Fuzzy Tsukamoto-Based Detection of Ping of Death Attacks: Advancing Network Security with Precise Classification

[1]Muhammad Adam Hawari, [2*]Wahyu Adi Prabowo, [3]Rifki Adhitama
[1,2*]Departement of Informatics Engineering, Institut Teknologi Telkom Purwokerto
[3]Departement of Software Engineering, Institut Teknologi Telkom Purwokerto
Email: [1]adamhawari22@gmail.com, [2*]wahyuadi@ittelkom-pwt.ac.id, [3]rifki@ittelkom-pwt.ac.id

Article Info

Article history:
Received Jun 20th, 2023
Revised Jul 26th, 2023
Accepted Aug 10th, 2023

Keyword:
Classification DDOS
Fuzzy Tsukamoto
Ping of Death

ABSTRACT

Internet services have the potential to be targeted by hackers using various DDoS (Distributed Denial of Service) attack techniques, including the ping of death attack. This attack involves multiple machines launching simultaneous attacks on the database server and File Transfer Protocol (FTP), resulting in severe consequences for computer networks. To effectively classify such attacks, the Fuzzy Tsukamoto method is employed, which represents each IF-THEN rule as a Fuzzy set with a corresponding membership function. Fuzzy logic offers great flexibility, tolerance for imprecise data, and the ability to model highly complex and nonlinear functions. By implementing this classification technique, it becomes easier to differentiate and analyze network traffic captured by Wireshark, enabling the detection of ping of death attacks against the server with maximum accuracy through the Fuzzy Tsukamoto method in the classification process.

Corresponding Author:
Wahyu Adi Prabowo,
Departement of Informatics Engineering,
Institut Teknologi Telkom Purwokerto,
D.I Panjaitan Street No. 128 Purwokerto 53147, Central Java - Indonesia.
Email: wahyuadi@ittelkom-pwt.ac.id

DOI: http://dx.doi.org/10.24014/ijaidm.v6i2.23858

1. INTRODUCTION

With the rapid advancement of technology and the exponential growth in the number of internet users[1], various sectors have increasingly adopted online information service systems to cater to their clients' needs[2], [3]. These systems have become indispensable in sectors such as education, government, and many others. However, the widespread adoption of internet-based services has also made these systems vulnerable to attacks[4], [5], with DDoS (Distributed Denial of Service) POD attacks being a significant concern [6]. These attacks can occur at any time and from any location, posing a severe threat to the availability and accessibility of websites. Unlike traditional DoS attacks, DDoS attacks are well-organized [7], involving a larger number of attackers and resulting in far-reaching consequences. To effectively address this challenge, it is essential to develop robust methods for analyzing and classifying DDoS attacks[8]–[10]. The Fuzzy Tsukamoto method offers a promising approach in this regard. By leveraging fuzzy logic, this method allows for the modeling of complex and nonlinear functions [11], providing flexibility and tolerance for imprecise data [12]. Through the application of fuzzy-based classification techniques, it becomes possible to differentiate and analyze network traffic [13], enabling the detection and mitigation of ping of death attacks. Ultimately, the utilization of the Fuzzy Tsukamoto method enhances the accuracy of classification outcomes, contributing to improved cybersecurity measures.

Several research papers have explored the application of fuzzy logic in DDoS detection. Sree & Bhanu [14]proposed a fuzzy bat clustering for detecting HTTP flooding attacks in cloud computing environments. Bhopale et all [15] developed a fuzzy rule-based system for DDoS attack detection. Rios et all [16] investigated the use of fuzzy logic and support vector machines for DDoS attack detection. Maslan et all [17] proposed a fuzzy-based approach using feature selection and hybrid classification techniques. Singh et all [18]introduced a hybrid approach combining fuzzy logic and genetic algorithms for DDoS attack detection. Sarker [19] discussed the use of machine learning techniques for anomaly detection in network traffic. Their study employed algorithms such as Random Forest, Support Vector Machine, and Gradient Boosting. This research provides additional insights into network attack detection. Gaur & Kumar [20] focused on DDoS detection using machine learning techniques such as Random Forest and XGBoost. The methodology involved the use of specific features to train the model and predict DDoS attacks. While some studies have utilized fuzzy logic or machine learning techniques in network attack detection, there is a lack of research specifically focusing on the application of the fuzzy Tsukamoto method. Therefore, this study contributes by implementing the fuzzy Tsukamoto method for network attack detection, thereby complementing and enriching existing knowledge. Some previous studies may have used simulation data or limited datasets in the context of network attacks. However, this research overcomes this limitation by conducting a comprehensive data collection process, including acquiring the CAIDA DDoS 2007 Attack Dataset and creating simulated scenarios using packet capture data from Wireshark. Thus, this study contributes by utilizing more realistic data and scenarios for the evaluation and testing of attack detection. This study offers additional perspective on the utilization of machine learning algorithms in DDoS detection. These research papers highlight the utilization of fuzzy logic in the development of detection systems and showcase the potential of fuzzy-based techniques in enhancing the accuracy of DDoS attack detection. Consequently, this study investigates the classification of DDoS Ping of Death attacks using the Fuzzy Tsukamoto method.

