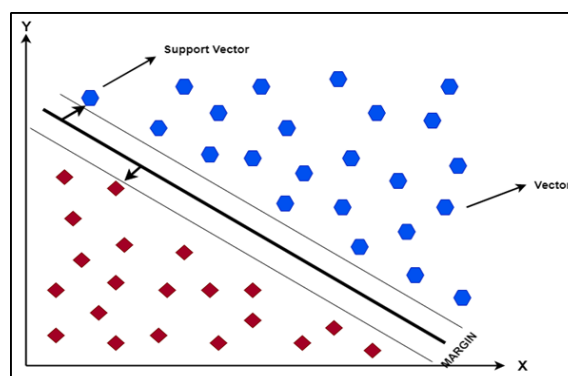
#### 4.4.5 CLASSIFICATION

Classification is the algorithms that were chosen to complete the project. There are a lot of algorithms that divided into its own category such as machine learning and deep learning. Since this project domain based on machine learning approach, so the selected algorithm or classifier is Support Vector Machine (SVM). Based on previous research, SVM is suitable in most cases because it has its own technique to differentiate the class. SVM is divided into two category which is Linear Support Vector Machine (LSVM) and Non-Linear Support Vector Machine (NLSVM).

The concept of SVM is the class is separated by hyperplane line in the middle of the vector or class. While, between the class, there have a support line to gap the class with hyperplane. The distance from support line to a hyperplane is called as margin. The support line will move further till it reach or touch any nearest class or vector. The figure below illustrated the concept of support vector machine.



**Figure 25 - black line that separate class blue and class red. The black line is called as hyperplane.**

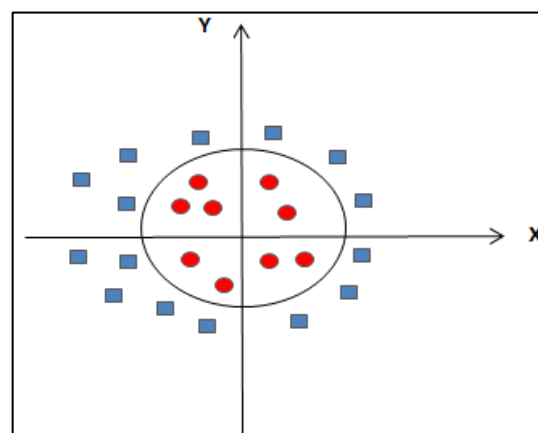


**Figure 26 - shows the support vector line to separate the class blue and red. The support line will find the nearest class. The distance between support vector line and hyperplane is called as margin.**

Both figures above showing that the class or vector is formed on linear way. But there has a case which the class cannot be divided into a linear such as 3D vector. This is where Non-Linear Support Vector Machine take the part. The concept of doing SVM is same to differentiate or separate the class using hyperplane. So, to get the hyperplane on the NLSVM, it must use a different kernel from linear. There are another 3 types of NLSVM kernel as listed in table below.

Type of Kernel	Description
Linear	Equation: $K(X,Y)=X^T Y$
Polynomial	Normally used in image processing. Equation: $k(x_i, x_j) = (x_i \cdot x_j + 1)^d$
Radial Basis Function	This kernel is a general-purpose kernel, it used when the data has no prior knowledge. Equation: $K(X,Y)=\exp(\ X-Y\ _2/2\sigma_2)$
Sigmoid	Normally used in neural networks. Equation: $K(X,Y)=\tanh(\gamma \cdot X^T Y + r)$

**Table 15 - equation of every kernel in SVM**



**Figure 27 – NLSVM example. This type of SVM cannot be done with simple match the hyperplane. So, kernel is needed to class the non-linear type of SVM**

In order to get the best or suitable kernel, all of the kernels need to be load and compared to each other. The output will give the best hyperplane accuracy. The kernel that can give high accuracy in prediction will be selected and be proposed as dos TCP syn flood detection system.

#### 4.4.6 CONCLUSION

In conclusion, this chapter is described in detail of what process that will be through during the implementation method. This will be the guide for make sure that the implementation and development phase are successful. Next chapter will be using all the knowledge in this chapter and convert it to the practical by doing from the beginning to the end of the project.





## CHAPTER 5: IMPLEMENTATION

### 5.1 INTRODUCTION

In this chapter, all the implementation and execution of the project will be discussed. All the implementation that carried in this chapter is dependent on what have been discussed in the previous chapter. The aim for this chapter is to achieve the project objectives and also to answer all the project question that asked in chapter 1. To carry this experiment, there have an environment that need to be setup and all of that will be discuss in this chapter.

### 5.2 SOFTWARE DEVELOPMENT ENVIRONMENT SETUP

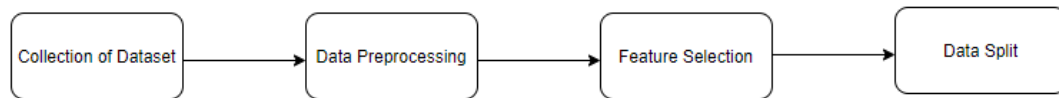
The operating system that has been used to complete this experiment is Windows Operating System this because windows is the one of the most popular OS and it also have a lot of software and hardware support and also dependent on this platform. The OS is installed on a mid-range laptop with Intel Core i5 processor.

The script has been coded in PyCharm 2021.1.3. PyCharm is the well-known editor for python programming language. This because, it can easily install another package and library that are not default by python. Besides, the written code can be debug by using another interpreter besides python. This will help for some problems because one of the disadvantages of using python is it quite slow when compared to java or C++.

For interpreter, the script has been debugging by using python 3.9 and it execute on CMD in the windows 10. The reason of using CMD is because to save the power and memory consumption by PyCharm software.

### 5.3 PROCESS MODULE

The main process or module have been spread into a different task to execute in sequences while carried out the experiment.



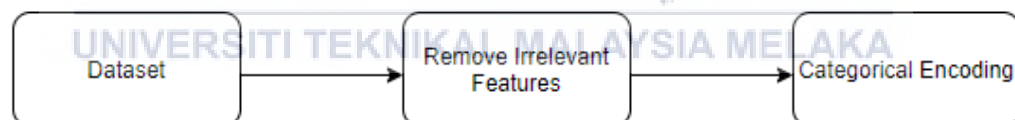
**Figure 28 – Process module task**

#### 5.3.1 COLLECTION OF DATASET

As mentioned, and explained in the previous chapter, the dataset was gained from CIC website. It has an eight file but, in this experiment, the NSL-KDD test+ will be used.

