

A Fuzzy Logic Based Network Intrusion Detection System for Predicting the TCP SYN Flooding Attack

Nenekazi Nokuthala Penelope Mkuzangwe^{1,2(✉)}
and Fulufhelo Vincent Nelwamondo^{1,2}

¹ Modelling and Digital Science, Council for Scientific and Industrial Research,
CSIR Main Site, Meiring Naude Road, Pretoria, South Africa

² Department of Electrical and Electronic Engineering,
University of Johannesburg, Corner of Kingsway and University Road,
Auckland Park, Johannesburg, South Africa
mmkuza@gmail.com

Abstract. Fuzzy logic is one of the powerful tools for reasoning under uncertainty and since uncertainty is an intrinsic characteristic of intrusion analysis, Fuzzy logic is therefore an appropriate tool to use to analyze intrusions in a Network. This paper presents a fuzzy logic based network intrusion detection system to predict neptune which is a type of a Transmission Control Protocol Synchronized (TCP SYN) flooding attack. The performance of the proposed fuzzy logic based system is compared to that of a decision tree which is one of the well-known machine learning techniques. The results indicate that the performance difference, in terms of predicting the proportion of attacks in the data, of the proposed system with respect to the decision tree is negligible.

Keywords: Fuzzy logic · Intrusion detection · Network intrusion detection system · Neptune · Decision tree · TCP SYN flooding attack

1 Introduction

Intrusion detection systems (IDSes) are software applications or hardware systems that perform intrusion detection in information systems. They may be classified according to the source of those events that they monitor, for example, network events or host events. They may also be classified based on the method they use to perform detection. In general two detection methods exists, namely, misuse detection and anomaly detection. In misuse detection, all know attack patterns are defined and the IDS is trained to recognise them. The misuse detection method is good in detecting known attacks, however, it is unable to detect new attacks since it will not have defined patterns for the new attacks. In anomaly detection the normal behaviour of network traffic is modelled and any network traffic that deviates from the normal behaviour would be defined as anomalous which may mean the network is under attack. The anomaly based detection can detect new

attacks, however, it generates a lot of false alarms. The effectiveness of an intrusion detection system (IDS) is evaluated based on its ability to correctly classify events to be attacks or normal network behaviour [1].

Intrusion detection research community has proposed various intrusion detection systems. These systems differ from each other mainly based on the techniques they employ to analyse data in order to classify it as intrusive or normal. These techniques have their origins, to mention a few, in statistical methods [2], machine learning methods [3] and data mining techniques [4]. Researchers also compare the performance of their IDSeS with IDSeS that are based on different techniques [3, 5].

In this study a fuzzy logic based network intrusion detection system for detecting neptune which is a type of a Transmission Control Protocol Synchronized (TCP SYN) flooding attack is presented. The TCP SYN flooding attack is of interest in this work since it is the most widespread denial of service (DOS) attack and a serious threat to organizations that provide online services. The performance of the proposed intrusion detection system is compared to that of a decision tree.

There rest of this paper is arranged as follows. In Sect. 2 the work done in detecting denial of service attack using fuzzy logic is presented. Section 3 gives a brief description of the decision tree and the fuzzy logic. Section 4 discusses the dataset that is used in this study. In Sect. 5 the proposed fuzzy logic based network intrusion detection system is presented. Sections 6, 7, 8 and 9 present the experimental work, results, discussion and conclusion.

2 Related Work

This section presents the fuzzy logic based techniques for detecting denial of service attacks. Tuncer and Tatar [5] propose a fuzzy logic based system for detecting SYN flooding attacks. They compared the performance of their proposed system with Cumulative Sum (CUSUM) algorithm. Their system yielded results that are comparable to existing IDS. Kanlayasiri and Sanguanpong [6] presented a BENEf (Behavior Statistic, Network Information Based and Fuzzy logic Decision) model. In this model the important features are extracted from the network packets. These features are used in the pre detector component to analyze events with the set of rules to determine if those events have intrusive patterns or not. The anomalous events are passed to the Decision Engine that uses fuzzy logic principle to calculate the possibility of intrusion. They used the model to detect the TCP SYN flooding attack. Their results were recorded as the percentage of intrusion possibility. It would have been interesting to know how far their prediction was from the ground truth. Gao and Zhou [7] proposed a method for detecting intrusion based on fuzzy rule-based technique. A Fuzzy Reasoning Petri Nets (FRPN) model is used to represent fuzzy rule base and derive the final detection decision. They implemented their model in detecting the TCP SYN flooding attack. Their results were recorded as the percentage of intrusion possibility but this does not tell much about the accuracy of their system i.e. how far was their prediction from the ground truth. Shanmugavadivu