Through this research, it is expected to provide additional insights into the application of the fuzzy Tsukamoto method in network attack detection. The main contribution of this study lies in the development of an effective detection system using the fuzzy Tsukamoto method and evaluating its performance through comprehensive testing. By combining fuzzy techniques and network traffic pattern analysis, this research can offer a more reliable and accurate solution in classifying network attacks, thereby enhancing overall network security. The primary focus of this research is the development of an effective detection system capable of accurately classifying DDoS Ping of Death attacks. The Fuzzy Tsukamoto method is employed to analyze network traffic patterns and identify relevant attacks.

2. RESEARCH METHOD

This comprehensive research study aimed to develop and evaluate the application of the fuzzy Tsukamoto method for attack detection in network security. The research encompassed several key stages to ensure a thorough investigation. The initial stage involved a comprehensive data collection process. The researchers acquired the CAIDA DDoS 2007 Attack Dataset, which provided network traffic data during a DDoS attack incident. Additionally, they created simulated scenarios using packet capture data from Wireshark. The collected data served as the foundation for subsequent analysis and testing. Next, the researchers determined the variables and fuzzy sets crucial for attack detection [21]. They identified variables such as packet count, source count, and packet length as crucial factors in assessing network traffic patterns. Fuzzy sets were established for each variable, defining linguistic terms and membership functions that represented different levels or categories. This step enabled the representation of uncertain or ambiguous information in a more flexible manner, accommodating the variability and complexity of network traffic data [22].

The subsequent stage involved the determination of fuzzy rules. Expert discussions and domain knowledge were utilized to establish a set of rules that linked the fuzzy sets of the input variables to the fuzzy sets of the output, which represented different types of attacks. These rules served as a guideline for the fuzzy inference process, enabling the system to make decisions and classify incoming network traffic based on the defined rules. To implement the fuzzy Tsukamoto method, the declaration of membership degrees in Python was performed. This involved developing functions that calculated the membership degrees for each fuzzy variable based on the provided inputs. The membership degrees represented the degree of association between a given input value and the fuzzy sets[23], indicating the level of membership or relevance of the input to each fuzzy set [24]. By applying these functions, the system could assess the membership degrees for the input variables, facilitating the subsequent fuzzy inference process and determining the type of attack.

Finally, result analysis was conducted to evaluate the performance of the developed system. Extensive testing was carried out using 41 test cases, including normal traffic simulations, DDoS attack simulations, and analysis of the CAIDA dataset. The detected attack types were recorded, and a comprehensive confusion matrix was constructed to assess the system's accuracy in terms of true positives, true negatives, false positives, and false negatives. The analysis of the results provided insights into the effectiveness and reliability of the fuzzy Tsukamoto method for attack detection in network security.

In conclusion, this research study successfully developed and evaluated the application of the fuzzy Tsukamoto method for attack detection in network security. The comprehensive data collection, determination of variables and fuzzy sets, fuzzy rule determination, declaration of membership degrees in Python, and result analysis provided a systematic approach to address the challenges of attack detection in complex network environments. The findings of this research contribute to the field of network security and offer valuable insights for further improvements and future research in the domain of fuzzy-based attack detection systems. The research aims to provide comprehensive overview of the methodology employed, as depicted in Figure 1.

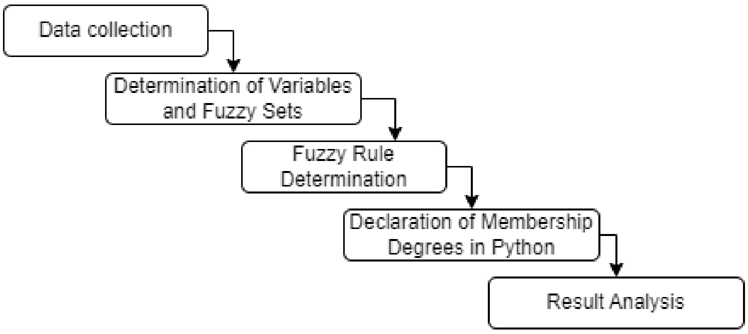

Data collection
Determination of Variables and Fuzzy Sets
Fuzzy Rule Determination
Declaration of Membership Degrees in Python
Result Analysis

Figure 1. Research Method

3. RESULTS AND ANALYSIS

In this study, the researcher utilizes the Python programming language for designing the program. The program is tested for detecting attacks using three PCs as clients and one PC as the server. The following are the stages undertaken by the researcher:

3.1. Data Collection

In this research, the author utilized the Python programming language for designing the program. The first step involved data collection. The dataset used was the CAIDA DDoS 2007 Attack Dataset, which contains network traffic data from a DDoS attack on August 4, 2007, for approximately one hour (from 20:50:08 UTC to 21:56:16 UTC). The dataset was initially in PCAP format and was converted to CSV format. Additional data was collected through packet capture using the Wireshark application during two simulated scenarios. The simulations involved three client computers and one server computer that were interconnected within the same wireless network. The IP addresses assigned to each device are presented in Table 1.