#### 5.3.2 DATA PREPROCESSING

In this step, there have 2 phase that need to be done as mentioned in previous chapter. The phase mentioned is as below:



**Figure 29 – Data Preprocessing phase**

### 5.3.2.1 REMOVE IRRELEVANT FEATURES

In this phase, WEKA software has been used to remove the irrelevant features. The main reason of using this software is because this method is simple and just a click to generate a new clean dataset. The step of doing this phase will be describe below:

Step 1: open Weka software and click on explorer

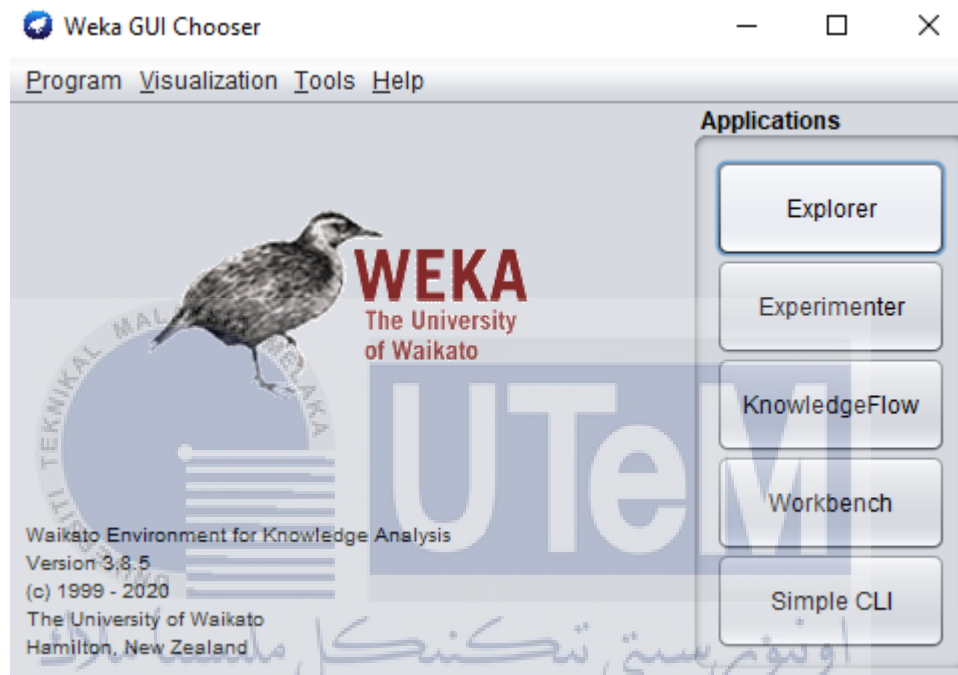


Figure 30 – Step 1 Phase 1

Step 2: click on open file and choose the dataset

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Open file... Open URL... Open DB... Generate... Undo Edit... Save...

Filter  
Choose None Apply Stop

Current relation  
Relation: KDDTrain  
Instances: 125973  
Attributes: 42  
Sum of weights: 125973

Selected attribute  
Name: class  
Missing: 0 (0%)  
Distinct: 2  
Type: Nominal  
Unique: 0 (0%)

No.	Label	Count	Weight
1	normal	67343	67343.0
2	anomaly	58630	58630.0

Class: class (Nom) Visualize All

67343 58630

Status  
OK Log x 0

Figure 31 – Step 2 Phase 1

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Step 3: tick on the irrelevant features and click remove. Weka will auto remove the ticked features.

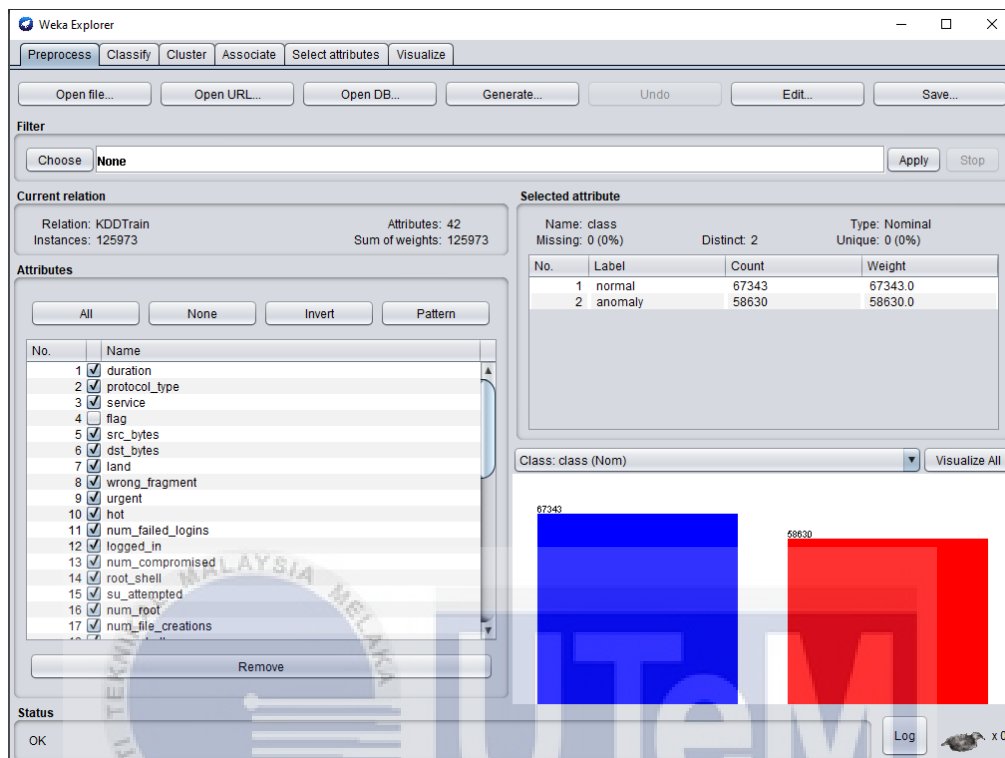


Figure 32 – Step 3 Phase 1

### 5.3.2.2 CATEGORICAL ENCODING

This last phase also will be used the WEKA software. Below shows the step of doing categorical encoding.

Step 1: click on choose and look for nominal to binary in Weka > filters > supervise > attribute and click apply.

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Open file... | Open URL... | Open DB... | Generate... | Undo | Edit... | Save...

Filter: Choose **NominalToBinary** [Apply] [Stop]

Current relation: Relation: KDDTrain-weka.filters.unsupervised.attrib... Instances: 125973 Attributes: 11 Sum of weights: 125973

Attributes: All | None | Invert | Pattern

No.	Name
1	<input checked="" type="checkbox"/> flag
2	<input type="checkbox"/> serror_rate
3	<input type="checkbox"/> srv_serror_rate
4	<input type="checkbox"/> same_srv_rate
5	<input type="checkbox"/> diff_srv_rate
6	<input type="checkbox"/> dst_host_srv_count
7	<input type="checkbox"/> dst_host_same_srv_rate
8	<input type="checkbox"/> dst_host_diff_srv_rate
9	<input type="checkbox"/> dst_host_serror_rate
10	<input type="checkbox"/> dst_host_srv_serror_rate
11	<input type="checkbox"/> class

Selected attribute

Name: flag Missing: 0 (0%) Distinct: 11 Type: Nominal Unique: 0 (0%)

