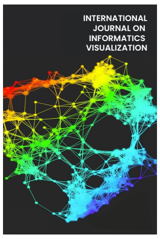
INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv



Feature Selection to Enhance DDoS Detection Using Hybrid N-Gram Heuristic Techniques

Andi Maslan [a,*], Kamaruddin Malik Bin Mohamad [b], Abdul Hamid [c], Hotma Pangaribuan [d], Sunarsan Sitohang [e]

[a] *Department of Informatic Engineering, Universitas Putera Batam, Batam, Indonesia*

[b] *Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Batu Pahat Johor, Malaysia*

[c] *Faculty of Technical and vocational Education, Universiti Tun Hussein Onn Malaysia, Batu Pahat Johor, Malaysia*

[d] *Universitas Putera Batam, Indonesia*

*Corresponding author: *lanmasco@gmail.com*

***Abstract*—Various forms of distributed denial of service (DDoS) assault systems and servers, including traffic overload, request overload, and website breakdowns. Heuristic-based DDoS attack detection is a combination of anomaly-based and pattern-based methods, and it is one of three DDoS attack detection techniques available. The pattern-based method compares a sequence of data packets sent across a computer network using a set of criteria. However, it cannot identify modern assault types, and anomaly-based methods take advantage of the habits that occur in a system. However, this method is difficult to apply because the accuracy is still low, and the false positives are relatively high. Therefore, this study proposes feature selection based on Hybrid N-Gram Heuristic Techniques. The research starts with the conversion process, package extract, and hex payload analysis, focusing on the HTTP protocol. The results show the Hybrid N-Gram Heuristic-based feature selection for the CIC-2017 dataset with the SVM algorithm on the CSDPayload+N-Gram feature with a 4-Gram accuracy rate of 99.86%, MIB- Dataset 2016 with the 2016 algorithm. SVM and CSPayload feature +N-Gram with 100% accuracy for 4-Gram, H2N-Payload Dataset with SVM Algorithm, and CSDPayload+N-Gram feature with 100% accuracy for 4-Gram. As a comparison, the KNN algorithm for 4-Gram has an accuracy rate of 99.44%, and the Neural Network Algorithm has an accuracy rate of 100% for 4-Gram. Thus, the best algorithm for DDoS detection is SVM with Hybrid N-Gram (4-Gram).**

***Keywords*— Chi-square distance; DDoS; Heuristic; N-Gram; Payload**

Manuscript received 31 Dec. 2022; revised 3 Feb. 2023; accepted 7 Jul. 2023. Date of publication 10 Sep. 2023.

I. INTRODUCTION

Currently, DDoS is a type of cyber-attack that can attack any website, be it a personal website, school website, online shop, or even an enterprise-level website. These attacks also continue to evolve as technology evolves. The target of the attack is from layer two to layer seven, where the server will receive and respond to Hypertext Transfer Protocol (HTTP) requests and load the website page. This category of attacks tends to be challenging to identify and overcome because they resemble natural web traffic.

Statistical statistics, such as the quantity, size, and length of data packets, are commonly used by researchers to analyze traffic. The traditional method of performing traffic analysis in the case of detecting DDoS assaults is to convert packet units to flow units based on packet sets of the same 5-tuple (Source IP, Source Port, Destination IP, Destination Port, Transport Layer Protocol) [1]. However, detecting DDoS attacks based on the HTTP protocol receives little attention because it cannot be analyzed until flow generation is complete. There is an additional cost disadvantage for calculating the statistical information of the flow occurring. Therefore, several methods can detect DDoS attack types, both bandwidth depletion and resource depletion attacks. The primary focus of detecting DDoS assaults is bandwidth depletion related to the quantity and kind of packets sent and received, which has a high false positive rate. While some scholars have lately used machine learning approaches to identify network attacks and other anomalies, others have published methodologies based on a statistical analysis of Management Information Base (MIB) and Canadian Institute for Cybersecurity (CIC) data. Other studies reviewed related work on anomaly detection using the SNMP-MIB, and CIC-2017 datasets, such as research conducted by Alkasassbeh et al. [2], which proposes a new dataset that includes modern attack types not used in previous studies. The suggested approach uses 91 MIB traffic features from 5 categories (IP, ICMP, TCP, UDP, and SNMP) that are periodically gathered from targets and attackers participating in the attack. Ping Flood, Targa3, and UDP Flood are three DDoS assaults that use controlled traffic loads.

In addition, the pattern recognition of DDoS attacks on IDS has two disadvantages. First, TCP/IP deficit [3] for hackers, DDoS attacks are easy to start, while the victims are hard to realize. In addition, DDoS attacks have developed a new technique; an example is the SYN-Flood attack. In general, a single SYN packet is a legal packet of network activity that is difficult to detect as a strange artifact by IDS. Therefore, IDS is challenging enough to generate a warning about whether SYN-Flood is attacking the network[4]. Second, false-positive alert issues in signature-based IDS frequently occur when standard network patterns are wrongly identified as DDoS attacks. As a result, when a DDoS attack occurs, it is imperative to quickly identify and take mitigation measures to secure networks that cannot function properly.

While the type of resource depletion attack, for example, [5] proposes Payload Based Signature Generation to detect DDoS attacks based on the similarity of the two payloads compared to the Similarity-Based Classification approach, classification based on similarity treats payloads as strings. It investigates methods for correlating those payloads based on similarity in structure and content. This classification aims to group related payloads, part of an attack with a different variant from other traffic.

II. Material and Method

A. DDoS detection Methods Category

These websites have been unavailable for several hours since the first DDoS attack in 2000, which caused damage to websites for companies including Amazon, CNN, eBay, and Yahoo. Researchers in network security are constantly looking at ways to stop assaults like this one. Several techniques, including statistical, knowledge-based, software computing-based, data mining-based, and machine learning-based [6], for detecting and preventing DDoS attacks. Like previous research [7], byte-level HTTP traffic analysis offers a practical solution to the problem of network intrusion detection and traffic analysis problems.

This study emphasizes statistical-based and knowledge-based methods with the N-Gram technique in detecting attacks on the HTTP protocol [8]. General attacks, Shell Code, and CLET datasets are the types of attacks detected. Attacks are detected based on the results of attack simulations by making HTTP requests to the server by sending normal packets as raw data. The subsequent request is by entering the shell code of the attack, then the normal raw data with the attack is compared by calculating the Chi-Square Distance and Pattern Counting values. This technique provides a better detection rate for 1-Gram 0.1107 milliseconds, 2-Gram 0.6599 milliseconds, 3-Gram 14,9650 milliseconds, 4-Gram 18.0545 milliseconds and 5-Gram 37.8059 milliseconds and is faster and more efficient than HMM-based techniques [9]. However, in conducting the analysis using the outdated DARPA'99 dataset, thus new types of attacks cannot be detected in this study.