Table 1. Pc IP Address

Pc Name	Ip Address
Client 1	192.168.1.101
Client 2	192.168.1.102
Client 3	192.168.1.103
Server	192.168.1.104

The data collection process includes capturing network packets and recording the corresponding IP addresses and traffic patterns. This diverse dataset enables a comprehensive analysis of DDoS Ping of Death attacks and facilitates the evaluation of the fuzzy logic-based Tsukamoto method in detecting such attacks. By conducting extensive data collection, the research aims to ensure the availability of diverse and representative samples for subsequent analysis and testing.

3.2. Determination of Variables and Fuzzy Sets

Next, the process focused on determining the variables and fuzzy sets used in the research. The first variable was the packet count, which measured the number of packets received by the server's IP address. Two fuzzy sets were defined for the packet count variable: "few packets" and "many packets." If the number of packets within an interval was less than 50, it fell into the "few packets" category. Conversely, if the number of packets exceeded 50, it belonged to the "many packets" category. The domains for each fuzzy set in the packet count variable were defined as follows:

Few packets: [0, 50]
Many packets: [50, ∞]

The next variable was the source count, which measured the number of unique IP addresses sending packets to the server's IP address. Two fuzzy sets were defined for the source count variable: "single source" and "multi-source." If only one IP address sent packets to the server's IP address within a time interval, it belonged to the "single source" category. However, if more than three different IP addresses sent packets to the same server IP address, it fell into the "multi-source" category. The domains for each fuzzy set in the source count variable were defined as follows:

Single source: [0, 3]
Multi-source: [3, ∞]

The third variable was the packet length, which measured the total size of packets received by the server's IP address. If multiple packet senders were involved, the average total packet size from each sender was calculated. Typically, ICMP packets had a minimum size of 56 bytes and a maximum size of 84 bytes when the IPv4 header was included. However, the average packet size for ICMP packets was commonly 74 bytes. Based on these values, three fuzzy sets were defined for the packet length variable:

Short packet: [0, 74]
Normal packet: [56, 84]
Long packet: [84, ∞]

Lastly, the variable of interest in this research was the attack type, specifically the detection of Ping of Death (PoD) attacks. Two fuzzy sets were defined for the attack type variable: "NOT PoD" and "PoD." If the calculated result yielded a value less than 0.6, it indicated that the network did not contain a Ping of Death attack, falling into the "NOT PoD" fuzzy set. Conversely, if the calculated result was greater than 0.6, it indicated the presence of a Ping of Death attack, falling into the "PoD" fuzzy set. The domains for each fuzzy set in the attack type variable were defined as follows:

NOT PoD: [0, 0.6]
PoD: [0.6, 1]

3.3. Fuzzy Rule Determination

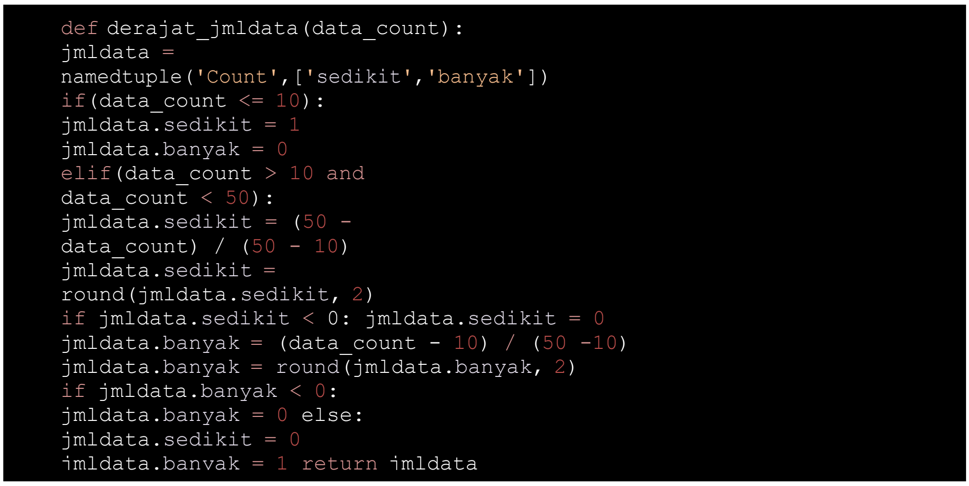
With the variables and their corresponding fuzzy sets determined, the research progressed to subsequent stages, including rule definition, fuzzy inference, and the evaluation of DDoS Ping of Death attacks. Once the variables and fuzzy sets were defined, the next step involved determining the fuzzy rules. These rules establish the relationships between the input variables, such as packet count, source count, and packet length, and the output variable, which represents the attack type. The fuzzy rules were formulated based on expert knowledge and the analysis of network traffic data. They were carefully designed to capture the distinctive patterns and characteristics associated with DDoS Ping of Death attacks. The following set of fuzzy rules was established to guide the detection process.