No.	Label	Count	Weight
1	OTH	46	46.0
2	REJ	11233	11233.0
3	RSTO	1562	1562.0
4	RSTOS0	103	103.0
5	RSTR	2421	2421.0
6	S0	34851	34851.0
7	S1	365	365.0
8	S2	127	127.0
9	S3	49	49.0
10	S4	74945	74945.0

Class: class (Nom) [Visualize All]

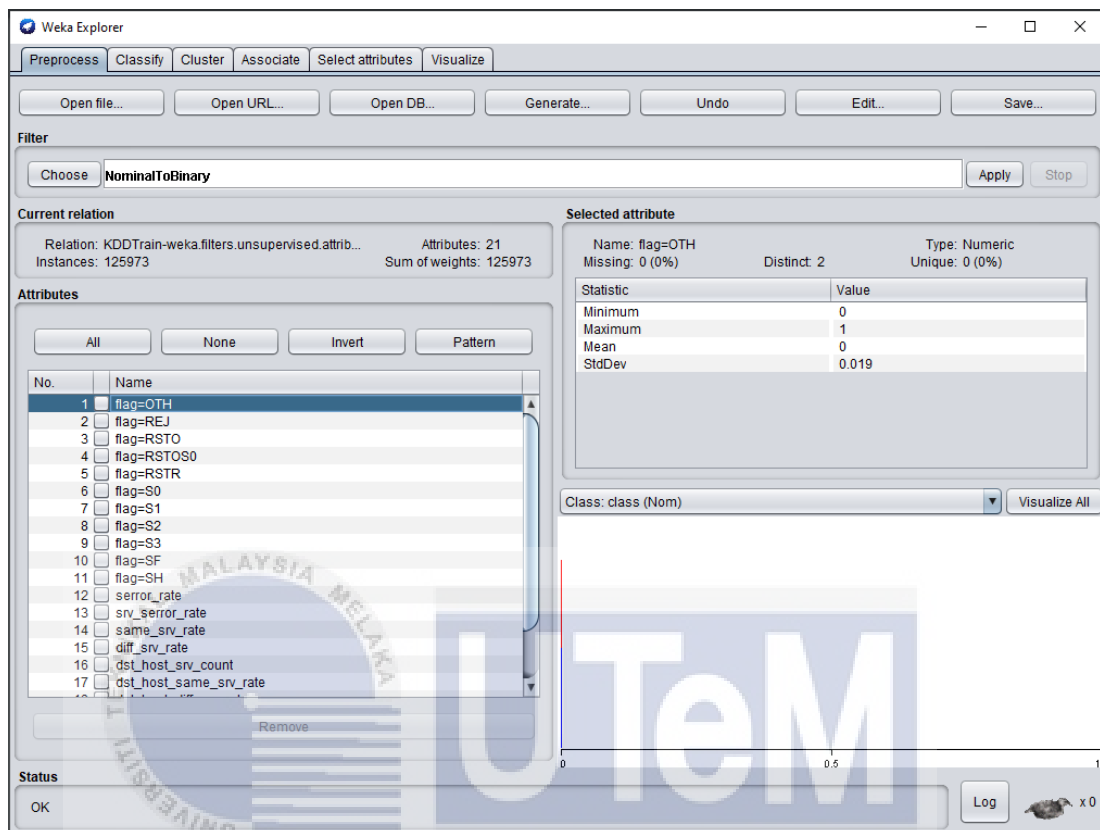
Status: OK [Log] x 0

Figure 33 – Step 1 Phase 2

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The instances in nominal features now will become the new features and next is click save to save the new clean dataset.



اونیورسیتی تکنیکا ملسیا ملاک  
Figure 34 – Complete categorical encoding

Step 2: save the new processed dataset in the selected folder and new name.

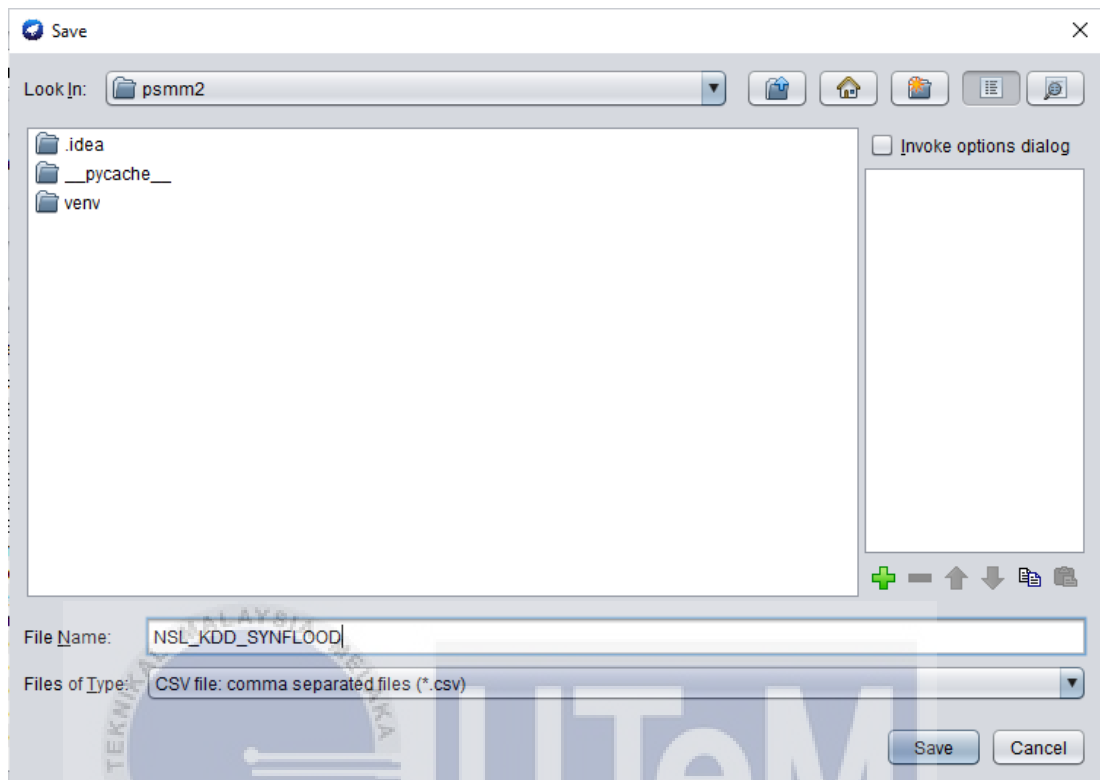


Figure 35 – Step 2 Phase 2



### 5.3.3 FEATURE SELECTION

In feature selection module, python script will be used. The type of feature selection algorithm is as mentioned in previous chapter. Below shows the script execution and process of ranking the most dependent features.