An intelligent method for identifying DDoS attack patterns was generated by network packet analysis and the application of machine learning [10]. The Center for Applied Internet Data Analysis provided many network packets for analysis in this study. They use the SVM technique to build a detection system, with Radial Kernel as the main goal (Gaussian). This study set up 4,000 IP addresses, 2,000 from the attacker pool and 2,000 from the victim pool, and four attributes as test data. The detection system has an overall accuracy rate of 85% for detecting DDoS attacks and an accuracy rate of 98.7% using five features. The system creation method for detecting DDoS attacks shows that the system using SVM was successfully trained using the recommended features to identify DDoS attacks with high accuracy.

Improved detection of Distributed Denial of Service attacks has been proposed based on fast entropy methods and flow-based analysis [11]. Compared to traditional computing, Fast Entropy and flow-based significantly reduce computation time while maintaining high detection accuracy. Entropy calculation per flow is performed after network traffic analysis on the fly demand. When the entropy difference of the flow calculation from the average entropy value over that period exceeds a threshold value modified adaptively based on traffic pattern conditions to improve detection accuracy, a DDoS attack is detected. This paper suggests three techniques for identifying DDoS: Fast Entropy, flow aggregation, and adaptive threshold. This adaptive threshold technique improves detection accuracy while reducing computation time compared to traditional entropy. The relationship between 192.95.27.190 and 71.126.222.64, e.g., The resultant value of 7.46 compared to the other relationships, is significant. However, because this approach performs forward tracking, previously found packets cannot be inspected again.

Machine learning-based methods for identifying DDoS attacks have been researched [2], [12],[13], along with new information about recently used and unresearched attack variants. The dataset consists of 27 attributes and five classes. Network Simulator (NS2) is used in this work because it can be used with good results and is quite reflective. Many attacks targeting the application and network layers have data recorded for them. Three machine learning techniques, MLP, Random Forest, and Naive Bayes, were used to classify Smurf, UDP-Flood, HTTP-Flood, and SIDDOS types from the obtained data set. The most accurate classifier is the MLP classifier. Multilayer Perceptron (MLP), Naive Bayes, and Random Forest are the three techniques used. According to the experimental findings, MLP achieves maximum accuracy (98.63%).

A previous study has demonstrated a deep learning-based DDoS detection system to identify multi-vector attacks involving TCP, UDP, and ICMP in a custom SDN environment [14]. The suggested approach has 95.65% accuracy in classifying certain DDoS attacks. Compared to other studies, it classifies traffic as normal and strikes with 99.82% accuracy while generating very few false positives. However, in future research proposals, the NIDS system in this study has not been able to identify attacks at the application layer, especially on raw data.

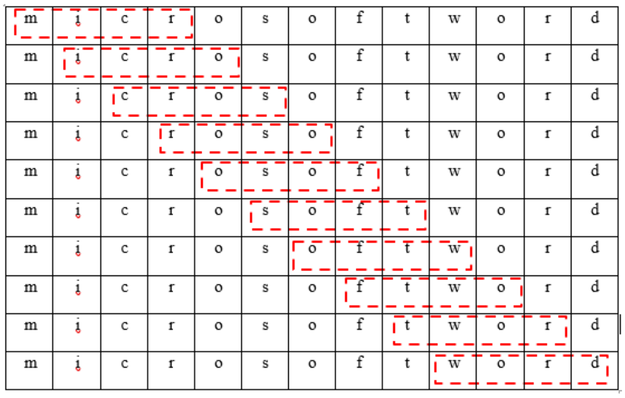
In order to reduce the false alarm rate, research by Khreich et al. [15] developed a new feature extraction technique that integrates frequency and temporal data from tracking system calls with a single class Support Vector Machine detector (OC-SVM). The approach is the feature extraction methodology. In order to train the OC-SVM detector, the proposed method first divides tracking system calls into n-grams of variable length and maps them to a fixed-size sparse feature vector. The system call dataset results show that our feature vector performs up to six grams better than the term vector model (using the most common weighting strategy) suggested in the related work. Its anomaly detection system, which used OC-SVM with a Gaussian kernel and was trained on our feature vectors, achieves greater detection accuracy rates than Markovian and n-gram-based models and more sophisticated anomaly detection methods (with alarm levels lower fake). While keeping the temporal link between events, the suggested feature extraction approach from event traces provides a fresh and well-liked data type for popular single-class machine learning techniques.

In contrast to Snort, the packets are matched against legal user access patterns to web pages rather than attack patterns. The test's findings show that the attack detection accuracy was 94.07% at a threshold value 0.85. This intrusion detection system is more resistant to zero-day attacks since it can identify different attacks without first describing existing attacks.

The research conducted by Sridharan [16] was continued from Oza [7], which states that web applications generate malicious HTTP requests that provide a platform to attack vulnerable machines to exploits. The network intrusion detection system must identify such malicious traffic based on traffic analysis. Previous research has shown that the N-Gram technique can be applied to detect HTTP attacks. This study analyzes the payload size by calculating Chi-square Distance, Pattern counting technique, and Ad-hoc N-Gram Technique. The results show that 2-Gram has an AUC value of 0.98 and an accuracy rate of detection of generic attacks, shellcode attacks, and CLET attack dataset of 98.16%, but the focus of the research is only on the size of the payload and 2-Gram to 3-Gram.

B. N-Gram Heuristic Techniques

Any problem-solving strategy that employs a realistic approach or numerous shortcuts to achieve answers that might not be ideal but are adequate given a constrained timeline or deadline is known as a heuristic or heuristic-based [17]. Heuristics-based approaches are adaptable and used for quick decisions, especially when working with complex data and finding the best solution is impossible or impracticable [18]. An N-Gram is a collection of N strings drawn from a collection of text or words. This series can be anything, depending on how to utilize it, such as letters, words, or sentences [19],[20]. A one-sized N-Gram is called a unigram, a two-sized one is called a bigram, and a three-sized one is called a trigram. Larger sizes are referred to as four-grams, five-grams, and so on. The working principle of the N-Gram can be seen in Figure 1.



m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d
m	i	c	r	o	s	o	f	t	w	o	r	d

Fig. 1 Principal diagram of 4-Gram segmentation [21]

From Figure 1, a 4-Gram string shift starting from the word "Microsoft word" by ignoring space, that a 4-Gram shift starting from "micr" and ending in "word", from all shifts is obtained the value of F, and each F may have the same pattern so that further analysis can be carried out [22].