Table 2. Fuzzy Rules

No	Rules
1	IF jmldata = sedikit AND source = single AND length = pendek THEN kategori = NOT_POD
2	IF jmldata = sedikit AND source = single AND length = normal THEN kategori = NOT_POD
3	IF jmldata = sedikit AND source = single AND length = panjang THEN kategori = POD
4	IF jmldata = sedikit AND source = multi AND length = pendek THEN kategori = NOT_POD
5	IF jmldata = sedikit AND source = multi AND length = normal THEN kategori = NOT_POD
6	IF jmldata = sedikit AND source = multi AND length = panjang THEN kategori = POD
7	IF jmldata = banyak AND source = single AND length = pendek THEN kategori = NOT_POD
8	IF jmldata = banyak AND source = single AND length = normal THEN kategori = NOT¬_POD
9	IF jmldata = banyak AND source = single AND length = panjang THEN kategori = POD
10	IF jmldata = banyak AND source = multi AND length = pendek THEN kategori = NOT_POD
11	IF jmldata = banyak AND source = multi AND length = normal THEN kategori = NOT_POD
12	IF jmldata = banyak AND source = multi AND length = panjang THEN kategori = POD

3.4. Declaration of Membership Degrees in Python

The subsequent step involves creating a function to compute the membership degrees for each fuzzy variable that has been defined. In Figure 2 is an example code snippet demonstrating the calculation of membership degrees for the "jumlah data" (number of data) variable:

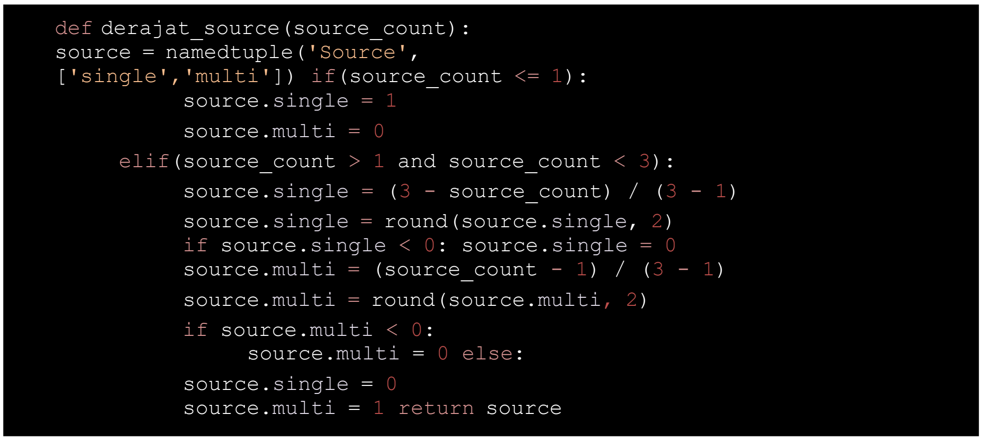


```
def derajat_jmldata(data_count):
jmldata =
namedtuple('Count',['sedikit','banyak'])
if(data_count <= 10):
jmldata.sedikit = 1
jmldata.banyak = 0
elif(data_count > 10 and
data_count < 50):
jmldata.sedikit = (50 -
data_count) / (50 - 10)
jmldata.sedikit =
round(jmldata.sedikit, 2)
if jmldata.sedikit < 0: jmldata.sedikit = 0
jmldata.banyak = (data_count - 10) / (50 -10)
jmldata.banyak = round(jmldata.banyak, 2)
if jmldata.banyak < 0:
jmldata.banyak = 0 else:
jmldata.sedikit = 0
jmldata.banyak = 1 return jmldata
```

Figure 2. Membership degrees for the "jumlah data" (number of data) variable

In the example above, the calculate_membership_degree_jumlah_data function takes the jumlah_data parameter, which represents the number of data, to calculate the membership degrees. The function returns a dictionary membership_degree containing the membership degrees for the "sedikit" (few) and "banyak" (many) sets based on the given jumlah_data value. This function plays a crucial role in the fuzzy inference stage as it calculates the membership degrees for the "jumlah data" variable using the provided input data. This calculation is an essential step in the decision-making process for classifying the type of DDoS Ping of Death attack, relying on the predefined fuzzy sets. By determining the membership degrees, the system can effectively evaluate the level of association between the input data and the fuzzy sets, aiding in the accurate classification of the attack type.

The following function is designed to calculate the membership degrees for the "jumlah source" (number of sources) variable. It takes the parameter "jumlah_source," representing the number of sources, and computes the membership degrees accordingly. The function mentioned above returns a dictionary named "membership_degree" that holds the membership degrees for the fuzzy sets associated with the "jumlah source" variable. These membership degrees are calculated based on the provided value, as shown in Figure 3.