Step 1: open CMD and go to script directory. And execute the python script using 'python main.py' command.



```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19043.1165]
(c) Microsoft Corporation. All rights reserved.

D:\PYTHON\psmm2>dir
Volume in drive D is KLSCODED
Volume Serial Number is 48C6-0613

Directory of D:\PYTHON\psmm2

22/08/2021 09:34 AM <DIR>      .
22/08/2021 09:34 AM <DIR>      ..
22/08/2021 09:34 AM <DIR>      .idea
20/08/2021 12:29 AM          759 datasets.py
05/08/2021 04:00 PM          208 Date_Time_Generator.py
22/08/2021 09:20 AM          20,763 IG.txt
22/08/2021 08:27 AM           1,030 IG_D3.py
08/08/2021 03:36 PM           569 Label_Encoding.py
22/08/2021 09:34 AM           909 main.py
22/08/2021 09:33 AM       7,038,369 NSL_KDD_SYNFLOOD.csv
22/08/2021 08:27 AM           297 Save_New_Dataset.py
22/08/2021 09:19 AM       1,799 Split_Train_Test.py
22/08/2021 08:27 AM           3,221 svm.py
21/08/2021 09:49 AM           1,358 SVM.txt
18/08/2021 03:41 PM <DIR>      venv
22/08/2021 09:19 AM <DIR>      __pycache__
                11 File(s)       7,069,282 bytes
                5 Dir(s)  59,495,436,288 bytes free

D:\PYTHON\psmm2>python main.py
  
```

Figure 36 – Step 1 Feature Selection

Step 2: Once execute command entered, the following prompt will be shown. Choose the dataset that have generated in previous module.

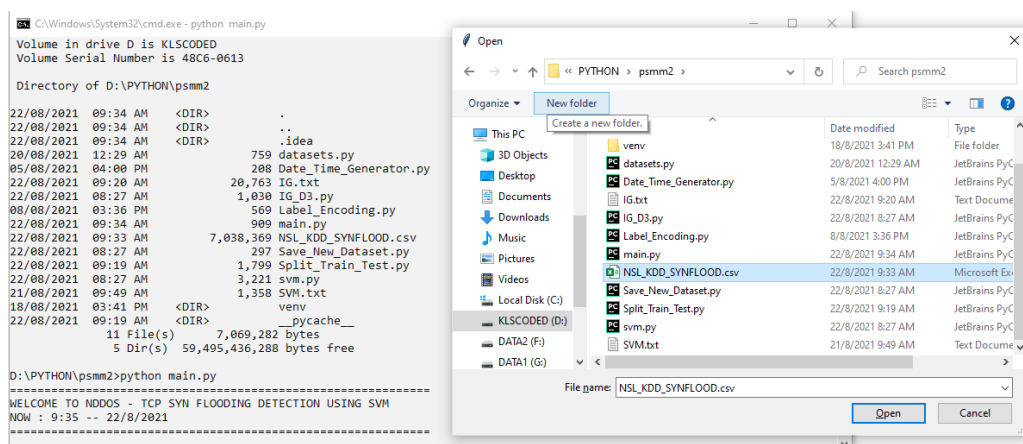


Figure 37 – Step 2 Feature Selection

Step 3: once dataset loaded, the script will automatically do the feature selection and result below shows the rank for each feature and it have been sorted from high to low rank

```

Select C:\Windows\System32\cmd.exe - python main.py
D:\PYTHON\psmm2>python main.py
=====
WELCOME TO NDDOS - TCP SYN FLOODING DETECTION USING SVM
NOW : 9:44 -- 22/8/2021
=====
same_srv_rate      0.361048
diff_srv_rate      0.360645
dst_host_srv_count 0.334798
flag=SF            0.331388
dst_host_same_srv_rate 0.307695
dst_host_diff_srv_rate 0.284379
dst_host_serror_rate 0.283828
dst_host_srv_serror_rate 0.276865
serror_rate        0.273054
srv_serror_rate    0.263995
flag=S0            0.259330
flag=REJ           0.018380
flag=RSTR          0.011181
flag=RSTO          0.001941
flag=S1            0.001631
flag=RSTOS0       0.001469
flag=SH            0.001134
flag=S3            0.000290
flag=S2            0.000000
flag=OTH           0.000000
dtype: float64
=====
Please enter the top features list to update the dataset :

```

Step 4: Enter the number of top features that will be used in train the model. This function will generate the new dataset according to the number of features that inserted.

```

C:\Windows\System32\cmd.exe - python main.py
D:\PYTHON\psmm2>python main.py
=====
WELCOME TO NDDOS - TCP SYN FLOODING DETECTION USING SVM
NOW : 9:44 -- 22/8/2021
=====
same_srv_rate      0.361048
diff_srv_rate      0.360645
dst_host_srv_count 0.334798
flag=SF            0.331388
dst_host_same_srv_rate 0.307695
dst_host_diff_srv_rate 0.284379
dst_host_serror_rate 0.283828
dst_host_srv_serror_rate 0.276865
serror_rate        0.273054
srv_serror_rate    0.263995
flag=S0            0.259330
flag=REJ           0.018380
flag=RSTR          0.011181
flag=RSTO          0.001941
flag=S1            0.001631
flag=RSTOS0       0.001469
flag=SH            0.001134
flag=S3            0.000290
flag=S2            0.000000
flag=OTH           0.000000
dtype: float64
=====
Please enter the top features list to update the dataset : 6_

```

Step 5: Once the new dataset is read. User will be prompted to key in the new dataset that have been remove the irrelevant features. This will be useful in future as references.

```
C:\Windows\System32\cmd.exe - python main.py
-----
same_srv_rate      0.361048
diff_srv_rate     0.360645
dst_host_srv_count 0.334798
flag=SF           0.331388
dst_host_same_srv_rate 0.307695
dst_host_diff_srv_rate 0.284379
dst_host_serror_rate 0.283828
dst_host_srv_serror_rate 0.276865
serror_rate       0.273054
srv_serror_rate   0.263995
flag=S0           0.259330
flag=REJ         0.018380
flag=RSTR        0.011181
flag=RSTO        0.001941
flag=S1          0.001631
flag=RSTOS0     0.001469
flag=SH          0.001134
flag=S3         0.000290
flag=S2         0.000000
flag=OTH        0.000000
dtype: float64
-----
Please enter the top features list to update the dataset : 6
-----
Enter new CSV name without Extension : psm2_
```