The fields of information retrieval [23] and statistical natural language processing [24] have both used N-Grams in the past. This technique allows it to recover a set of symbols from the input stream using a sliding window of length n. Everywhere, a sequence of length n is taken into consideration.

The formal definition of a feature set S, which corresponds to all feasible sequences of length n:

$$S: = \{0, \ldots, 255\}\ n \qquad (1)$$

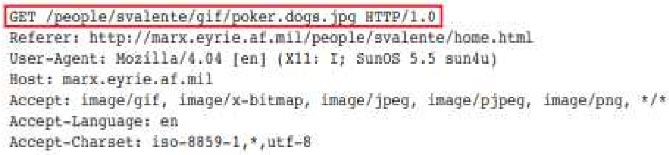
Chi-squared Distance is a technique for calculating the separation between two histograms of benign traffic that were seen with predicted frequency distributions and unknown payloads. Both X and Y are equal to [X1, X2,..., Xn]. First, the two histograms must be normalized, which requires that they add to one. X2 is determined between the n frequency distributions using this method. The training and testing phases make up this approach's two components.

$$D\ (X, Y) = \sum_{i=1}^{N} \frac{(Xi - Yi)^2}{Yi} \qquad (2)$$

Where:

n = Number of unique data on the histogram
Xi = Normality value of the value of xi (observed)
Yi = Normality value of the value of yi (Normal).

For example, in implementing the N-Gram technique, raw data is used in the HTTP protocol. The analyzed payload can be seen in Figure 2.



```
GET /people/svalente/gif/poker.dogs.jpg HTTP/1.0
Referer: http://marx.eyrie.af.mil/people/svalente/home.html
User-Agent: Mozilla/4.04 [en] (X11: I; SunOS 5.5 sun4u)
Host: marx.eyrie.af.mil
Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, image/png, */*
Accept-Language: en
Accept-Charset: iso-8859-1,*,utf-8
```

Fig. 2 An example of an HTTP packet payload

Since a normalcy model may be automatically created from the N-Grams present in a packet payload, the usage of N-Grams does not need the construction of necessary features by experts in the relevant subject [25]. Consider the artificial payload x = "ooddod" where the set of all possible symbols is restricted to "o" and "d" to show how the technique works. If n = 2, the sequences that can be extracted are “oo”, “od”, “dd”, “do”, and “od”, respectively.

In addition, Bazrafshan et al. [26] explain that N-Gram is a feature that can be used in feature selection, as shown in Figure 3.

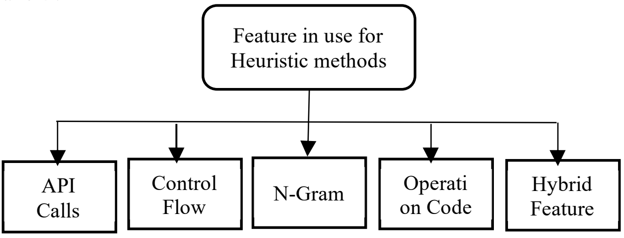


Fig. 3 Hybrid Methods Features [26]

Figure 3, the features used in the heuristics-based method consist of API Calls, CFG, N-Gram, Operation Code, and Hybrid features.

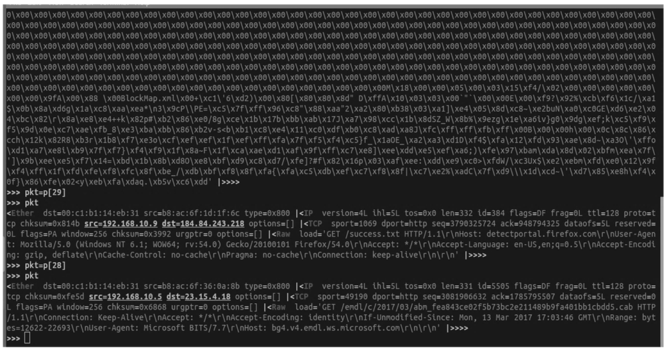
- API/System calls: Applications frequently communicate with the Operating System through application programming interface (API) calls. API call sequences are one of the most effective techniques to mimic the actions of malicious software [27]. As an illustration, API Calls to connect between access networks, like setWifiEnabled() and execHTTPRequest(), ZwOpenKeyEx
- Opcode: A machine language instruction subdivision known as a "opcode" designates the execution action. An organized set of assembly instructions makes up a program. An instruction is a pair comprising either a list of operands or an operational code. Opcode can be found in all programming languages, with examples in machine languages such as push, mov, call, StartupInfo [28].
- N-Gram: N-Gram is all substrings of a larger string of length N [29]. As an illustration, the string "ATTACK" can be divided into a number of 3-Grams, such as "ATT," "TTA," "TAC," "ACK," and so on. Several investigations have been conducted to identify unknown malware based on its binary code content over the last ten years. Based on the hex value of the HTTP protocol's content in DDoS attacks, the study will examine it [24],[30].
- Control flow graph: The Control Flow Graph (CFG), a graph that depicts the control flow of programs, has been extensively utilized in software analysis for many years [31], [32],[33]. CFG is a directed graph where every node corresponds to a program statement and every edge to the control flow between the statements. (i.e., what happens after what). Statements may be assignments, copy statements, branches, etc.
- Hybrid Features: Two key aspects affect how well machine learning classifiers perform: features and algorithms. Thus, Hybrid Feature combines feature selection algorithms and attack characteristic features to help machine learning models produce the best classification and predictions[34],[35],[36].

III. Results and Discussion

This section discusses the results of data packet construction using the N-Gram technique. There are two types of payloads extracted, normal Payload and DDoS Payload. The first stage is preparing data packets containing DDoS packets and normal packets originating from CIC-2017, MIB-2016, and H2NPayload, then extracting the hex payload using online tools and Python programming language.