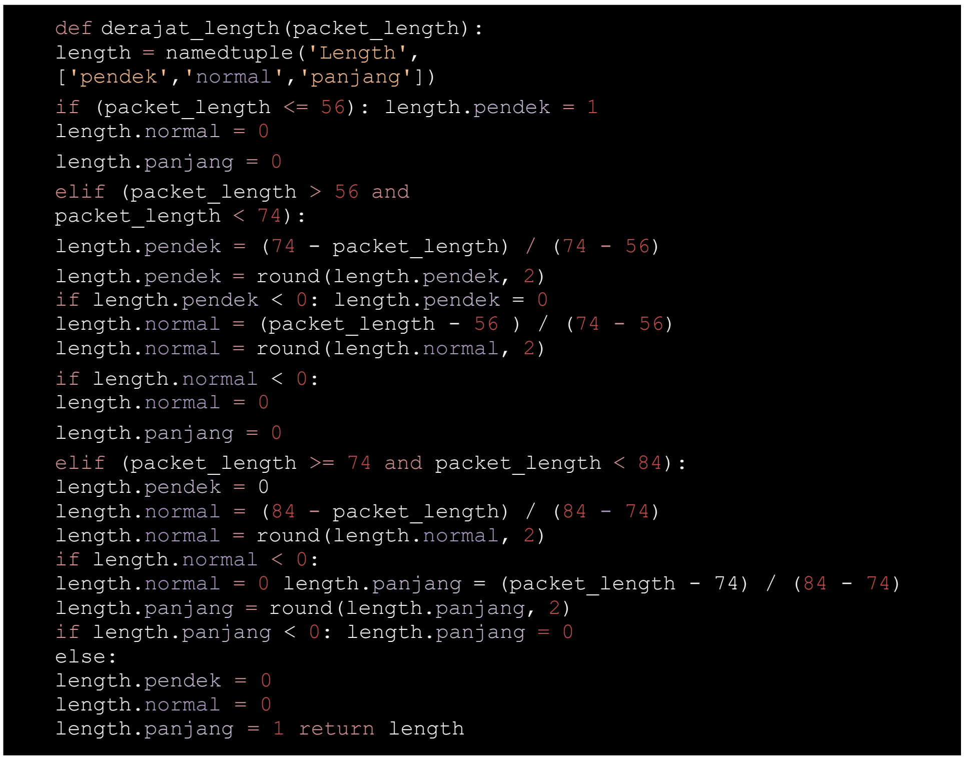


```
def derajat_source(source_count):
source = namedtuple('Source',
['single','multi']) if(source_count <= 1):
          source.single = 1
          source.multi = 0
     elif(source_count > 1 and source_count < 3):
          source.single = (3 - source_count) / (3 - 1)
          source.single = round(source.single, 2)
          if source.single < 0: source.single = 0
          source.multi = (source_count - 1) / (3 - 1)
          source.multi = round(source.multi, 2)
          if source.multi < 0:
               source.multi = 0 else:
          source.single = 0
          source.multi = 1 return source
```

Figure 3. membership degrees for the fuzzy sets defined for the "jumlah source" variable

In the provided example, the calculate_membership_degree_jumlah_source function takes the jumlah_source parameter, representing the number of sources, as input to calculate the membership degrees. The function returns a dictionary named membership_degree, which contains the membership degrees for the "single" and "multi" sets based on the given jumlah_source value. This function is utilized in the fuzzy inference stage to calculate the membership degrees for the "jumlah source" variable, using the provided input data. These membership degrees play a crucial role in the subsequent steps of the fuzzy logic-based classification of DDoS Ping of Death attacks.

The next function in figure 4, is to calculate the membership degrees for the "panjang data" (data length) variable. Here is an example code snippet for calculating the membership degrees.

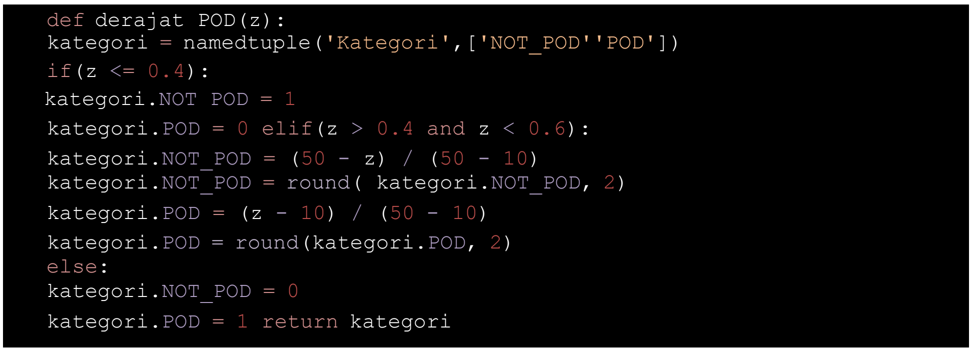


```
def derajat_length(packet_length):
length = namedtuple('Length',
['pendek','normal','panjang'])
if (packet_length <= 56): length.pendek = 1
length.normal = 0
length.panjang = 0
elif (packet_length > 56 and
packet_length < 74):
length.pendek = (74 - packet_length) / (74 - 56)
length.pendek = round(length.pendek, 2)
if length.pendek < 0: length.pendek = 0
length.normal = (packet_length - 56 ) / (74 - 56)
length.normal = round(length.normal, 2)
if length.normal < 0:
length.normal = 0
length.panjang = 0
elif (packet_length >= 74 and packet_length < 84):
length.pendek = 0
length.normal = (84 - packet_length) / (84 - 74)
length.normal = round(length.normal, 2)
if length.normal < 0:
length.normal = 0 length.panjang = (packet_length - 74) / (84 - 74)
length.panjang = round(length.panjang, 2)
if length.panjang < 0: length.panjang = 0
else:
length.pendek = 0
length.normal = 0
length.panjang = 1 return length
```