اونيورسيتي تيكنيكل مليسيا ملاك

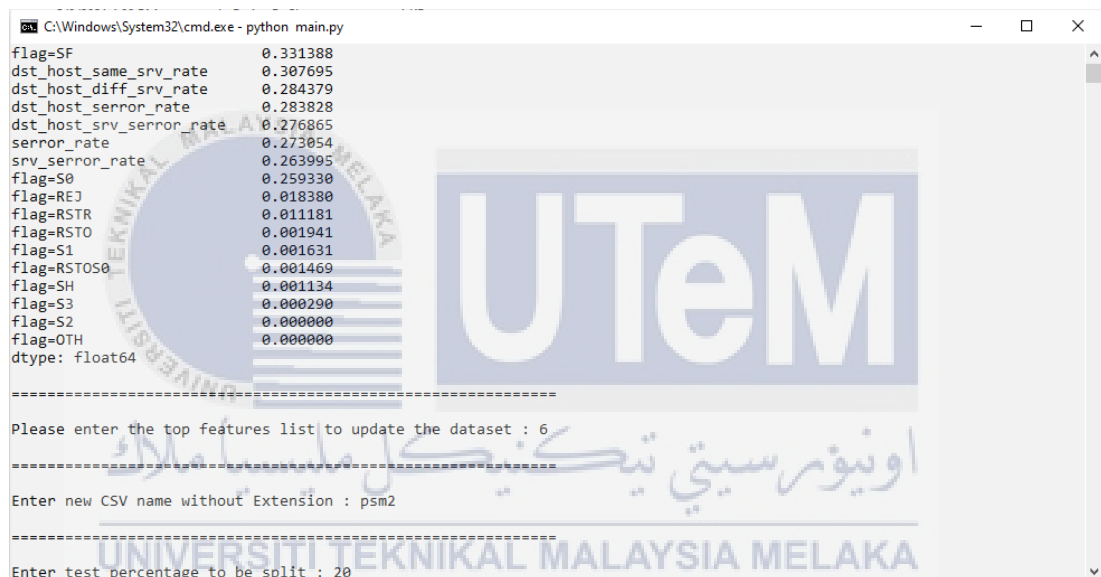
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

### 5.3.4 TRAIN AND TEST DATA

#### 5.3.4.1 SPLITTING DATA

As mentioned in previous chapter, data split is to split the dataset into two which is train and test. The rate of split that will be used is 80% for training the model while another 20% is for testing the trained model. Below shows the process of data split that continue from previous module. This because the script has been set into an order to ease the process.

Step 1: key in 20 and enter



```

C:\Windows\System32\cmd.exe - python main.py
flag-SF          0.331388
dst_host_same_srv_rate  0.307695
dst_host_diff_srv_rate  0.284379
dst_host_serror_rate   0.283828
dst_host_srv_serror_rate 0.276865
serror_rate        0.273054
srv_serror_rate     0.263995
flag-S0           0.259330
flag-REJ         0.018380
flag-RSTR        0.011181
flag-RSTO        0.001941
flag-S1          0.001631
flag-RSTO50     0.001469
flag-SH          0.001134
flag-S3         0.000290
flag-S2         0.000000
flag-OTH        0.000000
dtype: float64
=====
Please enter the top features list to update the dataset : 6
=====
Enter new CSV name without Extension : psm2
=====
Enter test percentage to be split : 20

```

**Figure 38 – Step 1 of splitting data**

Step 2: After finish split the dataset, there have a prompt shown that ask user if want to view the summarize of dataset. Click yes or y to view.

```

=====
Enter test percentage to be split : 20
Press y/yes to view summary of train and test data (y/n): y
Before Split summarize : (125973, 7) , Normal : 67343, Anomally : 58630
Train Data Summarize : (100778, 7) , Normal : 53842, Anomally : 46936
Test Data Summarize : (25195, 7) , Normal : 13501, Anomally : 11694
=====

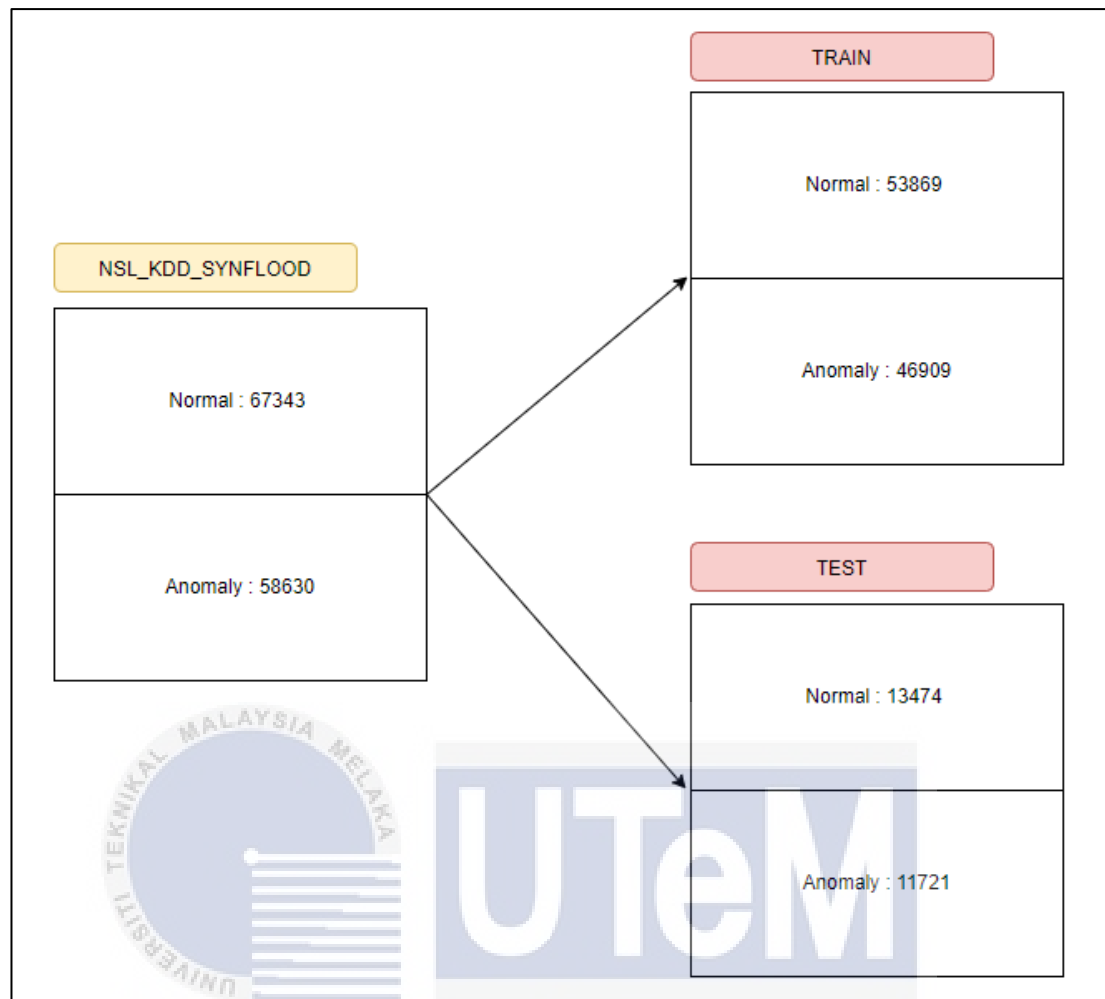
```

**Figure 39 – Step 2 of splitting data**

After through this process, the original dataset has been split into two which 80% of the total data will be train dataset and rest will be the test dataset. Table below indicate the summarize of dataset mentioned above.