A. Preparation Dataset Result

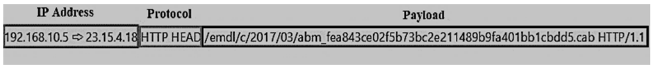
The identified payload is extracted from the raw data for additional analysis. The following results identify the raw data before converting it into hexadecimal form. This raw data is taken from the CIC-2017 dataset in the format of a PCAP file and then extracted using the scapy module in Figure 4.



```
0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00M\x18\x00\x00\x05\x00\x03\x15\xf4/\x02\x00\x00\x00\x00\x00\
x00\x00\x9fA\x00\x88 \x00BlockMap.xml\x00+\xc1\'6\xd2)\x00\x80[\x80\x80\x8d" D\xffA\x10\x03\x03\x00`"`\x00\x00E\x00\xf9?\x92%\xcb\xf6\x1c/\xa1
$\x0b\x8a\xd6g\x1a\xc8\xaa\xea*\n3\x9cP\\PE=\xc5\x7f\xff\x96\xc8"\x88\xaa"2\xa2\x80\xb38\x03\xa1]\xe4\x05\x8d\xc8~\xe2buN\xa0\xc0GE\xd6\xe2\x0
4\xbc\x82\r\x8a\xe8\xe4++k\x82p#\xb2\x86\xe0/8g\xce\x1b\x17b\xbb\xab\x17J\xa7\x98\xcc\x1b\x8dSZ_W\x8b%\x9ezg\x1e\xa6iv}g0\x9dg\xef;k\xc5\xf9\x
f5\x9d\x0e\xc7\xae\xfb_8\xe3\xba\xbb\x86\xb2v-s<b\xb1\xc8\xe4\x11\xc0\xdf\xb0\xc8\xad\xa8J\xfc\xff\xff\xfb\xff\x00B\x00\x00h\x00\x0c\x8c\x86\x
cch\x12k\x82R8\xb3r\x1b8\xf7\xe3o\xcf\xef\xef\x1f\xef\xff\xfa\x7f\xf5\xf4\xc5}f_\x1aOE_\xa2\xa3\xd1D\xf4$\xfa\x12\xfd\x93\xae\x8d~\xa3O\'\xffo
\xd1\xa7\xe8i\xb9\x7f\xf7}\xf4\xf9\x1f\x8a~F\x1f\xca\xae\xd1\xaf\x9f\xff\xc7\xe8]\xee\xdd\xe5\xef\xa6;)\xfe\x97\xbam\xda\x8d\x02\xbfm\xea\x7f\
']\x9b\xee\xe5\xf7\x14=\xbd\x1b\x8b\xd8O\xe8\xbf\xd9\xc8\xd7/\xfe]?#f\x82\x16p\x03\xaf\xee:\xdd\xe9\xc0>\xfdW/\xc3Ux$\xe2\xebm\xfd\xe0\x12\x9f
\xf4\xff\x1f\xfd\xfe\xf8\xfc\x8f\xbe_/\xdb\xbf\xf8\x8f\xfa{\xfa\xc5\xdb\xef\xc7\xf8\x8f|\xc7\xe2%\xadC\x7f\xd9\\\x1d\xcd~\'\xd7\x85\xe8h\xf4\x
0f}\x86\xfe\x02<y\xeb\xfa\xdaq.\xb5v\xc6\xdd' |>>>>
>>> pkt=p[29]
>>> pkt
<Ether  dst=00:c1:b1:14:eb:31 src=b8:ac:6f:1d:1f:6c type=0x800 |<IP  version=4L ihl=5L tos=0x0 len=332 id=384 flags=DF frag=0L ttl=128 proto=t
cp chksum=0x814b src=192.168.10.9 dst=184.84.243.218 options=[] |<TCP  sport=1069 dport=http seq=3790325724 ack=948794325 dataofs=5L reserved=
0L flags=PA window=256 chksum=0x3992 urgptr=0 options=[] |<Raw  load='GET /success.txt HTTP/1.1\r\nHost: detectportal.firefox.com\r\nUser-Agen
t: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0\r\nAccept: */*\r\nAccept-Language: en-US,en;q=0.5\r\nAccept-Encodi
ng: gzip, deflate\r\nCache-Control: no-cache\r\nPragma: no-cache\r\nConnection: keep-alive\r\n\r\n' |>>>>
>>> pkt=p[28]
>>> pkt
<Ether  dst=00:c1:b1:14:eb:31 src=b8:ac:6f:36:0a:8b type=0x800 |<IP  version=4L ihl=5L tos=0x0 len=331 id=5505 flags=DF frag=0L ttl=128 proto=
tcp chksum=0xfe5d src=192.168.10.5 dst=23.15.4.18 options=[] |<TCP  sport=49190 dport=http seq=3081906632 ack=1785795507 dataofs=5L reserved=0
L flags=PA window=256 chksum=0x6868 urgptr=0 options=[] |<Raw  load='GET /emdl/c/2017/03/abm_fea843ce02f5b73bc2e211489b9fa401bb1cbdd5.cab HTTP
/1.1\r\nConnection: Keep-Alive\r\nAccept: */*\r\nAccept-Encoding: identity\r\nIf-Unmodified-Since: Mon, 13 Mar 2017 17:03:46 GMT\r\nRange: byt
es=12622-22693\r\nUser-Agent: Microsoft BITS/7.7\r\nHost: bg4.v4.emdl.ws.microsoft.com\r\n\r\n' |>>>>
>>> 
```

Fig. 4 Sample Raw Data CIC-2017 Dataset

There are three steps to identify and analyze the raw data in a data packet: the first is to identify the IP Address, the second is the Network Protocol, and the third is to analyze the payload. All parts are converted from text to hexadecimal, as shown in Figure 4 below:



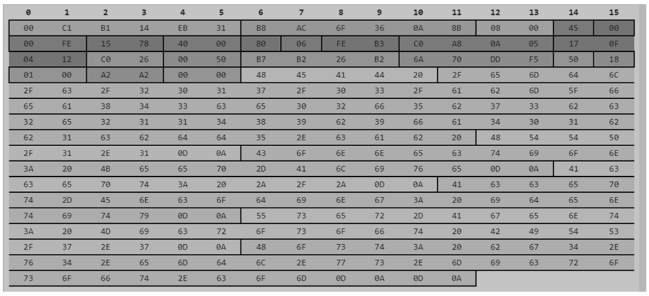
IP Address	Protocol	Payload
192.168.10.5 ⇨ 23.15.4.18	HTTP HEAD	/emdl/c/2017/03/abm_fea843ce02f5b73bc2e211489b9fa401bb1cbdd5.cab HTTP/1.1

Fig. 5 Payload Raw

Figures 4 and 5 are the results of the data collection process in this study. All data packets on each dataset will be analyzed in depth, focusing on the payload.