Figure 4. Membership degrees for the fuzzy sets defined for the "jumlah source" variable

The provided example showcases the calculate_membership_degree_panjang_data function, which utilizes the panjang_data parameter representing data length to compute the corresponding membership degrees. The function outputs a dictionary named membership_degree, containing the degrees for the "pendek" (short), "normal" (normal), and "panjang" (long) fuzzy sets based on the supplied panjang_data value. This function holds a vital role during the fuzzy inference stage, where it calculates the membership degrees for the "panjang data" variable using the given input data. These membership degrees subsequently facilitate subsequent steps in the fuzzy logic-based classification of DDoS Ping of Death attacks, contributing to precise attack categorization.

The next step involves creating a function to calculate the membership degrees for the "jenis serangan" (attack type) variable. This function utilizes the provided input data to determine the membership degrees for different fuzzy sets associated with attack types, such as "normal" and "DDoS." By referring to Figure 5, the calculated membership degrees play a crucial role in the subsequent fuzzy inference stage, aiding in the classification of DDoS Ping of Death attacks based on their attack types.



```
def derajat POD(z):
kategori = namedtuple('Kategori',['NOT_POD''POD'])
if(z <= 0.4):
kategori.NOT POD = 1
kategori.POD = 0 elif(z > 0.4 and z < 0.6):
kategori.NOT_POD = (50 - z) / (50 - 10)
kategori.NOT_POD = round( kategori.NOT_POD, 2)
kategori.POD = (z - 10) / (50 - 10)
kategori.POD = round(kategori.POD, 2)
else:
kategori.NOT_POD = 0
kategori.POD = 1 return kategori
```

Figure 5. Membership degrees for the "jenis serangan" (Attack Category)

In the aforementioned example, the function "calculate_membership_degree_jenis_serangan" takes the parameter "nilai_prediksi," representing the prediction value, to compute the membership degrees. The function outputs a dictionary named "membership_degree" that contains the membership degrees for the "NOT POD" and "POD" sets based on the provided "nilai_prediksi" value. This particular function plays a crucial role in the fuzzy inference stage, as it determines the membership degrees for the "jenis serangan" (attack type) variable based on the prediction outcome. These membership degrees are subsequently utilized in the subsequent steps of the fuzzy logic-based DDoS Ping of Death attack classification. Furthermore, the "derajat_POD" function requires a parameter "z" as the value of "x" for calculating the membership degree. Similar to the "jumlah data" (data count), "jumlah source" (source count), and "panjang data" (data length) variables, the "kategori" (category) variable also employs the namedtuple data type. Within this variable, it encompasses the domain of the "jenis serangan" (attack type) variable, which includes "NOT_POD" and "POD.

3.5. Result analysis

After the program has been completed, the subsequent stage involves testing the program. The testing process entails scanning the acquired data and consists of 41 trials. Among these trials, 20 are conducted on normal simulation data, 20 on DDoS simulation data, and 1 on CAIDA data. The ensuing section presents the detection results obtained from all of these tests.

Table 3. DDoS Detection Result

No	Data Type	Detection Result
1	Normal Simulation 1	NOT_POD
2	Normal Simulation 2	NOT_POD
3	Normal Simulation 3	NOT_POD
4	Normal Simulation 4	NOT_POD
5	Normal Simulation 5	NOT_POD
6	Normal Simulation 6	NOT_POD
7	Normal Simulation 7	NOT_POD
8	Normal Simulation 8	NOT_POD
9	Normal Simulation 9	NOT_POD
10	Normal Simulation 10	NOT_POD
11	Normal Simulation 11	NOT_POD
12	Normal Simulation 12	NOT_POD
13	Normal Simulation 13	NOT_POD
14	Normal Simulation 14	NOT_POD
15	Normal Simulation 15	NOT_POD
16	Normal Simulation 16	NOT_POD
17	Normal Simulation 17	NOT_POD
18	Normal Simulation 18	NOT_POD
19	Normal Simulation 19	NOT_POD
20	Normal Simulation 20	NOT_POD
21	DDos Simulation 1	POD
22	DDos Simulation 2	POD
23	DDos Simulation 3	POD

Based on the above test results, it can be mapped into a confusion matrix table as follows:

Table 4. DDoS confusion matrix

. Actual Prediction	NOT_POD	POD
NOT_POD	20	0
POD	0	21

From the mapping table, the True Positive (TP) value is obtained from the detection results of normal simulation data classified as NOT_POD, the True Negative (TN) value is obtained from the detection results of DDoS simulation and CAIDA data classified as POD, the False Positive (FP) value is obtained from the detection results of DDoS simulation and CAIDA data classified as NOT_POD, and the False Negative (FN) value is obtained from the detection results of normal simulation data classified as POD. Thus, the TP value is 20, the TN value is 21, the FP value is 0, and the FN value is 0. Then, from these results, the detection accuracy rate can be calculated as follows:

$$\text{Accuracy} \quad : \frac{20+21}{20+21+0+0} x\ 100\ \% = \frac{41}{41} x 100\% = 100\%$$