Dataset	Total Instances	Normal	Anomaly
NSL_KDD_SYN Flood	125973	67343	58630
TRAIN	100778	53869	46909
TEST	25195	13474	11721

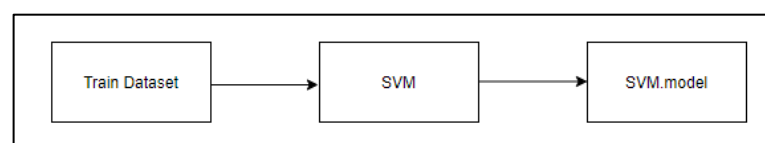
**Table 16- Shows the summarize of dataset**



**Figure 40 – Shows the summarize of data after split**

#### 5.3.4.2 MODEL FILE

Outcomes from previous split train data will be used to generate the train model. The generated train model can be used in testing the model to measure that it can be used in predicting by using the test data. Besides, the generated train model can also be implemented or deploy in future. This because, the train model has studied the behavior of train data and it will predict based on the studied that have been through during training the model.



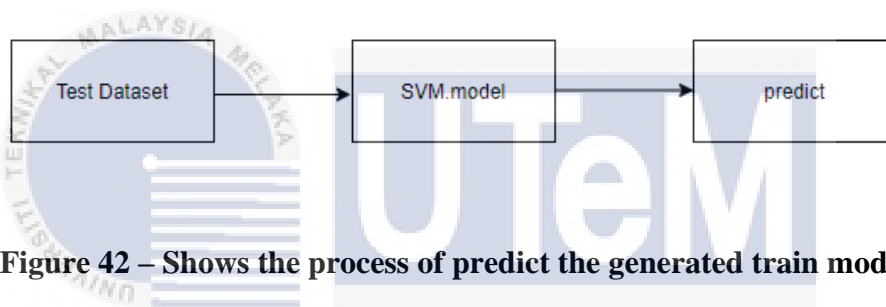
**Figure 41 – Shows the process of generate the model**

### 5.3.4.3 PREDICT FILE

As mentioned before, the generated train model will be used to test the model by using test data that have been generated in split process. Through this process, the classification report and confusion matrix will be generated to indicate the result and accuracy of the test process. In confusion matrix, the accuracy can be measured by using below formula.

$$ACCURACY = \frac{(TRUE POSITIVE + TRUE NEGATIVE)}{(TRUE POSITIVE + FALSE POSITIVE + TRUE NEGATIVE + FALSE NEGATIVE)}$$

**Equation 9 – Accuracy formula**



**Figure 42 – Shows the process of predict the generated train model**

## 5.4 CLASSIFICATION اونیورسیتی تیکنیکل ملیسیا ملاکا

In this phase, the process of train and testing the model is executed by the script that have been coded and its run according to the flow of all the experiment. First, lets review the code to see the parameter that involved in the model configuration.

### 5.4.1 CODE REVIEW

```
model = SVC(kernel="rbf", C=1, gamma="auto", random_state=0)
```

**Figure 43 – Model code review**

Parameter	Description
SVC()	Call the support vector machine constructor.
Kernel = "rbf"	Using the Radial Basis Function kernel.
C = 1	C is the must parameter in SVM. In this experiment, the C value have been set to 1.
Gamma = "auto"	Besides C, gamma also is the main parameter that need to be put in SVM. In this experiment, Gamma value have been set to auto.
random_state = 0	Random state is the value for randomizes the dataset. In this experiment, the value is set to 0 because the script has written the randomization code upon uploading the dataset.

**Table 17 – Parameter description**



```

kf = KFold(10)
train_Result = cross_val_score(model, X=X_train, y=y_train, cv=kf, n_jobs=-1, verbose=1)

```

**Figure 44 – Train result code review**

Parameter	Description
KFold (10)	Call the KFold constructor with passing the 10-fold parameter.
Cross_val_score ( )	Call the cross-validation score constructor. On this process, the model that have been set is fit to the pass x and y data.
model	Model is the variable for SVM algorithm
X = X_train	Give the X data
y = y_train	Give the y data
Cv = kf	State the cross-validation process. In these cases, the cv has been set as Kfold 10 which mean in this experiment, it uses KFold Cross-Validation in train the model.
N_jobs = -1	N_jobs indicate how many core that want to use during train the model. in this experiment, the jobs is set to -1 which mean use all the core and thread in the CPU.
Verbose = 1	Verbose is the information on behind training process. The value is set to 1.

**Table 18 – Training model parameter description**

```
y_predict = cross_val_predict(model, X=X_test, y = y_test)
```

**Figure 45 – Generate the y\_predict for confusion matrix**

Parameter	Description
Cross_val_predict ( )	Call the cross validation predict constructor. During this constructor, the model will be set to predict mode.
Model	Model is the variable for SVM algorithm
X = X_test	Use the test dataset to test the model
Y = y_test	Use the test dataset to test the model

**Table 19 – y\_predict description**

After the y\_predict have been generate, classification report will be executed by comparing the predict of y-axis or class and the original of class in the test dataset.

```
print(classification_report(y_test, y_predict))
```

**Figure 46 – Shows the code for print the report**

Last of the process, the script has been set to print the confusion matrix by compare the predict of class and the original class file. Outcome for this process is it state the TP, FP, TN and FN that can be used to calculate the accuracy.

```
print(confusion_matrix(y_test, y_predict))
```

**Figure 47 – Shows the code for print the confusion matrix**

### 5.4.2 EXECUTE

As mentioned earlier, the script has been set to follow the flow of all experiment. Upon finish the split process, the script will auto train and test the model. Below shows the process of train and testing the model.

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers.
[Parallel(n_jobs=-1)]: Done 10 out of 10 | elapsed: 89.4min finished
```

**Figure 48 – Shows the process of train model**

## 5.5 RESULT

In this phase, the documentation of the train and test process will be discussed briefly. As shown in previous part, from IG feature selection, the dataset has been customized to a less attribute that raw one. Which the total features have 7 include class and the total of instances as stated in table 16.

### 5.5.1 RESULT OF TESTING MODEL

```
=====
Enter the model name to save : SVM
=====
```

```
Average Train Result accuracy : 0.9113
=====
```

**Figure 49 – Show the prompt save file name and the average of the training accuracy**

Once the test process finished, the script will auto produce the classification result and confusion matrix. This will prove the model can be used or not in predicting the anomaly packet in the dataset. Confusion matrix that produced indicate the TN, FN, TP and FP in the testing phase. Outcomes from this, the accuracy can be measured using the formula in equation 9.