and Nagarajan [8] have developed an anomaly based intrusion detection system in detecting the intrusion behaviour within a network. A fuzzy decision-making module was designed using the fuzzy inference approach to detect the network attacks. They used automated strategy for the generation of fuzzy rules. They used KDD99 dataset to evaluate the performance of their system. The experimental results indicate that their system is able to effectively detect all four attacks types found in this dataset with accuracy of more than 90 percent. This supports the notion that fuzzy logic is appropriate for use in intrusion detection.

Our work implements a fuzzy logic based IDS to detect the TCP SYN flooding attack and it differs from the work done in [5–7] in terms of the dataset used. In this work the NSL KDD dataset is used whereas [5] observed the connection request coming to the Firat university web server for SYN flooding attack detection and TCP packets coming to port 80 were collected every specified seconds using ethereal application and [6, 7] did not specify the dataset they have used. [8] implemented a fuzzy logic based IDS to detect all four categories of attacks that exist in the KDD99 dataset and their work differs from ours in that they detected a denial of service attack as a whole while we are proposing a system to detect a particular denial of service attack in the NSL KDD dataset.

3 A Brief Overview of the Decision Tree and Fuzzy Logic

In this work a fuzzy logic based system is presented and its performance is compared to that of a decision tree therefore this section gives a refresher on decision trees and fuzzy logic.

3.1 Decision Tree

Decision trees are among the well-known machine learning techniques. A decision tree consists of three basic elements. Namely a decision node specifying a test attribute, a branch corresponding to the one of the possible test attribute values and a leaf which contains the class to which the object belongs. In decision trees, two major phases should be ensured:

- Building the tree. A decision tree is built based on a given training set. It consists of selecting the appropriate test attribute for each decision node and also to define the class labelling each leaf.
- Classification. Classification of a new case starts from the root of the decision tree where the attribute specified by this node is tested. The result of this test allows to move down the tree branch relative to the attribute value of the given case. This process is repeated until a leaf is reached. The case is then classified in the class that is described by the reached leaf.

Numerous algorithms have been developed to construct decision trees. The ID3 and C4.5 algorithms developed by Quinlan [9] are probably the most popular ones. There is also the CART algorithm of Breiman et al. [10].

3.2 Fuzzy Logic

The Fuzzy logic concept was conceived by professor Lofti Zadeh as a way of processing data by allowing partial set membership rather than crisp set membership or non-membership. It provides a very valuable flexibility for reasoning that takes inaccuracies and uncertainties into account [11]. It provides a simple way of arriving to a definite conclusion based upon vague, ambiguous, noisy, imprecise or missing input information. The process of reaching this definite conclusion is described below.

- All input values are fuzzified into fuzzy membership functions.
- Fuzzy rules are generated. The rules are in the form of IF THEN statement.
- Given an instance, some of the fuzzy rules will be activated.
- The activated rules are combined in the rule base to compute the fuzzy output distribution.
- The fuzzy output distribution is defuzzified to obtain a crisp output value.

4 Dataset

The NSL KDD [12] is the dataset that was used in this study. The NSL KDD dataset was generated from the KDD99 dataset [13] by removing redundant and duplicate instances and reducing the size of the dataset. The KDD99 dataset is a revised version of the DARPA 98 MIT Lincoln Lab dataset [14] that was summarized into network connections where each connection is a single row vector consisting of 41 features and is marked as either normal or an attack with exactly one particular type of attack. The network connections are referred to as cases in this work. The different attacks included in the dataset fall into four categories, namely, denial of service attack, remote to user attack, user to root attack and probes.