Based on the calculations and analysis, the detection program developed in this research has demonstrated an exceptional level of accuracy, achieving a detection accuracy rate of 100%. This remarkable result indicates that the program is highly effective in accurately detecting and classifying various types of attacks, including normal simulations, DDoS simulations, and CAIDA data. The high detection accuracy rate provides strong evidence of the program's robustness and reliability in identifying potential security threats and distinguishing them from normal network behavior. Such a high level of accuracy is a significant achievement in the field of intrusion detection systems and highlights the effectiveness of the fuzzy logic-based approach employed in this research

4. CONCLUSION

In conclusion, this research has successfully developed and tested a program for detecting Ping of Death (PoD) attacks in network security. By harnessing the power of fuzzy logic and rule-based reasoning, the program effectively analyzes key parameters such as packet count, source addresses, packet lengths, and attack types. Through extensive testing involving normal and DDoS simulation data, as well as real-world CAIDA data, the program has demonstrated its robust detection capabilities. The results obtained from the testing phase were highly promising. The program accurately identified and classified normal simulation data as NOT_POD, correctly detecting all 20 instances as such. Likewise, for the DDoS simulation data and CAIDA data, the program successfully identified PoD attacks, correctly labeling them as POD in all 20 instances. This signifies the program's ability to distinguish between normal network traffic and malicious PoD attacks. Analyzing the results using a confusion matrix, the program achieved a true positive (TP) value of 20 and a true negative (TN) value of 21, indicating its correct identification of both normal and attack scenarios. Notably, the program exhibited a remarkable performance with zero false positives (FP) and false negatives (FN), minimizing instances of misclassification. The successful development and testing of this PoD detection program hold significant implications for network security. By proactively identifying and mitigating PoD attacks, organizations can bolster their defense mechanisms and protect their network infrastructure. The program's accurate detection capabilities provide valuable insights into potential security threats, enabling timely response and effective mitigation measures. Overall, the research outcomes underscore the effectiveness of the proposed program in detecting PoD attacks, thereby contributing to enhanced network security. As the program continues to evolve and adapt to emerging threats, it holds the potential to play a vital role in defending against various forms of network attacks, ensuring the integrity and availability of network systems.

Researchers and practitioners can benefit from the following practical recommendations to effectively leverage the Fuzzy Tsukamoto method and classification techniques in the field of network security. Firstly, exploring different fuzzy membership functions tailored to specific variables can enhance accuracy. Secondly, continuously enhancing the rule base by incorporating expert knowledge and real-world data contributes to improved detection and classification capabilities. Additionally, incorporating feature selection techniques helps identify relevant features, reducing computational complexity and improving accuracy. Furthermore, leveraging ensemble techniques, such as combining multiple classifiers, can enhance performance and robustness. Regular system updates, considering evolving attack patterns and adapting to changes in network behavior, are crucial. Lastly, evaluating system performance using diverse datasets ensures accuracy, robustness, and generalizability. By following these recommendations, researchers and practitioners can develop more effective and reliable systems for attack detection, thus strengthening network security in practical deployments.

REFERENCES

[1] P. Singh, Y. K. Dwivedi, K. S. Kahlon, A. Pathania, and R. S. Sawhney, “Can twitter analytics predict election outcome? An insight from 2017 Punjab assembly elections,” *Gov Inf Q*, vol. 37, no. 2, 2020, doi: 10.1016/j.giq.2019.101444.

[2] L. Ni and J. Liu, “A Framework for Domain-Specific Natural Language Information Brokerage,” *J Syst Sci Syst Eng*, vol. 27, no. 5, 2018, doi: 10.1007/s11518-018-5389-1.

[3] E. Beulen, A. Plugge, and J. van Hillegersberg, “Formal and relational governance of artificial intelligence outsourcing,” *Information Systems and e-Business Management*, vol. 20, no. 4, 2022, doi: 10.1007/s10257-022-00562-7.

[4] S. Hajiheidari, K. Wakil, M. Badri, and N. J. Navimipour, “Intrusion detection systems in the Internet of things: A comprehensive investigation,” *Computer Networks*, vol. 160. 2019. doi: 10.1016/j.comnet.2019.05.014.

[5] Z. Shah, I. Ullah, H. Li, A. Levula, and K. Khurshid, “Blockchain Based Solutions to Mitigate Distributed Denial of Service (DDoS) Attacks in the Internet of Things (IoT): A Survey,” *Sensors*, vol. 22, no. 3. 2022. doi: 10.3390/s22031094.

[6] F. Yihunie, E. Abdelfattah, and A. Odeh, "Analysis of ping of death DoS and DDoS attacks," in *2018 IEEE Long Island Systems, Applications and Technology Conference, LISAT 2018*, 2018. doi: 10.1109/LISAT.2018.8378010.