```

=====
Classification Report :
              precision    recall  f1-score   support

   anomaly      0.97      0.82      0.89     11694
   normal      0.86      0.98      0.92     13501

 accuracy      0.91      0.91      0.91     25195
 macro avg      0.92      0.90      0.90     25195
 weighted avg   0.91      0.91      0.91     25195

=====

Confusion Matrix :
[[ 9623  2071]
 [  280 13221]]

```

**Figure 50 – Shows the report and confusion matrix from test process.**



### 5.5.2 ACCURACY TABLE

Type	Result
Accuracy	0.91
True Normal Predicted	9623
True Anomaly Predicted	13221
False Normal Predicted	2071
False Anomaly Predicted	280

**Table 20 – Accuracy table**

Based on accuracy table above, an experiment carried with SVM model, and the kernel RBF with C value is 1 and the gamma is set to auto gain the high accuracy in predicting the class. It achieved 91% of the accuracy. The consistency of the train also can be seen in figure 49. Which in 10 runs, the highest accuracy is 91.39% while the lowest accuracy is 90.88% with the differences only 0.51%.

### 5.6 CONCLUSION

As conclusion, based on accuracy gain in the test model, this model can be deployed and implement in the real world and can be used by anyone to minimize the threat in the network. Based on the table 20, this experiment gains the high percentage of predicting accuracy which is 91% with RBF kernel and used IG as feature selection. In the next chapter, it will discuss further about the experiment result and also it will conclude that whether the experiment that have been carried has meet the project objective or not.

## CHAPTER 6: DISCUSSION

### 6.1 INTRODUCTION

In past chapter, all the requirement, environment, implementation and obtained result have been discussed clearly include the lab environment preparation and needs to run this experiment. Besides, it also discusses regarding the module of process that required to finish this experiment starting from dataset, data processing, feature selection and also generating train and test data. Also, the results of the train and test process have been recorded and it was displayed using a table to ease reader while read this documentation.

In this chapter, it will discuss the flow of this experiment from start to finish together with the result that obtain during previous chapter. In addition, this chapter also will discuss regarding the proposed model and ensure that it meet all the objective as stated in chapter 1 or not. The comparison of the proposed model will be carried with the training and testing result.

### 6.2 DISCUSSION OF THE PROJECT

In past decade, Internet of everything industry was lead the world in many fields. Starting from the internet connection was widely used by people, now most of the services was depending on the internet and currently world is leading to Industrial Revolution 4.0 which on that century, all services in the world will depend on network and internet connection. This evolution will expose the service providers to a network threat. This research project implements three stages of process to detect the DoS activities in the network. The first stages are to do the data preprocessing, second is feature selection by using Information Gain and last one is classification to train and testing the model. The dataset was obtained by CIC website, and it is public to all users.

The first stages were carried is to clean the dataset this because the cleanest the dataset, the good the model will be produced. These stages required two phases which is remove the irrelevant features. This phase will help to compress the dataset to minimize the stress of the model while training. Because of this project scope only focuses on TCP Syn Flooding detection, while dataset obtain more than one type of

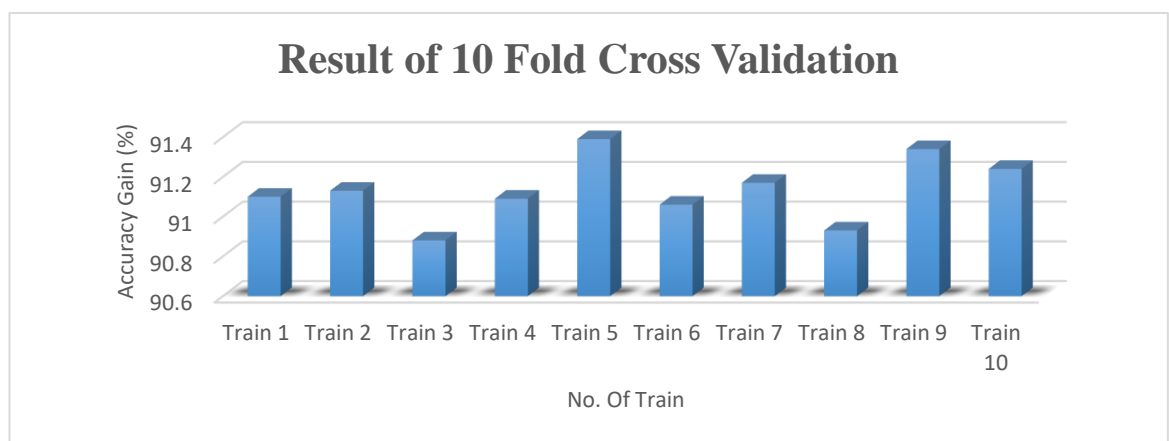
network threat. Next is, categorical encoding which One-Hot encoding. This phase also called as Nominal to Binary which mean all the Nominal grouped data type is convert to a binary form except for class features. This because, the model cannot read through features except Numerical data type.

Next stages are Feature Selection. In this phase, the algorithm that was used is Information Gain. This phase will rank all the features from high to low result. This can be defined as the highest the rank, the most dependent the features to determine class. Together with this phase, the script will auto generate the new data to a top highest rank according to user wants. Outcomes for this phase will make the next last phase easier and light because it does not have many features and it also will make the next phase become faster.

Lastly, to classify the data, SVM algorithm was used. This phase required two phases. Which training and testing. By using the same data, the script will also ask the user how much percentage for test data. The data now will split into two which is train and test. Outcomes for this project will produced the accuracy result in determine the anomaly activities in the data.

### 6.3 DISCUSSION ON THE PROPOSED METHOD

This topic will compare the result of the train and test data. The train process was using 10-Fold of cross validation method to see the effectiveness of the proposed model. The result of all fold process was recorded in the bar graph form as below.



**Figure 51 – 10-Fold Cross Validation score**

Based on 10-Fold run, the consistency of the proposed model during training phase can be seen by the result. From 1 to 10-fold, the result has a minimal difference with minimum point is 90.88% and the maximum is 91.39%. The differences of between these two points is only 0.51% and it's recorded the mean of all train is 91.13%. This average can be considered higher because it achieved more than 90% while training. The trained model then was used to predict the test dataset. This process is called as testing the model to see whether it is a good model and have an enough knowledge in predicting the class based on learned behavior during train process. The result with the details of classifier method will record in the form below.

Type	Description
Accuracy	91%
Kernel	RBF
C	1
Gamma	auto

**Table 21 – Shows the description of the proposed model**

#### 6.4 CONCLUSION

In conclusion, the proposed model can be considered good in predicting the class of the data based on learnt behavior of each train data during training process. The accuracy of the model can be improved by using hyperparameter tuning method. But this method cannot be done because of some limitation. This will be discussed further in the next chapter.



## CHAPTER 7: PROJECT CONCLUSION

### 7.1 INTRODUCTION

The previous chapter was discussed the whole of research phases and experiment together with result. In this chapter, it will summarize the whole research and also it will discuss the constraint, contribution and also limitation of this research. Besides, it also will plan for future work as closure for this research. To remind that this chapter will be the last chapter for the documentation report.