B. Payload Identifications

Next, identify and reconstruct the payload using the N-Gram technique described in the following section. The following results from extracting the payload from the CIC-2017 Dataset data packet using the Hex Packet Decoder tool (gasmi.net), which can be seen in Figure 6.



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00	C1	B1	14	EB	31	B8	AC	6F	36	0A	8B	08	00	45	00
00	FE	15	78	40	00	80	06	FE	B3	C0	A8	0A	05	17	0F
04	12	C0	26	00	50	B7	B2	26	B2	6A	70	DD	F5	50	18
01	00	A2	A2	00	00	48	45	41	44	20	2F	65	6D	64	6C
2F	63	2F	32	30	31	37	2F	30	33	2F	61	62	6D	5F	66
65	61	38	34	33	63	65	30	32	66	35	62	37	33	62	63
32	65	32	31	31	34	38	39	62	39	66	61	34	30	31	62
62	31	63	62	64	64	35	2E	63	61	62	20	48	54	54	50
2F	31	2E	31	0D	0A	43	6F	6E	6E	65	63	74	69	6F	6E
3A	20	4B	65	65	70	2D	41	6C	69	76	65	0D	0A	41	63
63	65	70	74	3A	20	2A	2F	2A	0D	0A	41	63	63	65	70
74	2D	45	6E	63	6F	64	69	6E	67	3A	20	69	64	65	6E
74	69	74	79	0D	0A	55	73	65	72	2D	41	67	65	6E	74
3A	20	4D	69	63	72	6F	73	6F	66	74	20	42	49	54	53
2F	37	2E	37	0D	0A	48	6F	73	74	3A	20	62	67	34	2E
76	34	2E	65	6D	64	6C	2E	77	73	2E	6D	69	63	72	6F
73	6F	66	74	2E	63	6F	6D	0D	0A	0D	0A				

Fig. 6 Payload hex

Figure 6, which is marked as the result of the identification of the payload of data packets, both normal data packets and data packets to be analyzed, which are separated by several fields; field descriptions for all data packets are as follows:

TABLE I
FIELD PACKET DESCRIPTION

Field	Hexadecimal
Ethernet	00c1b114eb31b8ac6f360a8b0800
IPV4	4500016c352a40008006e80dc0a80a0f0d6b0432
TCP	c12b0050a4f896d828237ace5018010249620000
HTTP **All**	00c1b114eb31b8ac6f360a8b0800450000fe157840 008006feb3c0a80a05170f0412c0260050b7b226b2 6a70ddf550180100a2a2000048454144202f656d64 6c2f632f323031372f30332f61626d5f66656138343 363653032663562373362633265323131343839 62 396661343031626231636264643 52e63616220485 454502f312e310d0a436f6e6e656374696f6e3a204b 6565702d416c6976650d0a4163636570743a202a2f 2a0d0a4163636570742d456e636f64696e673a2069 64656e746974790d0a557365722d4167656e743a20 4d6963726f736f667420424954532f372e370d0a48 6f73743a206267342e76342e656d646c2e77732e6d 6963726f736f66742e636f6d0d0a0d0a

Payload separation using the scapy module developed using Python programming. It is explained that the payload of the HTTP protocol can be separated from the data packet field.

C. Result in N-Gram Pattern Formation

To identify and analyze data payloads that include DDoS attack patterns and separate them into 2-Gram, 3-Gram, 4-Gram, 5-Gram, and 6-Gram by calculating the frequency of each payload packet string. After the conversion of all datasets, both the first, second, and third data sets, then determine the payload pattern using the N-Gram technique ranging from 2-Gram to 6-Gram as in the following payload example:

TABLE II
SLIDING STRING PAYLOAD

N-Gram	Sliding String Payload Observed	Sliding String Payload Normal
2	‘00’, ‘0c’, ‘c1’, ‘1b’, ‘b1’, ‘11’…	‘00’, ‘0c’, ‘c1’, ‘1b’, ‘b1’…
3	‘00c’, ‘0c1’, ‘c1b’, ‘1b1’…	‘00c’, ‘0c1’, ‘c1b’, ‘1b1’, ‘b11’…
4	‘00c1’, ‘0c1b’, ‘c1b1’, ‘1b11’…	‘00c1’, ‘0c1b’, ‘c1b1’, ‘1b11’…
5	‘00c1b’, ‘0c1b1’, ‘c1b11’, ‘1b114’…	‘00c1b’, ‘0c1b1’, ‘c1b11’, ‘1b114’…
6	‘00c1b1’, ‘0c1b11’, ‘c1b114’, ‘1b114e’…	‘00c1b1’, ‘0c1b11’, ‘c1b114’…

D. Result Calculation of Chi-square Distance

The Chi-Square Distance method will be used by applications to determine the Distance between regular packets and packets being analyzed from each other. Calculate pattern occurrence frequency, percentage, and Chi-Square distance starting from 2-Gram, 3-Gram, 4-Gram, 5-Gram, and 6-Gram after extracting the hex payload and creating payload string shifts. The steps for calculating CSD manually based on this formula are as follows:

$$D2 = \frac{(0.00186915887850467 - 0.00332225913621262)^2}{0.00332225913621262} + \frac{(0.00747663551401869 - 0.0166112956810631)^2}{0.0166112956810631} + \cdots \ldots + \frac{(0.016822429906542 - 0.0299003322259136)^2}{0.0299003322259136} = 0{,}327$$

The Pearson Chi-Square Test analysis was carried out as a threshold determination to determine the status of the payload observed, which was formed based on the following hypothesis:

$H0 : D2 \leq X2(\alpha, b - 1)$
$H1 : D2 > X2(\alpha, b - 1)$

H0 is interpreted as a DDoS packet, and H1 is not a DDoS attack or normal Payload. D2 is Chi-Square Distance between two payloads. X^2 is the value of the chi-square table with the significant value of a = 0.05, and the degree of freedom b-1, b is the number of unique patterns that appear in the reference packet (Normal/DDoS)

The chi-squared Distance between the analyzed packet and the reference packet will now be compared with the chi-squared table value of = 0.05 and the degree of freedom b-1. From the calculation of the chi-squared Distance, the value is 0.327. The value of X2 (0.05,146) is 176,293. Since the chi-squared distance value is less than the value of x2, the payload is a DDoS attack.