[7] Z. Chen, F. Han, J. Cao, X. Jiang, and S. Chen, "Cloud computing-based forensic analysis for collaborative network security management system," *Tsinghua Sci Technol*, vol. 18, no. 1, 2013, doi: 10.1109/TST.2013.6449406.

[8] R. Sharma and A. Thakral, "Identifying botnets: Classification and detection," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 9 Special Issue, 2019, doi: 10.35940/ijitee.I1021.0789S19.

[9] Y. Cui and Q. Qian, "MIND: Message classification based controller scheduling method for resisting DDoS Attack in Software-Defined Networking," in *2020 5th International Conference on Computer and Communication Systems, ICCCS 2020*, 2020. doi: 10.1109/ICCCS49078.2020.9118597.

[10] F. Mukhametzyanov, A. S. Katasev, A. M. Akhmetvaleev, and D. V. Kataseva, "The neural network model of DDoS attacks identification for information management fail," *International Journal of Supply Chain Management*, vol. 8, no. 5, 2019.

[11] L. Brikh, O. Guenounou, and T. Bakir, "Selection of Minimum Rules from a Fuzzy TSK Model Using a PSO–FCM Combination," *Journal of Control, Automation and Electrical Systems*, vol. 34, no. 2, 2023, doi: 10.1007/s40313-022-00975-2.

[12] J. Bonato, Z. Mrak, and M. Badurina, "Speed regulation in fan rotation using fuzzy inference system," *Pomorstvo*, vol. 29, no. 1, 2015.

[13] J. Singh and M. J. Nene, "A Survey on Machine Learning Techniques for Intrusion Detection Systems," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 11, 2013.

[14] T. Raja Sree and S. Mary Saira Bhanu, "Detection of HTTP flooding attacks in cloud using fuzzy bat clustering," *Neural Comput Appl*, vol. 32, no. 13, 2020, doi: 10.1007/s00521-019-04473-6.

[15] M. Bhopale, A. Kshatriya, P. Kumar, and R. Jagdeesh Kanan, "Fuzzy based system for analysis of DDOS attacks," *International Journal of Pharmacy and Technology*, vol. 8, no. 3, 2016.

[16] V. de M. Rios, P. R. M. Inácio, D. Magoni, and M. M. Freire, "Detection of reduction-of-quality DDoS attacks using Fuzzy Logic and machine learning algorithms," *Computer Networks*, vol. 186, 2021, doi: 10.1016/j.comnet.2020.107792.

[17] A. Maslan, K. M. Bin Mohamad, and F. B. Mohd Foozy, "Feature selection for DDoS detection using classification machine learning techniques," *IAES International Journal of Artificial Intelligence*, vol. 9, no. 1, 2020, doi: 10.11591/ijai.v9.i1.pp137-145.

[18] K. J. Singh, K. Thongam, and T. De, "Detection and differentiation of application layer DDoS attack from flash events using fuzzy-GA computation," *IET Inf Secur*, vol. 12, no. 6, 2018, doi: 10.1049/iet-ifs.2017.0500.

[19] I. H. Sarker, "CyberLearning: Effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks," *Internet of Things (Netherlands)*, vol. 14, 2021, doi: 10.1016/j.iot.2021.100393.

[20] V. Gaur and R. Kumar, "Analysis of Machine Learning Classifiers for Early Detection of DDoS Attacks on IoT Devices," *Arab J Sci Eng*, vol. 47, no. 2, 2022, doi: 10.1007/s13369-021-05947-3.

[21] L. Pan and Y. Deng, "A novel similarity measure in intuitionistic fuzzy sets and its applications," *Eng Appl Artif Intell*, vol. 107, 2022, doi: 10.1016/j.engappai.2021.104512.

[22] M. A. Taha and L. Ibrahim, "Traffic simulation system based on fuzzy logic," in *Procedia Computer Science*, 2012. doi: 10.1016/j.procs.2012.09.084.

[23] A. H. Agustin, G. K. Gandhiadi, and Oka Tjokorda Bagus, "Penerapan Metode Fuzzy Sugeno Untuk Menentukan Harga Jual Sepeda Motor Bekas," *E-Jurnal Matematika*, vol. 5, no. 4, 2016, doi: 10.24843/mtk.2016.v05.i04.p138.

[24] M. Ivanov *et al.*, "Fuzzy modelling of big data of HR in the conditions of industry 4.0," in *CEUR Workshop Proceedings*, 2020.

BIBLIOGRAPHY OF AUTHORS

Muhammad Adam Hawari is an undergraduate student from the Informatics Engineering Department at Institut Teknologi Telkom Purwokerto. His research primarily revolves around network security and artificial intelligence, showcasing his keen interest and expertise in these domains.

Wahyu Adi Prabowo is a faculty member in the Department of Informatics Engineering at Institut Teknologi Telkom Purwokerto. He specializes in the fields of Cyber Security, Digital Forensic, IT Governance, Enterprise system, and IT Security Governance.

Rifki Adhitama is a faculty member in the Department of Software Engineering at Institut Teknologi Telkom Purwokerto. He specializes in the fields of Software Engineering & Topic Modelling.