### 7.2 PROJECT SUMMARY

As stated earlier, during Industrial Revolution 4.0. world is depending on the network and the internet. So, a detection model of network threat us required to minimize the threat in future. Our world still not reach that era, but we are leading to it. So, it still has a lot of time to keep improving our security so that if we come to that era, our model is ready to deploy to protect the network and minimize the risk of getting threat by bad people.

This experiment is carried by using a machine learning algorithm by learning the behavior of each instance during training process. During the testing process, the trained model will be used to predict the new test dataset and it is measured by accuracy of predicting the class. The concept of predicting is the trained model is used to predict the new class based on test dataset. Then the predicted class is compared to the original test class to produce the confusion matrix. By this, the accuracy can be calculated and measure using a formula in equation 9.

### **7.3 PROJECT CONSTRAINT**

This project faced several constraints during execute the experiment such as time-consuming and resources-consumption because this project requires a high-end workstation and have a high-end CPU model. As stated in previous chapter, due to these constraints, this project cannot achieve max accuracy because it cannot do the hyperparameter tuning. This method will be comparing the several parameters that was set in the dictionary list and it will run all the parameter and give the best combination of parameters. Due to this constraint, this project only restricted to RBF kernel with stated C and gamma value.

### **7.4 PROJECT CONTRIBUTION**

The contribution to the project is to proposes the Machine Learning script that can be used in predicting the anomaly activities in the network and this effectiveness of the proposed script is measured using accuracy.

### **7.5 PROJECT LIMITATION**

As discussed earlier, this research project only focusses on the TCP Syn Flooding attack. Besides, this project only uses the dataset that produced by NSL-KDD that obtained in the CIC website. Finally, this research only done by using SVM algorithm with the stated parameter.

### **7.6 FUTURE WORK**

In future, this project can be upgrade and improve by implement the hyperparameters tuning to get the max accuracy value. Besides, it also can be improved by increased the type of attack in network such as network scan, buffer overflow, phishing attack and etc. This can be called as multiclass SVM classifier.

### **7.7 CONCLUSION**

To sum up, this experiment has met all the three objectives and have answered all the project question. However, it still needs to improve a lot since this experiment facing limitation in terms of hardware limitation. Upon this problem have been patched, the new model with higher accuracy score will be gain.

## REFERENCES

- Basyir, M. (2021, July 16). Malaysians suffered RM2.23 billion losses FROM cyber-crime frauds: New Straits Times. NST Online. <https://www.nst.com.my/news/crime-courts/2021/07/708911/malaysians-suffered-rm223-billion-losses-cyber-crime-frauds>.
- [1] Bogdanoski M, Shuminovski T, Risteski A. TCP SYN Flooding Attack in Wireless Networks TCP-SYN Flooding Attack in Wireless Networks. DOI: 10.13140/2.1.3487.3282.
- [2] GADE VAIBHAV KRISHNA. *Intrusion Detection System as a Service GADE VAIBHAV KRISHNA Providing Intrusion Detection System on a subscription basis for cloud deployment*, www.bth.se.
- [3] Mazhar N. *Signature-based intrusion detection using NFR filter coding*, <https://lib.dr.iastate.edu/rtd> (2000).
- [4] Chan P. *Signature Based Intrusion Detection Systems*.
- [5] Jose S, Malathi D, Reddy B, et al. A Survey on Anomaly Based Host Intrusion Detection System. In: *Journal of Physics: Conference Series*. Institute of Physics Publishing, 2018. Epub ahead of print April 26, 2018. DOI: 10.1088/1742-6596/1000/1/012049.
- [6] Kappes M, Hock D, Kappes M. A Self-Learning Network Anomaly Detection System using Majority Voting. Epub ahead of print 2014. DOI: 10.13140/2.1.2448.5761.
- [7] Milad Helalat S. *An Investigation of the Impact of the Slow HTTP DOS and DDOS attacks on the Cloud environment*, www.bth.se.
- [8] Mishra S, Gogoi M. *DETECTING DDoS ATTACK USING Snort Implementation of the Quoted-Printable Conversion Algorithm View project Security issues in Block chain and crypto currency View project DETECTING DDoS ATTACK USING Snort*, <https://www.researchgate.net/publication/338660054>.
- [9] ACLs S. *75-2 Chapter 75 Denial of Service (DoS) Protection Security ACLs and VACLs*.
- [10] Eklund M. *Comparing Feature Extraction Methods and Effects of Pre-Processing Methods for Multi-Label Classification of Textual Data*. 2018.
- [11] Dhanabal L, Shantharajah SP. A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*; 4. Epub ahead of print 2015. DOI: 10.17148/IJARCCE.2015.4696.
- [12] She C, Wen W, Lin Z, et al. Application-Layer DDOS Detection Based on a One-Class Support Vector Machine. *International Journal of Network Security & Its Applications* 2017; 9: 13–24.
- [13] Daneshgadeh S, Baykal N, Ertekin S. DDoS Attack modeling and detection using SMO. In: *Proceedings - 16th IEEE International Conference on Machine Learning and Applications, ICMLA 2017*. Institute of Electrical and Electronics Engineers Inc., 2017, pp. 432–436.
- [14] Prasad MD, V PB, Amarnath C. Machine Learning DDoS Detection Using Stochastic Gradient Boosting. *International Journal of Computer Sciences and Engineering* 2019; 7: 157–166.
- [15] Mašetić Z, Kečo D, Dođru N, et al. SYN flood attack detection in cloud computing using support vector machine. *TEM Journal* 2017; 6: 752–759.

- [16] Sambangi S, Gondi L. A Machine Learning Approach for DDoS (Distributed Denial of Service) Attack Detection Using Multiple Linear Regression. *Proceedings 2020*; 63: 51.
- [17] NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB, <https://www.unb.ca/cic/datasets/nsl.html> (accessed June 24, 2021).
- [18] Nouredien NA, Yousif IM. Accuracy of Machine Learning Algorithms in Detecting DoS Attacks Types. *Science and Technology* 2016; 6: 89–92.
- [19] Azhagusundari B, Thanamani AS. *Feature Selection Based on Information Gain*. 2013.
- [20] Zargari S. *Feature Selection in the Corrected KDD-dataset*, <http://shura.shu.ac.uk/17048/3/Zargari%20Feature%20Selection%20in%20the%20Corrected%20KDD.pdf> (accessed June 24, 2021).
- [21] Elektronik J, Udayana IK, Bagus G, et al. Implementation of Feature Selection using Information Gain Algorithm and Discretization with NSL-KDD Intrusion Detection System, <http://nsl.cs.unb.ca/NSL-KDD/>.
- [22] Andersson C, Ortiz ML. *RESERVOIR COMPUTING APPROACH FOR NETWORK INTRUSION DETECTION* Examiner: Sasikumar Punnekkat.



**APPENDIX A – PROJECT GANN CHART**

PHASE	TASK	WEEK																								
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Planning	Project Proposal	█	█	█																						
	Chapter 1 – Introduction				█	█	█	█	█	█																
Analysis	Chapter 2 – Literature Review																									
Design	Chapter 4 – Analysis And Design																									
Implementation	Chapter 5 – Implementation																									
Discussion	Chapter 6 – Discussion																									
Conclusion	Chapter 7 – Project Conclusion																									

**Table 22 – Project Gann Chart**