E. Experimentation Summary

This study uses three datasets, CIC-2017, MIB-2016, and H2N-Payload, to detect DDoS attacks. N-Gram technique analysis is used to determine whether a packet is malicious. The analysis is based on string patterns in each payload, ranging from 1-Gram to 6-Gram. The frequency of occurrence of patterns in each string is used to calculate the Chi-Square and Cosine Similarity value. Therefore, this value becomes a new feature in this study. Calculation of Chi-Square Distance and Cosine Similarity is performed on the three datasets. The result of the calculation will be the value for all features. The accuracy of each dataset is assessed using the SVM model once values are acquired for each feature. Each feature evaluates both datasets from CIC-2017, MIB-2016, and H2N-Payload. After each feature has been evaluated for correctness, a combined test of the two features is run. The combination of these two features is called a hybrid.

TABLE III
SUMMARY ACCURACY FOR THE CIC-2017 DATASET USING THE SVM ALGORITHM

Data set	Features	N-Gram using SVM (%)					
		1-G	2-G	3-G	4-G	5-G	6-G
CIC-2017	CSDPayload+N-Gram (21 features)	99.02	99.45	99.00	99.86	99.03	99.02
	CSPayload+N-Gram (21 features)	98.93	99.23	99.32	99.37	98.98	98.95
	CSDPayload+CSPayload+N-GRAM (22 features)	99.23	99.49	99.38	99.65	99.16	99.29
	(78 features)	**Without N-Gram** 84.16 **(%)** [23]					

Based on Table 3, it is explained that the 4-Gram feature is the best feature that can classify each payload. The accuracy rate for the CSDPayload+N-Gram feature is 99.86%, the CSPayload+N+Gram feature is 99.37%, and the CSDPayload+CSPayload feature is 99.65%. When compared with research conducted [23], it was concluded that there was an increase in the detection of DDoS attacks in the N-Gram technique compared to without using N-Gram, an increase of 15.70%, as well as with other features there was a significant increase in accuracy.

TABLE IV
SUMMARY ACCURACY FOR MIB-2016 DATASET USING SVM ALGORITHM

Data set	Features	N-Gram using SVM (%)					
		1-G	2-G	3-G	4-G	5-G	6-G
MIB-2016	CSDPayload+N-Gram (6 features)	99.00	99.92	99.94	99.98	99.84	99.66
	CSPayload+N-Gram Six features)	99.42	99.92	100.0	100.0	99.44	99.42
	CSDPayload+CSPayload+N-GRAM (7 features)	98.72	97.46	99.64	99.74	93.94	95.12
	(34 features)	Without N-Gram 97.90 % [37]					

Table 4 explains the accuracy rate for the CSDPayload+N-Gram feature is 99.98%, the CSPayload+N+Gram feature is 100%, and the CSDPayload+CSPayload feature is 99.74%. When compared with the research conducted, it was concluded that there was an increase in the detection of DDoS attacks in the N-Gram technique compared to without using N-Gram, an increase of 15.82%, as well as with other features there was a significant.

TABLE V
SUMMARY ACCURACY FOR H2NPAYLOAD DATASET USING SVM ALGORITHM

Data set	Features	N-Gram using SVM (%)					
		1-G	2-G	3-G	4-G	5-G	6-G
H2N-Payload	CSDPayload+N-Gram (7 features)	96.31	99.54	99.85	100.00	96.36	93.50
	CSPayload+N-Gram (7 features)	99.08	98.98	99.18	99.48	98.98	98.98
	CSDPayload+CSPayload+N-GRAM (8 features)	98.52	98.36	98.41	99.64	97.13	98.41

Table 5 explains that the 4-Gram feature is the best feature that can classify each payload. The accuracy rate for the CSDPayload+N-Gram feature is 100%, the CSPayload+N+Gram feature is 99.48%, and the CSDPayload+CSPayload feature is 99.64%. It is a new dataset produced in this study; therefore, there is no comparison of the level of accuracy in the same study that uses the N-Gram feature.

TABLE VI
THE OVERALL RESULTS IMPROVEMENT

Data set	Feature	Improvement					
		1-G	2-G	3-G	4-G	5-G	6-G
CIC-2017 [38], [13]	CSDPayload +N-Gram (21 features)	14.86	15.29	14.84	15.70	14.87	14.86
	CSPayload+ N-Gram (21 features)	14.77	15.07	15.16	15.21	14.82	14.79
	CSDPayload +CSPayload +N-GRAM (22 features)	15.07	15.33	15.22	15.49	15	15.13
MIB-2016[39]	CSDPayload+N-Gram (6 features)	1.10	2.02	2.04	2.08	-6.06	1.76
	CSPayload+ N-Gram 6 features)	1.52	2.02	2.10	2.10	1.54	1.52
	CSDPayload+CSPayload+N-GRAM (7 features)	0.82	-0.44	1.74	1.84	-3.96	-2.78

From Table 6, it can be explained that for the CIC-2017 dataset, when evaluating the performance of the N-Gram technique in detecting DDoS attacks, there was an increase in the level of accuracy for the CSDPayload+N-Gram feature on the 4-Gram subset reaching 15.70%, CSPayload +N-Gram 15.21 %, the Hybrid Payload+N-Gram feature is 15.49%. In comparison, for the MIB-2016 dataset, there is an increase in the accuracy rate for the CSDPayload+N-Gram feature in the 4-Gram subset reaching 2.08%, CSPayload +N-Gram 2.10% and Hybrid+N-Gram 1.84 %.

The results show that feature selection in detecting DDoS attacks uses the Hybrid N-Gram heuristic technique for the CIC-2017 dataset with the SVM algorithm on the CSDPayload+N-Gram feature with a 4-Gram accuracy rate of 99.86%, the MIB-2016 dataset with the algorithm SVM and features CSPayload+N-Gram with 100% accuracy rate for 4-Gram, payload H2N-Dataset with SVM Algorithm and CSDPayload+N-Gram feature with 100% accuracy rate for 4-Gram.

F. Compare the Algorithm and Result

The comparison algorithm in this study uses 2 algorithms and the same dataset. Therefore, a comparison is needed against other algorithms besides SVM to measure the performance of the proposed N-Gram technique. The comparison starts with the KNN algorithm and the Neural Network.

TABLE VII
H2N-PAYLOAD DATASET ACCURACY TEST CSDPAYLOAD+CSPAYLOAD+N-GRAM FOR KNN

No	N-Gram Size	Accuracy	Precession	Recall	ROC
1	1-Gram	91.97%	93.51%	96.67%	0.9710
2	2-Gram	89.00%	77.02%	71.57%	0.9540
3	3-Gram	73.15%	68.15%	57.65%	0.8110
4	4-Gram	88.74%	90.86%	92.77%	0.9630
5	5-Gram	82.91%	84.62%	90.75%	0.9100
6	6-Gram	90.69%	93.09%	94.49%	0.9700

Based on Table 7, it is explained that the experimental results on the H2N-Payload dataset with the KNN algorithm on the combined features of CSD + Cosine Similarity (Hybrid N-Gram), the highest level of accuracy obtained in detecting

DDoS attacks in this study was 91.97% for 1-Gram compared to another level of N-Gram accuracy.