In this study the NSL KDD training and test data were filtered for normal and neptune (a type of a denial of service attack) connections that are referred to as normal and neptune cases respectively. The training data consisted of 108558 cases with 67343 normal cases and 41215 neptune cases. The test data consisted of 14368 cases with 9711 normal cases and 4657 neptune cases. Our interest is to predict the actual proportion of neptune cases in the test data where the actual proportion of neptune cases in the test data is the number of neptune cases in the test data divided by the number of cases in the test data.

The NSL KDD dataset has 41 attributes and we had to decide on the attributes to use to detect neptune. [15] recommended ten attributes as relevant in identifying neptune. In this studied we initially selected four of those attributes to train and test our system. These attributes were % of connections that have SYN errors, % of connections to the same service, % of connections to the different service and count of connections having the same destination host and using the same service. In this work they are denoted as SynErrorRate, SameSrvRate, DiffSrvRate and DStHostSrvCount respectively. However, we were not satisfied with the performance of our system. Furthermore, when we trained the decision tree using

the four attributes the resulting decision tree did not include DstHostSrvCount attribute. This happens when the attribute does not improve the accuracy of the decision tree [16] which means this attribute is redundant and including a redundant attribute badly affects the accuracy of a classifier [17]. Based on the trained decision tree we decided to implement our proposed system using the attributes that were included in the trained decision tree and the performance of our system improved.

5 The Proposed Fuzzy Logic Based Network Intrusion Detection System

In this section the proposed fuzzy logic based network intrusion detection system used in detecting neptune is presented. In which the membership values of the three attributes and the output and their membership functions are defined, the rules are generated and the fuzzy inferencing and defuzzification methods are described. The implementation of this system was done in Matlab.

5.1 Fuzzification and Membership Functions

To derive the membership values for each of the attributes, we observed the range of values each attribute takes and calculated the average value for each attribute in the normal, attack and mixed (contains both normal and attack) data of the training data. Based on the minimum, average and maximum values of each attributes, three membership values L (low), M (medium) and H (high) were defined. The triangular membership function was used to define membership index associated with each membership value of the three attributes as shown in Fig. 1. The output is in the form of percentage of intrusion in the data (% Intrusion). It is also defined into three membership values, namely, L (low), M (medium) and H (high).

5.2 Fuzzy Rules Generation

The rules were created in the form of an IF THEN statement. We enumerate all possible permutations of the membership values of the three attributes. At first all twenty seven permutations were used to create the rules, however, for some permutations it was difficult to infer the consequent which led to a poor performance of the system. We decided to select only the permutations that led to an easy way to deduce the consequent of the rules and it resulted to only nine permutations. The permutations were used as the antecedent for each rule. From the training data we observed the average value for each attribute in the normal, attack and mixed data and noticed that for some attributes the average value decreased in the presence of attacks while it increased for some attributes. The consequent of each rule was then based on the behavior of the average value of each attribute in the presence or absence of an attack. An example of the rules is given below.

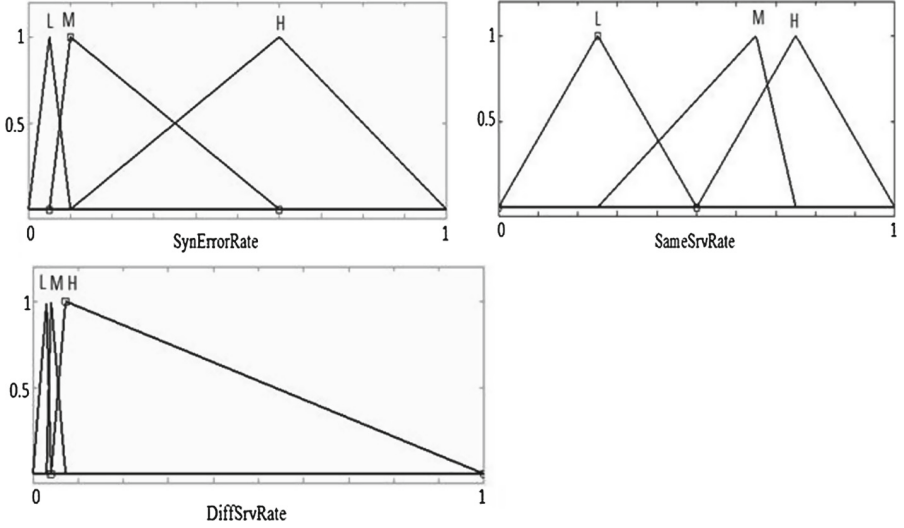


Fig. 1. The membership functions of the proposed system

IF Average of SynErrorRate = H AND Average SameSrvRate = L AND Average of DiffSrvRate = H THEN % Intrusion = H.

5.3 Fuzzy Inferencing and Defuzzification

The activated rules were aggregated using the Mamdani fuzzy inferencing that is explained in [18] and the crisp value of the output was obtained using the centroid approach that returns the centre of the area under the curve of a membership function.

6 Experimental Work

This section outlines how neptune was predicted using the decision tree and the proposed system and defines the performance metrics used in this study.

6.1 Decision Tree Construction

The decision tree was constructed in R Studio using the processed training data set as described in Sect. 4. The resulting decision tree is presented in Fig. 2. The constructed decision tree was used to predict neptune from the processed test data.

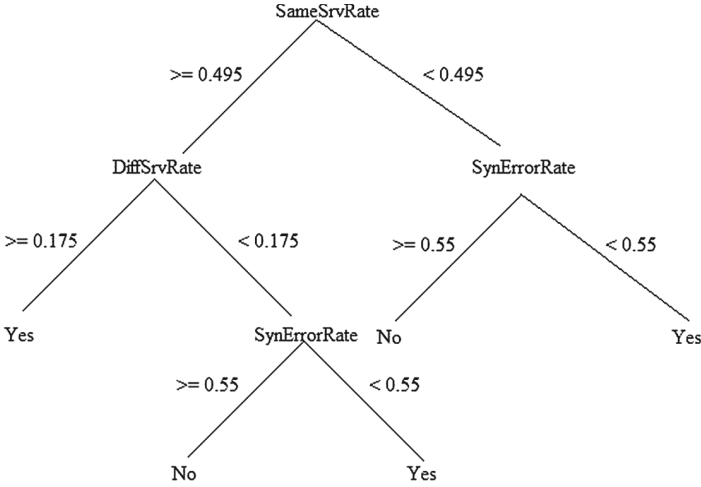


Fig. 2. The decision tree.

6.2 Prediction with the Proposed System

For each attribute an average was calculated from the test data. These averages were tested against the generated rules of the proposed system to predict the percentage of intrusion in the test data. Figure 3 illustrates the crisp value of the output variable (% intrusion) where the solid vertical black line corresponds to the crisp value of the output variable.

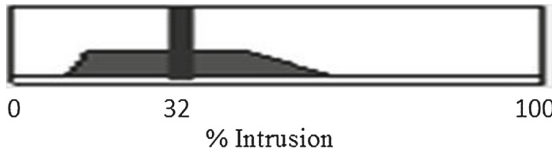


Fig. 3. The crisp value of the output variable, % intrusion.

6.3 Performance Metrics

The performance metrics that are used in this study are the proportion of attacks predicted by the two algorithms in the test data where each algorithm predicts the actual proportion of neptune cases in the test data and the accuracy of the two algorithms where accuracy is the sum of the true positives and true negatives divided by the sum of the true positives, false positives, false negatives and true negatives of the two algorithms.

7 Results

The actual proportion of attacks (neptune cases) in the test data is 0.3241. Table 1 presents the predicted proportion of attacks by the two algorithms and their accuracies.

Table 1. The results of the proposed system and the decision tree

Performance metrics	Proposed system	Decision tree
Predicted proportion of attacks	0.3200	0.3241
Accuracy	0.9324	0.9980

The results in Table 1 indicate that the performance difference, in terms of predicting the proportion of neptune cases in the test data, of the proposed system with respect to the decision tree is negligible and the accuracy of the decision tree is better than that of the proposed system.

8 Discussion

Our method generate the rules using a simple strategy where we enumerated permutation of the features of interest and decided on the output based on the behavior of each feature in the permutation. The results indicate that the performance difference, in terms of predicting the proportion of neptune cases in the test data, of the proposed system with respect to the decision tree is negligible and the accuracy of the decision tree is better than that of the proposed system. However, decision trees are known to be highly unstable with respect to minor changes in the training data whereas fuzzy logic based system may be more stable in the presence of minor changes in the training data due to the elasticity of the fuzzy sets [19]. This gives a fuzzy logic based system an advantage over a decision tree which means the proposed system has a higher chance of yielding the same accuracy if the training data can be slightly changed as compared to the decision tree.