TABLE VIII
H2N-PAYLOAD DATASET ACCURACY TEST CSDPAYLOAD+CSPAYLOAD+N-GRAM FOR NEURAL NETWORK

No	N-Gram Size	Accuracy	Precession	Recall	ROC
1	1-Gram	98.67%	99.37%	98.98%	0.9990
2	2-Gram	99.18%	99.35%	96.97%	1.0000
3	3-Gram	99.18%	99.08%	98.80%	1.0000
4	4-Gram	99.33%	99.46%	98.80%	1.0000
5	5-Gram	98.00%	98.17%	98.84%	0.9980
6	6-Gram	96.67%	99.03%	96.49%	0.9920

Table 8 explains the experimental results on the H2N-Payload dataset with the Neural Network algorithm on the combined features of CSDPayload+N-Gram+CSPayload+N-Gram (Hybrid N-Gram). This study's highest accuracy level in detecting DDoS attacks was 99.33% for 4-Gram compared to another level of N-Gram accuracy. Evaluation result using three Machine Learning algorithms, then the best algorithm for selecting features to improve the detection of DDoS attacks is the SVM for the 4-Gram algorithm, with an accuracy rate of up to 100%.

IV. CONCLUSION

As explained in the introduction, network security is important today because data protection in an organization is mandatory. It involves corporate confidentiality. One crucial aspect is data availability when accessed, but sometimes the data is unavailable due to server disturbances, one of which is a DDoS attack. Attacks known as denial-of-service (DoS) use the internet to attack vital Web services. By sending the target a substantial amount of unsolicited traffic to use up connection or bandwidth, this attack seeks to lower the quality of service a genuine service provides. DoS attacks are becoming more common, increasing the risk to servers and other devices connected to the internet. DDoS attacks have been happening for some time. Only a few defense systems could stop single-source attacks in the past, so better traceability prevents or repels attack sources. However, many systems today are vulnerable to attackers due to the rapid growth of the internet these days.

Therefore, this study proposes a DDoS attack detection technique using a hybrid N-Gram heuristic technique. The research stage shows that this technique can detect attacks by recognizing the percentage of two network class conditions (Normal and DDoS) for the CIC-2017 dataset with the SVM algorithm and the CSDPayload+N-Gram feature with a 4-Gram accuracy rate of 99.86%. MIB-2016 with SVM algorithm and PayloadCS+N-Gram features with 100.00% accuracy rate for 4-Gram, H2N-Payload dataset with SVM Algorithm and CSDPayload+ N-Gram feature with 100% accuracy for 4-Gram. In contrast, the KNN algorithm for 3-Gram has an accuracy rate of 99.44%, and the Neural Network Algorithm has an accuracy rate of 100% for 4-Gram. Thus, the best algorithm to detect DDoS is to use SVM. In contrast, the KNN and Neural Network algorithms are less consistent in classifying because the level of accuracy varies from 1-Gram to 6-Gram features.

ACKNOWLEDGMENT

Universiti Tun Hussein Onn Malaysia provided funding for this research (UTHM) and Book Hack Enterprise through SEPADAN RE-SIP (vot M071)

REFERENCES

[1] J. J. Kim, Y. S. Lee, J. Y. Moon, and J. M. Park, "Network payload and correlation analysis in bigdata environments," *Int. J. Grid Distrib. Comput.*, vol. 11, no. 3, pp. 109–124, 2018, doi: 10.14257/ijgdc.2018.11.3.10.

[2] M. Alkasassbeh, A. B. A. Hassanat, and G. Al-naymat, "Detecting Distributed Denial of Service Attacks Using Data Mining Techniques," vol. 7, no. 1, pp. 436–445, 2016.

[3] A. W. Muhammad and I. Riadi, "DDoS Attack Detection Using Neural Network with Fixed Moving Average Window Function," vol. 1, no. 3, pp. 115–122, 2017.

[4] A. Rahmatulloh, G. M. Ramadhan, I. Darmawan, N. Widiyasono, and D. Pramesti, "Identification of Mirai Botnet in IoT Environment through Denial-of-Service Attacks for Early Warning System," vol. 6, no. September, pp. 623–628, 2022.

[5] K. M. I. A. Fouda, "Payload Based Signature Generation for DDoS Attacks," University of Twente, 2017.

[6] K. M. Prasad, A. R. M. Reddy, and K. V. Rao, "DoS and DDoS Attacks: Defense, Detection and TracebackMechanisms -A Survey," vol. 14, no. 7, 2014.

[7] A. Oza, "HTTP Attack Detection using N-gram Analysis," 2013.

[8] M. Najafimehr, S. Zarifzadeh, and S. Mostafavi, *A hybrid machine learning approach for detecting unprecedented DDoS attacks*, no. 0123456789. Springer US, 2022.

[9] D. Ariu, R. Tronci, and G. Giacinto, "HMMPayl: An intrusion detection system based on Hidden Markov Models," *Comput. Secur.*, vol. 30, no. 4, pp. 221–241, 2011, doi: 10.1016/j.cose.2010.12.004.

[10] K. Kato and V. Klyuev, "An Intelligent DDoS Attack Detection System Using Packet Analysis and Support Vector Machine," *Int. J. Intell. Comput. Res.*, vol. 5, no. 3, pp. 464–471, 2014.

[11] J. David and C. Thomas, "DDoS attack detection using fast entropy approach on flow-based network traffic," *Procedia Comput. Sci.*, vol. 50, no. August, pp. 30–36, 2015, doi: 10.1016/j.procs.2015.04.007.

[12] N. Bindra and M. Sood, "Evaluating the impact of feature selection methods on the performance of the machine learning models in detecting DDoS attacks," *Rom. J. Inf. Sci. Technol.*, vol. 23, no. 3, pp. 250–261, 2020.

[13] K. Bouzoubaa, Y. Taher, and B. Nsiri, "Predicting DOS-DDOS Attacks: Review and Evaluation Study of Feature Selection Methods based on Wrapper Process," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 5, pp. 132–145, 2021, doi: 10.14569/IJACSA.2021.0120517.