9 Conclusion

We proposed a simple fuzzy logic network intrusion detection system to detect neptune which is a type of TCP SYN flooding attack. The NSL KDD dataset was used to train and evaluate our system. The performance of our system was compared to that of a decision tree. The results indicate that the performance difference, in terms of predicting the proportion of neptune cases in the test data, of the proposed system with respect to the decision tree is negligible.

Acknowledgments. I would like to thank my promoter Professor Fulufhelo Nelwamondo for his guidance and support and the CSIR: Modelling and Digital Science Unit for financially supporting my studies.

References

1. Wu, S.X., Banzhaf, W.: The use of computational intelligence in intrusion detection systems: a review. *Appl. Soft Comput. J.* **10**, 1–35 (2010)
2. Chin-Ling, C.: A new detection method for distributed denial of service attack traffic based on statistical test. *J. Univ. Comput. Sci.* **15**, 488–503 (2009)
3. Amor, N.B., Benferhat, S., Elouedi, Z.: Naive Bayes vs decision trees in intrusion detection systems. In: *Proceedings of the 2004 ACM Symposium on Applied Computing*, pp. 420–424. ACM (2004)
4. Kanwal, G., Rshma, C.: Detection of DDoS attacks using data mining. *Int. J. Comput. Bus. Res. (IJCBR)* **2**, 1–10 (2011)
5. Tuncer, T., Tatar, Y.: Detection SYN flooding attacks using fuzzy logic. In: *Proceedings of the International Conference on Information Security and Assurance, ISA 2008*, pp. 321–325. IEEE (2008)
6. Kanlayasiri, U., Sanguanpong, S.: Network-based intrusion detection model for detecting TCP SYN flooding. In: *Proceedings of the 4th National Computer Science and Engineering Conference, Bangkok, Thailand*, pp. 148–153 (2000)
7. Gao, M., Zhou, M.: Fuzzy intrusion detection based on fuzzy reasoning Petri nets. In: *IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, pp. 1272–1277. IEEE (2003)
8. Shanmugavadivu, R., Nagarajan, N.: Network intrusion detection system using fuzzy logic. *Indian J. Comput. Sci. Eng. (IJCSE)* **2**(1), 101–111 (2011)
9. Quinlan, J.R.: *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo (1993)
10. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification and Regression Trees*. Wadsworth and Brooks, Monterey (1984)
11. Introduction to fuzzy logic. http://www.francxy.me/doc/course/fuzzy_logic.pdf
12. Tavallae, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. In: *Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defence Applications* (2009)
13. KDD Cup 1999 Data. <https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>
14. DARPA Intrusion Detection Data Sets. <https://www.ll.mit.edu/ideval/data/>
15. Kayacik, H.G., Zincir-Heywood, A.Z., Heywood, M.I.: Selecting features for intrusion detection: a feature relevance analysis on KDD 99 intrusion detection datasets. In: *Proceedings of the Third Annual Conference on Privacy, Security and Trust* (2005)
16. Decision Tree Models. https://www.ibm.com/support/knowledgecenter/SS3RA7_15.0.0/com.ibm.spss.modeler.help/nodes.treebuilding.htm
17. Choudhary, N.K., Shinde, Y., Kannan, R., Venkatraman, V.: Impact of attribute selection on the accuracy of Multilayer Perceptron. *Int. J. IT Knowl. Manag. (IJITKM)* **7**(2), 32–36 (2014)
18. Fuzzy inference systems. <http://www.cs.princeton.edu/courses/archive/fall07/cos436/HIDDEN/Knapp/fuzzy004.htm>
19. Engineering Decision Trees in Fuzzy Logic ABSTRACT - International. <http://isindexing.com/isi/papers/1413180985.pdf>

Intelligent Information and Database Systems

9th Asian Conference, ACIIDS 2017, Kanazawa, Japan,

April 3-5, 2017, Proceedings, Part II

Nguyen, N.-T.; Tojo, S.; Nguyen, L.M.; Trawiński, B. (Eds.)

2017, XLIII, 827 p. 286 illus., Softcover

ISBN: 978-3-319-54429-8