[14] Q. Niyaz, W. Sun, and A. Y. Javaid, "A Deep Learning Based DDoS Detection System in Software-Defined Networking (SDN)," *ICST Trans. Secur. Saf.*, vol. 4, no. 12, p. 153515, 2017, doi: 10.4108/eai.28-12-2017.153515.

[15] W. Khreich, B. Khosravifar, A. Hamou-Lhadj, and C. Talhi, "An anomaly detection system based on variable N-gram features and one-class SVM," *Inf. Softw. Technol.*, vol. 91, pp. 186–197, 2017, doi: https://doi.org/10.1016/j.infsof.2017.07.009.

[16] S. Sridharan, "Defeating n-gram Scores for HTTP Attack Detection," *SJSU Sch. Work.*, vol. 6, no. San Jose State University, pp. 1–37, 2016, doi: 10.31979/etd.japx-z6eu.

[17] S. Khunkitti, A. Siritaratiwat, and S. Premrudeepreechacharn, "Multi-objective optimal power flow problems based on slime mould algorithm," *Sustain.*, vol. 13, no. 13, 2021, doi: 10.3390/su13137448.

[18] L. Csikar, "Decision making in the sciences : understanding heuristic use by students in problem solving," p. 124, 2018, doi: 10.25777/dzej-k872.

[19] R. R. Rumare and H. T. Ciptaningtyas, "HTTP Attack Detection Application Using N-Gram," *J. Tek. ITS*, vol. 6, no. 2, pp. 2–5, 2017, doi: 10.12962/j23373539.v6i2.24230.

[20] S. Bista and R. Chitrakar, "DDoS Attack Detection Using Heuristics Clustering Algorithm and Nave Bayes Classification," *J. Inf. Secur.*, vol. 09, no. 01, pp. 33–44, 2018, doi: 10.4236/jis.2018.91004.

[21] A. Maslan, K. M. Mohamad, and C. F. M. Foozy, "Enhancement detection distributed denial of service attacks using hybrid n-gram techniques," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 20, no. 1, pp. 61–69, 2022, doi: 10.12928/Telkomnika.v20i1.18103.

[22] H. Zhao, Z. Chang, G. Bao, and X. Zeng, “Malicious Domain Names Detection Algorithm Based on N -Gram,” vol. 2019, 2019.

[23] W. B. Cavnar and J. M. Trenkle, “N-Gram-Based Text Categorization N-Gram-Based Text Categorization,” *Proc. Third Annu. Symp. Doc. Anal. Inf. Retr.*, no. May, pp. 1–14, 2001.

[24] J. Daniel and J. H. Martin, “stanford n-gram_Speech and Language Processing,” 2021.

[25] F. Angiulli, L. Argento, and A. Furfaro, “Exploiting n-gram location for intrusion detection,” *CS.CR*, vol. 3, no. Cornell University, pp. 1–6, 2016, doi: 10.1109/ICTAI.2015.155.

[26] Z. Bazrafshan, H. Hashemi, S. M. H. Fard, and A. Hamzeh, “A survey on heuristic malware detection techniques,” *IKT 2013 - 2013 5th Conf. Inf. Knowl. Technol.*, no. May, pp. 113–120, 2013, doi: 10.1109/IKT.2013.6620049.

[27] W. Halim, “Deteksi Malware dengan Menggunakan API Calls,” *Paper*, p. 15, 2020.

[28] A. Shabtai, R. Moskovitch, C. Feher, S. Dolev, and Y. Elovici, “Detecting unknown malicious code by applying classification techniques on OpCode patterns,” *Secur. Inform.*, vol. 1, no. 1, p. 1, 2012, doi: 10.1186/2190-8532-1-1.

[29] T. Abou-Assaleh, N. Cercone, V. Keselj, and R. Sweidan, “N-gram-based detection of new malicious code,” in *Proc of the 28th Annual International Computer Software and Applications Conference, IEEE Computer Society*, 2004, vol. 2, pp. 41–42 vol.2, doi: 10.1109/CMPSAC.2004.1342667.

[30] I. Journal, O. F. Engineering, C. Of, M. Virus, and U. N. Gram, “International journal of engineering sciences & research technology classification of metamorphic virus using n gram analysis,” vol. 6, no. 2, pp. 364–370, 2017.

[31] L. Tan, “The worst-case execution time tool challenge 2006,” *STTT*, vol. 11, pp. 133–152, 2009, doi: 10.1109/ISoLA.2006.72.

[32] T. McCabe, “A Complexity Measure,” *IEEE Trans. Softw. Eng.*, vol. SE-2, pp. 308–320, 1976.

[33] P. Jalote, *An Integrated Approach to Software Engineering*. 1997.

[34] M. A. H. Azmi, C. F. M. Foozy, K. A. M. Sukri, N. A. Abdullah, I. R. A. Hamid, and H. Amnur, “Feature Selection Approach to Detect DDoS Attack Using Machine Learning Algorithms,” *Int. J. Informatics Vis.*, vol. 5, no. 4, pp. 395–401, 2021, doi: 10.30630/JOIV.5.4.734.

[35] A. Martín, R. Lara-Cabrera, and D. Camacho, “Android malware detection through hybrid features fusion and ensemble classifiers: The AndroPyTool framework and the OmniDroid dataset,” *Inf. Fusion*, vol. 52, no. December, pp. 128–142, 2019, doi: 10.1016/j.inffus.2018.12.006.

[36] S. Almutairi, S. Mahfoudh, S. Almutairi, and J. S. Alowibdi, “Hybrid Botnet Detection Based on Host and Network Analysis,” *J. Comput. Networks Commun.*, vol. 2020, no. Hindawi, pp. 1–16, 2020, doi: 10.1155/2020/9024726.

[37] C. Ma, X. Du, and L. Cao, “Analysis of multi-Types of flow features based on hybrid neural network for improving network anomaly detection,” *IEEE Access*, vol. 7, pp. 148363–148380, 2019, doi: 10.1109/ACCESS.2019.2946708.

[38] Z. Chiba, N. Abghour, K. Moussaid, A. El, and M. Rida, “Intelligent and Improved Self-Adaptive Anomaly based Intrusion Detection System for Networks,” vol. 11, no. 2, pp. 312–330, 2019.

[39] T. Mahjabin, Y. Xiao, G. Sun, and W. Jiang, “A survey of distributed denial-of-service attack, prevention, and mitigation techniques,” *Int. J. Distrib. Sens. Networks*, vol. 13, no. 12, 2017, doi: 10.1177/1550147717741463.